

# Cycle-accurate Energy Measurement and High-Level Energy Characterization of FPGAs

Hyung Gyu Lee, Sungyuep Nam and Naehyuck Chang\*

*School of Computer Science & Engineering, Seoul National University, Korea*

*naehyuck@snu.ac.kr*

## Abstract

*Field programmable gate arrays (FPGAs) play many important roles, ranging from small glue logic replacement to System-on-Chip designs. Nevertheless, FPGA vendors can not accurately specify the energy consumption information of their products on the device data sheets because the energy consumption of FPGAs is strongly dependent on target circuit including resource utilization, logic partitioning, mapping, placement and route. While major CAD tools have started to report average power consumption under given transition activities, energy optimal FPGA design demands more detailed energy estimation.*

*In this paper, we introduce an in-house cycle-accurate energy measurement tool and energy characterization schemes from low level to operation level. The tool offers all the necessary capability to investigate the energy consumption of FPGAs for high-level, operation-based energy characterization, which is useful for high-level, system-wide energy estimation. It also includes features for low-level energy characterization. We compare our tool with Xilinx XPower and demonstrate state machine energy characterization of an LCD controller and an SDRAM controller.*

## 1. Introduction

Although SRAM-based FPGAs are naturally low-power, it does not mean that we are free from their power consumption. First, as gate counts of FPGAs increase, their power consumption becomes distinct from the system-wide view point. Secondly, their power behavior, not only quantitatively but qualitatively is invaluable for system-level energy reduction as well as future technology migration for final products. In this paper, we introduce an in-house energy measurement and characterization tool for SRAM-based FPGAs and demonstrate its applications. Our methods are not limited

\*Corresponding author.

\*The RIACT at Seoul National University provides research facilities for this study. This work was partly supported by the Brain Korea 21 Project.

to SRAM-based FPGAs, but we focus on SRAM-based FPGAs in this paper.

Power consumption is a mandatory information in modern digital system design. Chip vendors are naturally in charge of supplying energy consumption information of their products on the device data sheets. However, it is not possible for vendors to specify power consumption information of SRAM-based FPGAs because it is not only dependent on the target device and operating frequency but is highly dependent on the design and operating conditions. Power consumption is strongly dependent on the target circuit including resource utilization, low-level features such as logic partition, mapping, placement and route.

Power estimation of individual components is mandatory to determine system-wide power supply system design. Power supply system does not mean only the power supply unit but includes the entire power distribution system that ensures signal integrity. Recently, it has been found that power estimation is more useful when performed on a system wide basis and the results are used to apply high-level energy reduction techniques. A common power estimation method is based on a switching capacitance model and average activity factors. Activity factors are largely dependent on operating frequency when there is no distinct slack time, *i.e.* idle time of the device or that of a part of the device. FPGAs are commonly used for peripheral devices and their control logic whose activities are determined by memory transactions of a microprocessor, which are again determined by software running on the microprocessor. Generally, since timing behavior also significantly affects the energy consumption of the peripheral devices, average activity-based estimation may not be desirable for accurate system-wide power estimation. On the other hand, our in-house tool measures energy consumption based on a cycle-accurate measurement technique.

The ultimate goal of energy characterization of FPGAs is to minimize energy consumption of the FPGAs. There may be three-different strategies in energy minimization of FPGAs. First, once the architecture has been fixed, the designer may change the logic partitioning, mapping, placement and route. Secondly, the designer may change the high-level architectural design of the FPGA. Finally, although the designer does not change the FPGA design, there may be still promising energy reduction chances by enhancing the operating scheme of the FPGA. The last challenge requires accurate operation-based energy characterization of the FPGA. In this paper, we demonstrate the ability of our tool to perform energy characterization that will enable a full range of energy reduction techniques.

The rest of the paper introduces details of our in-house tool starting from the measurement circuit and energy calculation. We demonstrate low-level energy characterization of FPGA design

comparing the results with Xilinx XPower tool [1]. We introduce macro energy state machine and characterization results on an LCD controller and an SDRAM controller.

## 2. Related work

The easiest way of power estimation is to use an existing tool if available. Power estimation tool for standard cell-based design [2] is often used for FPGA power estimation with some modification [3, 4]. Since FPGA power consumption is quite different from standard cell-based design such as heavy interconnection power [5], a more elaborated method is desirable.

