

Low-Power Integrated Circuit Design Optimization Approach

Y. A. Durrani¹

Electronics Engineering Department, UET, Taxila, Pakistan

¹yaseer.durrani@uettaxila.edu.pk

Abstract— In this tutorial survey, the paper presented a general review of the state-of-the-art techniques in optimizing the power dissipation on digital electronic systems. The source of power dissipation is focused on complementary metal-oxide-semiconductor (CMOS) circuits. This basic information cannot be implemented directly to optimize power dissipation due to the low abstraction level, but will be helpful to solve the power related problem. The major power factors are considered for the hardware and the software with the most trustful approaches of all levels of the design flow. The paper review is organized in three different types of digital system design: interpret conceptually, design flow, and management.

Keywords— Low-power, Power Optimization, Power Estimation, CAD/EDA Tools

I. INTRODUCTION

Nowadays, the new trends of technology scaling brought serious challenges in nano-scale technologies. The size of the transistors became smaller and faster, other important issues like power consumption, cost, error tolerance, verification, testing and integrity are the new challenges in digital system design. Among these issues, low-power very-large-scale-integrated (VLSI) design has become the most focused research area in fabrication processes and design techniques. In response to the increasing demand of electronic industry, power efficient electronic design automation (EDA) tools are highly desirable to handle low-power problems at all steps of the design techniques. These tools are the beginning to market products that helps with the minimum power dissipation of ICs. At the cost of performance, the reduction of the voltage offers the power minimization in the digital circuits. Typically supply-voltage is kept high to permit high speed of the clock and limits the deep sub-micron (DSM) effects. As compared to other type of transistors, CMOS transistors dissipate less energy during non-conducting state. Therefore, low-power design is nowadays focused to reduce the transition activity at the lowest level that requires to perform different tasks. The low-power integrated circuit (IC) design techniques can be used at every step of the design.

This paper is the guideline for IC designers for all

levels of abstraction and the low-power design flow. The section II of this paper demonstrates about the brief overview of the sources of power dissipation. This basic information cannot be implemented directly to optimize power dissipation due to the low abstraction level, but will be helpful to solve the power related problem. In section III gives the general review of the low-power design methodologies. Section IV explains about the power-reduction techniques at different levels of abstraction, which is the main part of this work. Finally, the section V is the conclusion. It is not possible to cover all related research in this specific area and we only discuss carefully selected topic in this paper.

II. SOURCES OF POWER DISSIPATION

The sources of power dissipation in CMOS circuit is comprised of dynamic and static power. Further, the dynamic power consists of capacitive switching power, short-circuit power and glitch power. The static power is divided into the leakage current and the dc standby power expressed in (1), (2), (3) and shown in Fig. 1.

$$P_{Total} = P_{Dynamic} + P_{Static} \quad (1)$$

where

$$P_{Dynamic} = \text{Capacitive Switching} + \text{Short Circuit Power} + \text{Glitch Power} \quad (2)$$

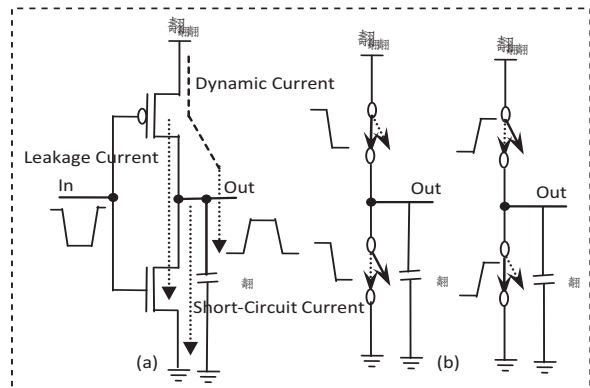


Fig. 1. (a) Sources of power dissipation in CMOS inverter (b) Charging/discharging the load capacitance C_L in equivalent circuit.

$$P_{Static} = \text{Leakage Current} + \text{DC standby Power} \quad (3)$$

In Fig. 2, the average power per clock period T_{clk} with current flow $I_{supply}(t)$ and supply voltage V_{DD} can be calculated as:

$$P_{avg} = \frac{1}{T_{clk}} \int_0^{T_{clk}} V_{DD} \cdot I_{supply}(t) dt \quad (4)$$

or power estimating by adding the total charge flow over each switching event through the circuit simulation.