Real measurement gives the most accurate power information as far as the measurement is correct. It is, however, tricky in that the device must be a representative among the sample space, which means the selected device may have odd characteristics due to uncommon environmental condition during manufacturing. Traditional power supply current measurement is performed for a Xilinx FPGA [6] with a digital filter application. Similar experiment compares power consumption of a Xilinx FPGA with that of an Altera FPGA [7]. They measure the average power consumption [6, 7] and convert the power value to energy per unit operation [6].

Power estimation using a simulation-based tool is convenient but may not be accurate because it is based on switching capacitance and average switching activity. A power estimation tool, which is based on the above method, is implemented for Xilinx 4000 series FPGAs, but it does not include static power consumption model [8]. In advance, a detailed power analysis of Xilinx XC4003A FPGA is performed by a power estimator considering physical details such as CLB (Configurable Logic Block), routing paths and clock paths [9]. They are verified by measurement. A statistical method, based on switching capacitance, input pattern and input statistics, is also applied to an FPGA power estimator [10].

XPower is a first generation commercial-off-the-shelf tool to estimate power consumption of Xilinx SRAM-based FPGAs. In this paper, we compare our in-house tool with Xilinx XPower and thus describe more details of the XPower in this section. XPower reads in either pre-routed or post-routed design data, and then makes a power model either by net or for the overall device based on power equation:  $P = CV^2f$  where  $P$  is average power consumption,  $C$  is equivalent switching capacitance,  $V$  is supply voltage and  $f$  is operating clock frequency or toggle rate. It considers resource usage, toggle rates, input/output power, and many other factors in estimation. Because XPower is an estimation tool, results may not precisely match actual power consumption. The frequency,  $f$ , is determined by users or provided by simulation data from the ModelSim family of HDL (Hardware Description Language) simulators. In the absence of simulation information, the user is required to enter a clock frequency and estimated toggle rate percentage to be applied to all the signals in each path. [1].

XPower provides two types of information called data view and report view. The data view shows the power consumption of individual parts of a design such as signals, clocks, logic and outputs. The report view represents the total power consumed by a given design, which is again classified into power consumption of clocks, logic and outputs, and static (leakage) power. The power consumption of clocks, logic and outputs are calculated by equivalent switching capacitance models. The static power is based on constant value quoted in a data book or calculated by an equation associated with temperature, device utilization and supply voltage. The value quoted in the data book is in the worst case and thus generally results in overestimation. They continue to revise the static

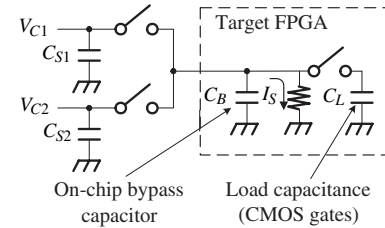


Figure 1: Cycle accurate energy measurement system for FPGAs .

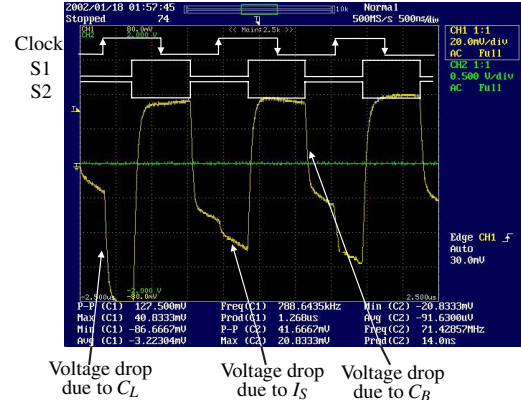


Figure 2: Waveform for cycle accurate energy measurement.

power estimation and go by typical values from 4.2i version.

We assume XPower is based on accurate power information of the physical details of the FPGA because it is designed by the chip vendor and is fully integrated with the FPGA design tool. However, it is similar to the estimators mentioned above, which are based on the lumped capacitance model; it provides only average energy consumption. Although average energy consumption is useful information to estimate the energy consumption for FPGAs, it is not sufficient to reduce the energy consumption using a high-level approach. Accurate system-wide power estimation, which may inspire proper power reduction strategy, often uses real application traces as the testbench. But lumped capacitance model with average activity does not fully utilize the testbench such as timing behavior, address and data values.

Altera has also announced a power estimation tool for their FPGAs. They consider device-dependent parameters for the power supply current, the number of used macrocells, the number of the total macrocells, the maximum operating frequency and the average rate of the logic cells in the FPGA core. External power calculation requires the average capacitance of the output, DC output load current and average toggle rate of the output [11]. Most of all, the user must specify the average toggle rate of the internal logic cell and the output pins.

## 3. Cycle-accurate energy measurement

### 3.1 Theory of operation

The switched capacitor method [12] is ideal for energy measurement of SRAM-based FPGAs because most designs are synchronous to the system clock. We have added many features to

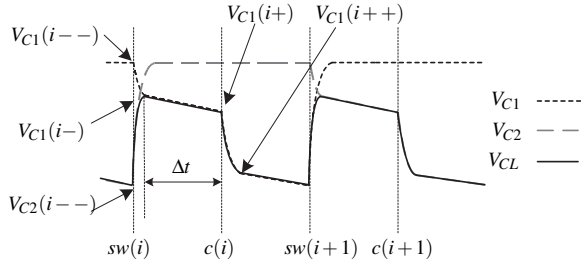


Figure 3: Detailed waveform for energy calculation.

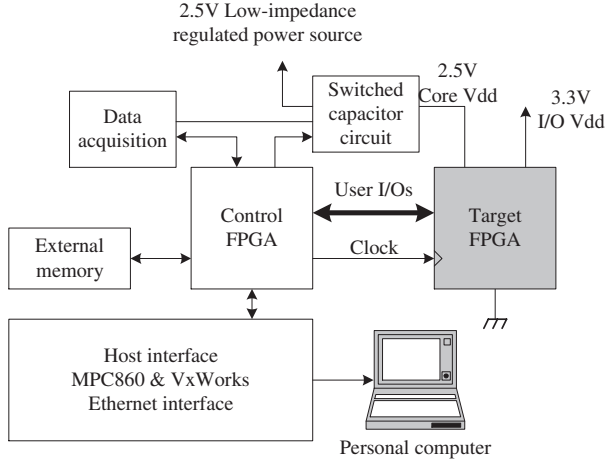


Figure 4: Real-time cycle-accurate energy measurement system for SRAM-based FPGAs.

the existing energy measurement system for investigating energy behavior and developing guidelines for proper trade-offs in the design space. Fig. 1 illustrates the theory of operation for the energy measurement by switched capacitors [12]. Since the Xilinx FPGAs consume distinct leakage power, we add a parallel resistor model in the target equivalent circuit. There are on-chip bypass capacitors for mitigating power supply fluctuation, which make energy calculation complex. The load capacitance is periodically connected to the power supply line when every clock edge arrives. Fig. 2 shows the real waveform of the measurement setup captured by high-performance digital storage oscilloscope. Depending on design, the amount of voltage drop is variable. Dynamic energy consumption causes the major voltage drop that appears on the switched capacitors. The slope of the continuous voltage droop denotes leakage power consumption.

Fig. 3 illustrates notations for the energy calculation. The voltage of the two capacitors,  $C_{S1}$  and  $C_{S2}$  in Fig. 1, is denoted by  $V_{C1}(\cdot)$  and  $V_{C2}(\cdot)$ , respectively. The argument,  $\cdot$ , denotes four-different states of the capacitor currently supplying power to the target circuit ( $C_1$ ):  $(--)$ ,  $(-)$ ,  $(+)$  and  $(++)$  which denote fully charged, connected to the on-chip bypass capacitor,  $C_B$ , discharged by the leakage energy consumption, and discharged by the dynamic energy consumption. At the same time,  $C_2$  is discharged at  $(--)$  and remains in fully charged state for  $(-)$ ,  $(+)$  and  $(++)$ .

First, we calculate the capacitance of the on-chip bypass capacitor,  $C_B$ . Generally, We have no prior knowledge of the on-chip

Table 1: Specification of the In-house measurement system.