A. Switching Power:

Switching power includes both short-circuit and dynamic power, resulting to charging and discharging of the output load capacitance C_L . It is observed that 50% of the energy is drawn as heat and other 50% of the energy dissipated from the power-supply that is stored in C_L . This output capacitance energy is released during the transition from the logical 0 to 1 or 1 to 0 as shown in Fig. 3. There are two activity factors involve in dynamic power:

- Factor 1: The fraction of clock periods when the output of the circuit has binary transition

$$P_{avg} = 0.5\alpha C_L V_{DD}^2 f_{clk} \quad (5)$$

- Factor 2: The fraction of clock periods during the output is switching from high to low or low to high

$$P_{avg} = \alpha_{off} C_L V_{DD}^2 f_{clk} \quad (6)$$

where V_{DD} is the supply voltage, C_L is the total load capacitance, f is the frequency and α is the factor of the switching activity. The C_L of the CMOS circuit consists of the input node capacitance, the output node capacitance of the driven gate and the effective capacitance of the interconnects/wires.

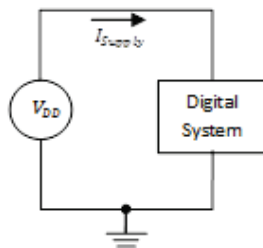


Fig. 2. Average power dissipation in digital system.

A. Short-Circuit Power:

Short-circuit power $P_{Short-Circuit}$ is dissipated due to the current path from V_{DD} to V_{SS} during NMOS and PMOS

transistors are active at same time for short instant of time, when the signal switching at the input to the logic circuits. It is stated in (7), (8):

$$P_{Short-Circuit} = \alpha_{0 \rightarrow 1} f_{clk} I_{peak} \left(\frac{t_r + t_f}{2} \right) V_{DD} \quad (7)$$

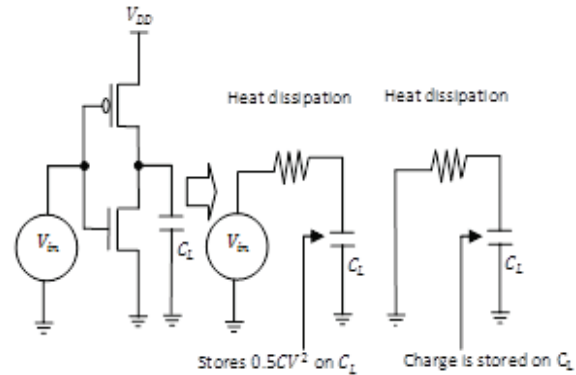


Fig 3. The switching activity factors involved in dynamic power.

$$P_{Short-Circuit} = K \frac{\beta}{12} (V_{DD} - 2V_T)^3 ft \quad (8)$$

where β is the CMOS transistor gain factor, V_T is the threshold voltage and τ is the rise and fall time of the logic gate input and K is the gain factor of the transistor. When K is zero then no power lost during the transition of a CMOS circuit.

$P_{Short-Circuit}$ can be minimize to a small amount of the overall power dissipation by using proper transistors sizing and reduction of the input rise-fall times in the circuit. Short-circuit power can also be reduced if the output rise-fall time of a gate is longer than the input rise-fall time. It consumes 10-20% of the overall power consumption. Normally, $P_{Short-Circuit}$ exists in the static CMOS logic families.

A. Dynamic glitch power:

Glitch or electronic pulse is an undesired switching activity for the short duration that occurs before the signal settles to its expected value. Glitches can cause sufficient amount of dynamic power dissipation due to the undesired transitions of the gate.

B. Leakage power:

Leakage power $P_{Leakage}$ is caused by the reverse-biased power built between the diffusion regions of the source-drain and substrate-well in the transistors. Due to several internal factors, the transistor continuous to dissipate $P_{Leakage}$ at all junctions, even-though if it is an active mode and not switching. In other words, $P_{Leakage}$ is due to sub-threshold and substrate injection factors at pn-junctions, which can be determined during fabrication process. It can be ignored, because it only contributes not more than 1% of the total power.

However, it is considerable in deep-submicron level. $P_{Leakage}$ is expressed in (9):

$$P_{Leakage} = (I_{Diode} + I_{Sub-Threshold})V_{DD} \quad (9)$$

where I_{Diode} is the current flows in the reverse-biased diodes that are build between substrate and diffusion regions as shown in Fig. 4. V_{DD} is the supply-voltage and $I_{Sub-Threshold}$ refers the current flows through the transistor. $P_{Leakage}$ is due to the following five components:

- Reverse-biased diode leakage current at the transistor domains
- Sub-threshold current through a turned-off transistor channel
- Gate induced drain leakage
- Punch through
- Gate oxide tunneling