Target FPGA: Xilinx SpartanII XC2S50TG144
Target control FPGA: Xilinx SpartanII XC2S150FG456
Data acquisition FPGA: Xilinx SpartanII XC2S150FG456
Vector and configuration memory: Samsung SRAM 256 KByte
Data acquisition memory: Samsung SRAM 256 KByte
ADC resolution: 10 Bit ADC @50MS/s
Data transfer method: TCP/IP communication

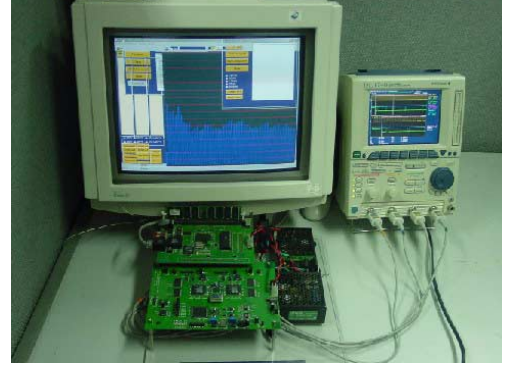


Figure 5: In-house energy measurement system.

bypass capacitor. It is determined by the charge sharing rule:

$$C_B = \frac{V_{C1}(i-)C_{S1} - V_{C1}(i--)C_{S1}}{V_{C2}(i--) - V_{C1}(i-)} \quad (1)$$

The static or leakage energy consumption is denoted by the slope of the waveform. Let us denote the static energy of  $i$ -th clock cycle by  $E_S(i)$ :

$$E_S(i) = \frac{1}{2}(C_{S1} + C_B) \frac{V_{c1}(i-)^2 - V_{c1}(i+)^2}{\Delta t} \quad (2)$$

We eliminate  $\Delta t$  by converting the static power to energy consumption for the clock period,  $\tau$ . It turns out that the static energy consumption is constant, *i.e.*  $E_{S0} = E_{S1} = \dots = E_{Sn}$ . The dynamic energy of  $i$ -th clock cycle,  $E_D(i)$ , is denoted by

$$E_D(i) = \frac{1}{2}(C_{S1} + C_B)(V_{c1}(i+)^2 - V_{c1}(i+)^2) \quad (3)$$

Finally, the total energy consumption is determined by

$$E_{TOT} = \sum_{i=0}^n (E_D(i) + \tau E_S(i)) = \sum_{i=0}^n E_D(i) + n\tau E_S \quad (4)$$

### 3.2 In-house measurement tool

We develop an in-house energy measurement tool for Xilinx FPGAs based on the measurement circuit in Fig. 1. The tool is fully integrated with an automatic data acquisition system consisting of pipelined A/D converters, a vector generator, a system management CPU, network interface and PC-based software (Fig. 4). Table 1 summarizes the specification of the in-house tool.

Fig. 5 shows a photograph of our tool. The tool also has many convenient features such as bit-stream download without the Xilinx

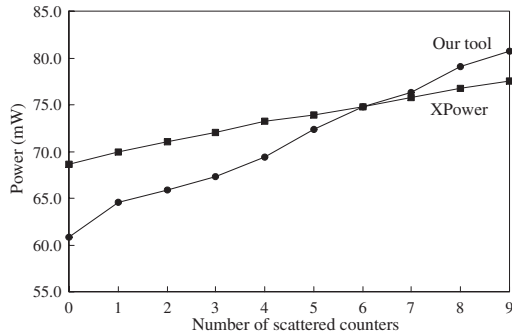


Figure 6: Power consumption against the P&R methods.

XChecker or the JTAG cables, which simplifies measurement process and enhances efficiency of handling complex, repetitive measurement.

## 4. Energy characterization of FPGAs

### 4.1 Low-level characterization

Low-level characterization reflects detailed energy variation due to physical mapping of the logic. Low-level power optimization of FPGA design changes physical implementation such as look-up table input variable reordering [13] and routing high-fan-out nets to low-capacitance paths [14].

Although our tool measures the energy consumption of the whole FPGA core, it offers enough functionality for detailed low-level energy characterization against actual physical implementation. Previous FPGA power estimators can also perform this kind of low-level power characterization if users spend valuable time on slow simulation. However, measurement-based approach gathers the results on the fly. Our tool supports all the necessary features for the low-level energy characterization because it basically measures cycle-accurate energy consumption.

SRAM-based FPGAs consume about 65% of the total energy for programmable interconnections [5]. To confirm that our tool is suitable for verifying power variation due to the interconnection length, we measure the power consumption of the switch matrices. We implement many 4-bit binary counters with a Xilinx Spartan FPGA. We start from an optimized physical implementation and scatter the logic blocks with the Floorplanner. Fig. 6 shows that power consumption increases due to the number of scattered counters. Next, we try more specific power characterization against P&R (Placement and Route) results. The more scattered counters result in the longer routing paths and thus the more switch matrices. Fig. 7 shows power variation due to the number of the switch matrices experienced by the same designs. We perform the same experiments with XPower and compare them with our measurement results. Note that the static energy is not variable to the design. Of course, we verify the experimental results with a digital multimeter and confirm that our results match multimeter results.

We demonstrate the higher level energy characterization with a Block RAM implementation of Xilinx Spartan II FPGA. We implement a  $256 \times 16$  Block RAM and measure the cycle-accurate energy consumption. Read and write energy turns out to be independent of the address values. On the other hand, read energy is proportional to the Hamming distance between the current and the pre-

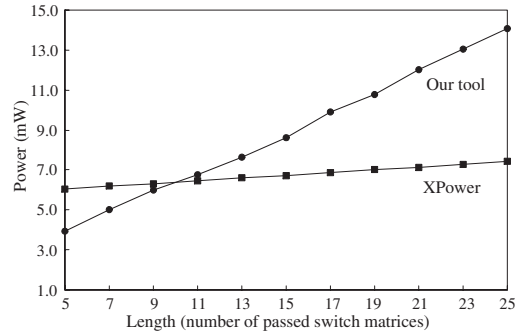


Figure 7: Power consumption against the number of the switch matrices.

vious access data values. Write energy is also variable to the Hamming distance between the current write data and the previous access data whether the previous operation is read or write. We derive an analytical model for the read and write energy. Let us denote the number of zero-to-one transitions and one-to-zero transitions between the data values  $d_n$  and  $d_{n-1}$  by  $f_{0 \rightarrow 1}(d_n, d_{n-1})$  and  $f_{1 \rightarrow 0}(d_n, d_{n-1})$ , respectively. Read energy,  $E_R$ , is denoted by

$$E_R = 0.02f_{0 \rightarrow 1}(d_n, d_{n-1}) + 0.06f_{1 \rightarrow 0}(d_n, d_{n-1}) + 0.20N_B + 0.22 + 2.62 \cdot 10^{-3} \tau \text{ (nJ)}. \quad (5)$$

Write energy,  $E_W$ , is given by

$$E_W = 0.02f_{0 \rightarrow 1}(d_n, d_{n-1}) + 0.06f_{1 \rightarrow 0}(d_n, d_{n-1}) + 0.19N_B + 0.23 + 2.62 \cdot 10^{-3} \tau \text{ (nJ)}. \quad (6)$$

The number of Block RAMs and the clock period are denoted by  $N_B$  and  $\tau$ , respectively. During the idle state, it consumes  $0.26 + 2.62 \cdot 10^{-3} \tau$  (nJ) per clock.

### 4.2 Operation-based characterization

More importantly, high-level energy characterization is mandatory for system-wide energy optimization. System-wide energy consumption is highly dependent on access patterns and the way of access to the peripheral components. They are governed by software such as operating systems, application programs, runtime data and user behavior if the application is interactive. This sort of system-wide behavior is very complex and thus low-level energy estimation is not practical. Fig. 8 shows a method for high-level energy characterization of 4-bit binary counters implemented in Xilinx Spartan FPGA. Here we characterize the dynamic energy consumption of the binary counter. We characterize the energy variation by the Hamming distance of the flip flops in the logic blocks. We verify the energy variation by increasing the number of the 4-bit binary counters from 10 to 20. This characterization is useful to estimate energy consumption of binary counters which are not free running and sometimes may be set or cleared on certain conditions. Unfortunately, previous FPGA power estimators must perform many iterations to have this kind of energy characterization because they are generally based on the switching capacitance models with activity information. For example, we need to control the counter to repeat the 000B to 001B to have the energy data for the 1-bit Hamming distance change. In addition, only limited peripheral components allow such a repeated internal state change.