However, these current components can be controlled technologically. Diode leakage current occurs, when one transistor is not conducting while other transistors are conducting and charges up/down the drain to its substrate potential. Fig. 4 demonstrates a PMOS transistor with a negative gate bias V_{DD} to its substrate. Hence, the diode formed by the drain diffusion and the substrate is reverse-biased. The reverse bias current in (10):

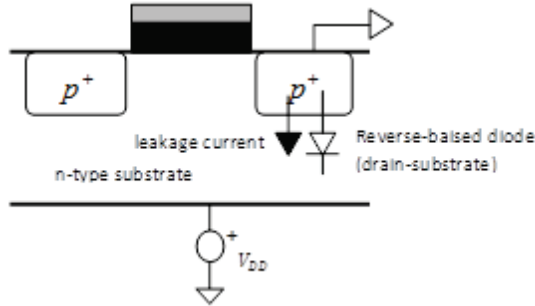


Fig. 4. The leakage current in a reverse-biased PMOS transistor.

$$I_{Leakage} = I_S \left(e^{\frac{V_{DD}}{V_T}} - 1 \right) \quad (10)$$

where $V_T = kT/q$ is the thermal voltage, V_{DD} is the biased voltage, and I_S refers to the reverse-saturation current.

Leakage power in DSM technologies is the most dominant factor in digital electronic circuits. The contribution of dynamic power dissipation also increases due to increased functionality requirements and clock frequencies. Consequently, the majority of existing low-power estimation techniques focus on this dynamic component of dissipation. Low-power design deals with the ability to minimize all sources of power

dissipation in CMOS logic gates. To control power delivery cost, operational voltage V_{DD} is reduced 30% of each technology generation. To sufficiently overdrive large gates, threshold voltage V_t can be reduced with the same rate. However, due to the reduction of V_t , transistor sub-threshold current exponentially increases. In case of V_{DD} and V_t , both reduced, the leakage power can be comparable to the dynamic power. Dynamic and leakage power increases the temperature. If heat is not dissipated effectively, temperature and $P_{Leakage}$ may cause physical damage in the CMOS circuits. Therefore, power optimization can be achieved by focusing several complex problems from different dimensions. Technology improvements can enhance the design capabilities and contributes to the reductions of all components, mostly impact on dynamic switching and short-circuit power.

III. LOW POWER DESIGN METHODOLOGIES

The major decisions for the low-power design starts with an understanding of the power consumption goals. The early strategy can severe effects the IC throughputs. These decisions can play important results in achieving the power specifications. With the changing environment, today's designers need a complete literate understanding to monitor and address the power at each level of the abstraction and obtain the maximum energy efficiency. As low-power has become major design parameter, the designers target on the sensible power designs where the power reduction approaches are applied. Nowadays System-on-Chip (SoC) requires a holistic and concurrent approach that includes the relationship among: system level design, architectural design, software/hardware co-design, intellectual property (IP) design, physical design, and Power verification.

The low-power design flow has several steps to perform designers and implement the digital electronic system. However, designers have to consider certain flow in the trajectory of the top-down technique, i.e. system to layout level, within each level there is no pre-defined design method. Every step includes several low-power design approaches. The low-power term includes all attempts that are possible to improve the logic circuits for power reduction. Power analysis follows power optimization. The accurate approaches must be adopted so that designers use proper power estimation function in the power reduction tools. The Low-power term can be sub-categories as:

- Power optimization and low-power synthesis
- Power estimation

A. Power optimization and low-power synthesis

In most cases electronic circuits are designed for small area and high speed, which are not the optimum circuit optimization. At all abstraction levels, the

dedicated low-power techniques and EDA tools must be developed to fully optimize the optimum implementation. The opportunities for power savings are the largest at system and architectural levels, whereas the accuracy is highest at the layout and gate levels as shown in Fig. 5. Modern low-power SoC requires concurrent and holistic techniques.

A. Power estimation

The computation and prediction of power dissipation is a challenging task, especially when the instantaneous

System Level	50...90%
Behaviour/Algorithm Level	40...70%
Architecture Level	30...50%
Logic Level	20...30%
Circuit Level	10...20%
Layout Level	5...10%

Fig. 5. Power reduction in low-power at top to bottom abstraction levels.

power is to be determined during the timing considerations.

Power estimation technique estimates the power consumption of electronic circuits. It is observed that power measurement approaches are more effective at high abstraction levels. The low-power estimation is interested in the optimization, analysis and trade-off of the circuit operation. Power does not only depends on typical power consumption factors (i.e. voltage, capacitance etc.), but it is also input patterns dependent. Furthermore, it is very difficult to determine an absolute estimate of the average power dissipation. More abstract variables such as correlation factors, transition density and signal probabilities are implemented in the power calculations. The main purpose of power measurements and analysis are to ensure the power consumption targets are achieved and not violated the power goals in the design of efficient low-power electronic system.