High-level state machine-based energy characterization [15] is ideal for the system-wide energy estimation because it exactly distinguishes energy variation with separate static and dynamic energy

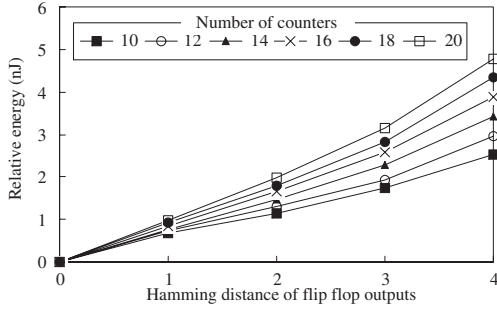


Figure 8: Verifying dynamic energy consumption against the Hamming distance of 4 bit counters.

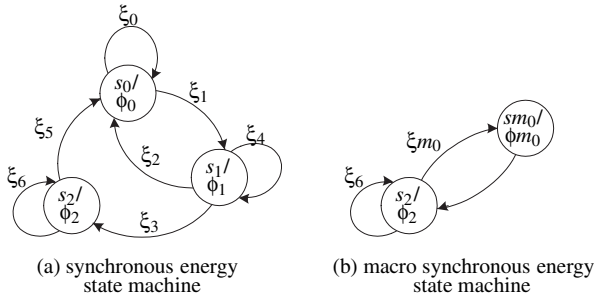


Figure 9: Macro energy state machine.

consumption models. Generally, the state machine-based energy characterization is impossible when using the previous switching capacitance and activity sensitive power estimators such as Xilinx XPower. High-level energy reduction does not try to change the design and thus the energy consumption of a lower-level design. For instance, software energy reduction does not try to change hardware design to save energy consumption. Rather, the designer tries to avoid bad usage of the low-level design that results in heavy energy consumption. This means that average power information is not helpful for high-level energy reduction.

Energy state machines directly represent the clock-cycle behavior of the FPGA circuit. It offers precise energy characteristics without any information loss due to abstraction of synchronous circuits. However, sometimes it may be too detailed information for system-level energy optimization. Fig. 9 introduces a macro energy state machine. Fig. 9 (b) merges the two states,  $s_0$  and  $s_1$  in

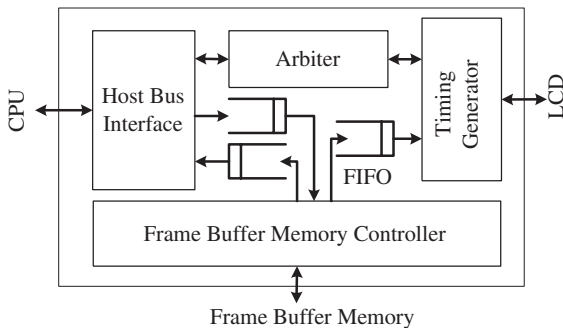


Figure 10: Block diagram of an LCD controller.

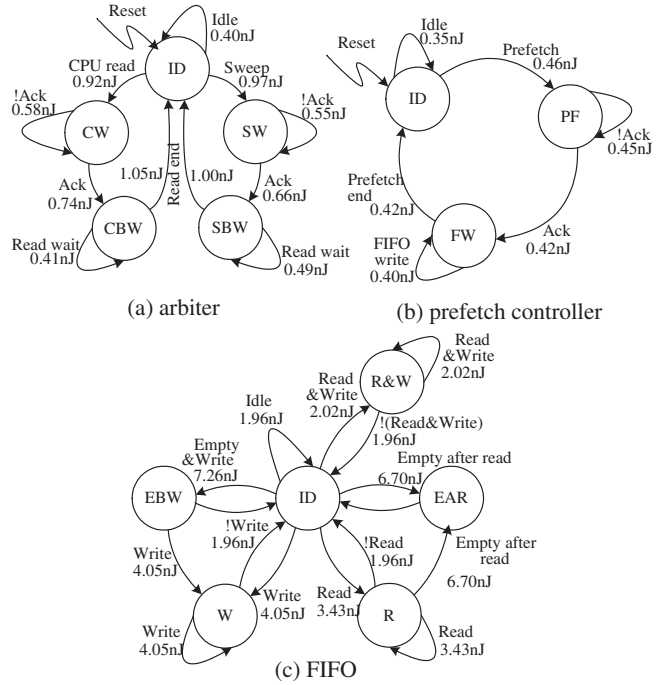


Figure 11: State machine characterization of the LCD controller.

Fig. 9 (a), in to a macro state,  $sm_0$ . Note that the dynamic energy to and from the macro state does not have meaning anymore. As far as we can calculate the correct dynamic energy and static energy by the clock frequency and way of operation, we may summarize the total dynamic and static energy of the macro state and map them to the either outgoing or incoming edge and the macro state, respectively.

In this paper, we select two popular components, an LCD controller and an SDRAM controller and perform high-level energy characterization. Fig. 10 shows the structure of the LCD controller. It consists of a host bus interface, a frame buffer memory controller, an arbiter between the host bus interface and the sweeper, video timing generator and FIFOs. We use a Xilinx Spartan II FPGA and use 70 slices, 33 Slice flip flops, 122 four input LUTs and one GCLK for the arbiter, 457 Slices, 34 Slice flip flops, 672 four input LUTs and 176 LUTs as shift registers for the FIFO, and 8 Slices, 12 Slice flip flops, 14 four input LUTs and one GCLK for the prefetch controller. The prefetch controller is a part of the frame buffer memory controller.

It is difficult to accurately explain the energy consumption of an LCD controller by average activity factors and equivalent switching capacitance because of its complex operation. It is quite similar to that it is dangerous to handle network performance only with average traffic. Trace-driven simulation much more accurately estimates the energy consumption as far as the energy model is suitable to utilize the information. The software traces give access patterns of the LCD controller from the CPU. But it is not easy to determine activity factors of each internal components such as the arbiter and the frame buffer memory controller with the traces. We simplify the state diagram to be suitable for trace-based energy estimation. We actually operate the LCD controller using the in-house tool and measure the cycle-accurate energy and complete the energy state



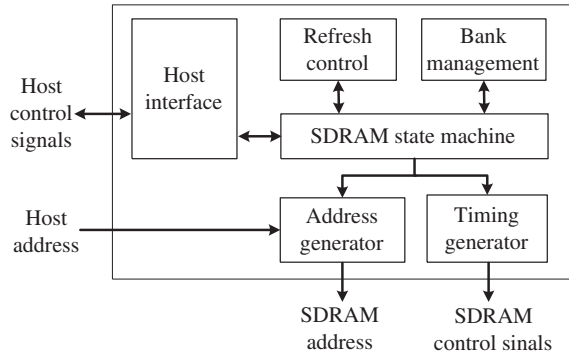


Figure 12: Block diagram of an SDRAM controller.

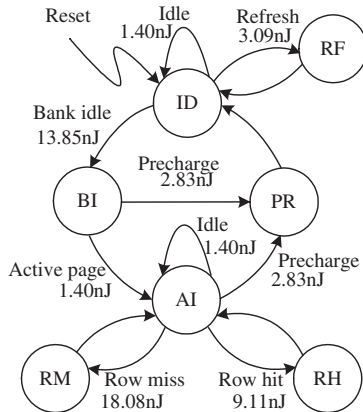


Figure 13: State machine characterization of the SDRAM controller.

machine as shown in Fig. 11.

In today's market, SDRAM is a virtually standard memory device from battery-operated hand-held devices to enterprise servers. Fig. 12 illustrates the structure of the SDRAM controller. We use 173 Slices, 115 Slice flip flops, 316 four input LUTs and one GCLK for the SDRAM controller. The controller's behavior is difficult to estimate in advance and thus its energy behavior because they are dependent on the spatial and temporal locality between the consecutive memory transactions. We may control the SDRAM in two ways. First, the controller sends the SDRAM in idle mode precharging the row right after a transaction. Secondly, the controller does not close the row and let the SDRAM remain in row-active mode after a transaction. These two methods affect both on performance and energy consumption [15]. All the memory cells must be refreshed every 64ms. Thus the energy consumption of the SDRAM controller cannot be accurately explained by average switching capacitance and signal transition activities. In addition, annotating an energy value per access may result in significant errors. Fig. 13 shows energy state machine of the SDRAM controller whose energy values are obtained by the in-house tool. Since this type of controller is operated mainly by the control signals, and the address and the data values generally bypass the controller, control-oriented characterization is superior to data-oriented characterization.