I. POWER OPTIMIZATION TECHNIQUES

A. System level

At system level the decisions making, the organizational and technological choices, the data representation, and the selection of algorithms for data processing have strong influence on the power dissipation. The power consumption in electronic system is scalable with the process technology scaling. System level optimization can be performed prior to the design of architecture that allows designers in short time to save all possible components of power. Several technological techniques have been adopted for the low-power based system design. Some of them are the

multiple input voltages on a single chip; low supply-voltage operations, clock speed, variable supply-voltage, adoptive voltage scaling, dynamic clock frequency, voltage scaling, and the static voltage scaling. In portable systems, frequency reduction can save sufficient amount of power in the processors. Some techniques reduce power for specific applications, where noise is not an important issue and can be tolerated sufficiently.

The choice of exact algorithm may provide the required functionality of the system under the design limitations. Algorithms used for the estimation of power dissipation are influenced by overall complexity, basic operations, and communication requirements etc. Algorithm selection is an important factor for the low-power design that can be made through an object-oriented approach. Data-type selection has major factor on the overall power dissipation. The choice of algorithm may neglect the effect of interaction between executing the remained computations and the hardware implementation function. Hence, power estimation of the pre-characterizing function in isolation mode can be inaccurate. This issue must be considered by approach for low-power designs. During each operation of execution, many algorithms have branches, procedure calls and loops that create the distribution which is not uniform. Such non-uniformity is important for the energy-efficient system design. Execution of algorithm flow under the input patterns/signals can easily be detected in the kernels of the computations. To optimize sufficient amount of power, each kernel must be reduced in the implementation on hardware design. Approximate consumption algorithm technique [i] can be adopted to improve the quality to power constraints and the requirements of the users.

Electronic system modeling is very useful to abstract system characteristics and design goals. Several data processing engines can communicate between each other through these characteristics. For example in multimedia application with different processing engines such as digital signal processor core, microcontroller core and the general-purpose core can execute directly in hardware, some with different instruction set architecture (ISA) and some executing parallel threads of processing. Therefore, macro-modelling architecture is suitable at high abstraction level for implementation of the models. Power efficient system design modelling must be appropriate by low-power design flow in all stages. Power dissipation in later stages of the design may be not very effective due to the important decisions on system-level have already been considered.

Software and hardware partitioning is a well-known design methodology with the goal to minimize the power consumption. A large amount of off-chip operations can be reduced by partitioning. During switching activities of the transistors, these off-chip

operations sufficiently contribute to the overall power consumption at the system level. When two concurrent processes are implemented with different hardware, we can turn-off the clock of inactive hardware and power saving power can be made by preventing the transition activity in the circuit. Through the efficient power management system, processor can keep an idle state during doze, nap, and sleep modes.

Normally, data influences the power dissipation of communication resources and data access in memory affects the energy usage, while software compilation affects the instruction sets in the computational blocks. On the other side, software does not have physical hardware realization and efficient hardware-based system design addresses the power optimization in storage, computational and the communication units.

Several approaches use aggressive methods to reduce power in applications for dealing with noise can be tolerated. Approximate computation algorithms [ii] can optimize a large amount of power but in the expense of inaccuracies in the computation. Such technique can handle quality of power parameters and the user requirements. For example in digital filters, input bit-width can be altered. The power consumption can be reduced by disabling fan-in and fan-out of inactive bit lines. In computation, minimization of bit widths can cause inaccuracy and quality of output is affected by noise. In some applications noise can be tolerated under different operating conditions.

Power-modelling approach has been developed to focus the entire system's power consumption such as operating system, processors, re-configurable circuits, and memories. These models are used in the library in the context of multi level design exploration: to deal with the energy/power optimization and estimations at system level.

Re-configurable processors are high enough efficient than general purpose processors. For example, the power efficiency of XPP processor is higher than ARM-9. However, such processor still lower than

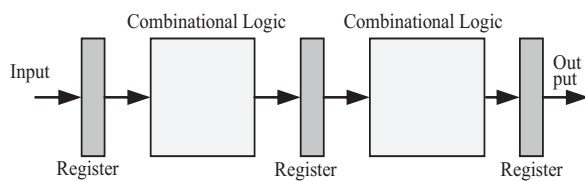


Fig. 6. A Synchronous Pipelined Model

application specific integrated circuit (ASIC) processing the same function due to run-time redundant logic circuit that is designed for the flexibility. Then the dual V_{DD} is applied to improve the power efficiency of reconfigurable processors. This approach is used to increase the slack time among several operations to minimize the power dissipation by executing the more fast operations with minimum voltage.