## 5. Conclusions

This paper introduces a practical approach to discover power and energy consumption of SRAM-based FPGAs, which is not really restricted to the SRAM-based FPGAs. Our approach is based on cycle-accurate measurement using switched capacitors. This method is ideal for the SRAM-based FPGAs because they are designed for synchronous operation. Furthermore, modern devices have separate power supply pins for the core, which helps avoid bad signal integrity due to power plane isolation for measurement. It makes possible to characterize not only the low-level physical implementation but also the high-level operation dependent energy consumption. We compare the capability of the tool with existing switching activity and switching capacitance based tool, XPower from Xilinx. We demonstrate featured energy measurement and characterization capability of the tool introducing high-level energy characterization of FPGAs with macro energy state machines.

## 6. References

- [1] S. Wenande and R. Chidester, "Xilinx takes power analysis to new levels with XPower," *Xcell Journal Online*, pp. 26 – 27, Fall – Winter 2001.
- [2] E. M. Sentovich, K. J. Singh, C. Moon, H. Savojo, R. K. Brayton, and A. Sangiovanni, "Sequential circuit design using synthesis and optimization," in *Proceedings for the IEEE International Conference on Computer Design*, pp. 328 – 333, October 1992.
- [3] B. Lin and H. D. Man, "Low-power driven technology mapping under timing constraints," in *IEEE International Conference on Computer Design*, pp. 421 – 427, October 1993.
- [4] C.-S. Chen, T. Hwang, and C. L. Liu, "Low power FPGA design - A re-engineering approach," in *Proceeding of DAC 1997*, pp. 656 – 661, June 1997.
- [5] V. George, H. Zhang, and J. Rabaey, "The design of a low energy FPGA," in *Proceedings of ISLPED 1999*, pp. 188 – 193, August 1999.
- [6] G. C. Cardarilli, A. D. Re, A. Nannarelli, and M. Re, "Power characterization of digital filters implemented on FPGA," in *Proceedings of the International Symposium on Circuit and System*, vol. 5, pp. 801 – 804, 2002.
- [7] Altera Corporation, "Power consumption comparison: APEX 20K vs. Virtex devices," *Altera Technical Brief*, October 1999.
- [8] T. Osmulski, J. T. Mudhring, B. Veale, J. M. West, H. Li, S. Vanichayobon, S.-H. Ko, J. K. Antonio, and S. K. Dhall, "A probabilistic power prediction tool for the Xilinx 4000-series FPGA," in *Proceedings of the 5th International Workshop on Embedded/Distributed HPC Systems and Application*, pp. 776 – 783, May 2000.
- [9] E. A. Kusse, "Analysis and circuit design for low power programmable logic modules," in *Master's thesis, Dept. of Electrical Engineering and Computer Science, University of California at Berkeley*, 1998.
- [10] B. Kumthekar, L. M. ii, and F. Somenzi, "Power optimization of FPGA-based designs without rewiring," in *Proceedings of IEE Computers and Digital Techniques*, vol. 147, pp. 167 – 174, May 2000.
- [11] Altera Corporation, "Evaluation power for Altera devices," *Altera Application Note*, July 2001.
- [12] N. Chang, K. Kim, and H. G. Lee, "Cycle-accurate energy consumption measurement and analysis: Case study of ARM7TDMI," *IEEE Transactions on VLSI Systems*, vol. 10, pp. 146 – 154, April 2002.
- [13] M. J. Alexander, "Power optimization for FPGA look-up tables," in *Proceedings of ISPD*, pp. 156 – 162, 1997.
- [14] K. Roy, "Power-dissipation driven FPGA place and route under timing constraints," *IEEE Transaction On Circuits And Systems*, vol. 46, pp. 634 – 637, May 1999.
- [15] Y. Joo, Y. S. Choi, H. Shim, H. G. Lee, K. Kim, and N. Chang, "Energy exploration and reduction of SDRAM memory systems," in *Proceedings of DAC 2002*, pp. 892 – 897, June 2002.