B. Architecture or Behavioral Level

For many electronic systems, the design entry point is architecture level. An accurate and clear power profile can be designed when the architecture is selected and register-transfer level (RTL) description is used for the specification of selected architecture. A huge variety of transformations from behavioural synthesis can be used such as unrolling, loop pipelining, control flow optimization and the design decisions have the dramatic effect on the requirements of power budget.

For improving performance parallelism is a well-known technique that optimizes power by reducing the supply voltage. A voltage reduction of up to 30-40% can be achieved depending on the data-path architecture. Architectural-based voltage scaling is the efficient technique at this level. It lowers the supply voltage V_{DD} and handles with the low circuit performance but compensate with a higher-throughput architecture. Parallelism technique compensates for the loss in throughput due to the voltage reduction.

Pipelining is another well-known technique for the power reduction. Adding the pipeline registers in the certain points of a data-path that decreases the critical path for allowing the data-path blocks to operate at the high rate as shown in Fig. 6. Increasing the level of pipelining may affect reducing the logic depth and power contribution because critical races are minimized. Simultaneously exploiting parallelism and pipelining approaches can obtain huge power optimization.

Today the designers have different alternatives to choose data representation such as: encoded vs. unencoded data, fixed-point vs. floating-point, sign-magnitude vs. two's complement methods. Each of these decisions involves a trade-off in accuracy between the power, the performance, and the simplicity of the design etc. The fixed-point representation has the minimum complexity that exhibits the low-power consumption but suffers the dynamic range difficulties. Software data scaling approach handles this problem; however, it must be incorporated into the processor micro-code. In contrast, floating-point alleviates the dynamic range difficulties at the expense of additional hardware. Two's complement method is widely used in arithmetic computations. In this type of representation, the most significant bit contains redundant information that causes additional switching activities and the high power dissipation. In contrast, signed data uses a single bit which also causes switching in one-bit of the data. Such problem can be solved through the data encoding, floating-point, and logarithmic companding techniques. In logarithmic scale many computations such as addition, the sign-magnitude do not have simple implementations; however, some arithmetic operations dissipate less power and become easier in the logarithmic scale: e.g. multiplication translates to additions. Applications such as large multiplications

can be helpful by using logarithmically encoded data.

Due to finite propagation delays several logic blocks can exhibit invalid transitions. Hence it is important that all signal paths have same propagation delays and reduce the logic-depth (through more cascading). Glitches are another reason of the additional switching activity in the logic circuits. Balanced circuits may have less glitches as the chain topology has been explained in [iii]. To increase the logic-depth may increase the capacitance due to glitches in the circuits and reduce the logic-depth may increase register power. So the selection of logic depth is the trade-off between the glitch capacitance and the register capacitance.

The selection of the hardware module for execution of a given instruction is an important aspect of the power optimization. For each different architecture of the same operation may have different area, performance and the power characterizations for example addition operation can be performed using ripple carry adder, carry save adder or carry look ahead adder [iv-v].

The global architectural technique for large memories and the driving signals across the chip for possible power dissipation tasks is the Global data transfer. Partitioning is another efficient technique for maximizing locality among different blocks. Exploiting locality through the distributed processors, controllers and the memories can improve sufficient power savings [vi].

C. Gate Level

Logic level has a huge impact on the power consumption, performance and area of its final gate level implementation. Several power optimizing techniques can be applied to the gate level design. Traditionally gate logic optimization can be divided into two steps: *technology-dependent* and *technology-independent* optimizations. Technology-dependent optimization can be made through the mapping of the library of a particular technology using technology-dependent algorithms for the power, performance, or area. Technology-independent optimization is the important factor for the logical functions. The commonly used algorithm for multi-level logic optimization is kernel extraction [vii]. From a given function kernels are extracted and the kernel with minimum literal counts is selected. For power dissipation, the main function is switching activity in the logic not the literal counts. The transition activity power can be optimized by Modified kernel extraction methods which are discussed in [vii]. The literal number of the factors represent in the logic gate function is effective technology-independent optimization. It may require an accurate power estimation of every network node to the mapped network power consumption.

In technology dependent approach, low-power technology mapping produces more accurate estimates

by accurately estimating the gate input capacitances and their impact on the power. These technology mapping techniques have been used in design tools used commercially. Several prototype mappers have been proposed and reported 10-15% average power savings [viii]. The adaptation of the mapping graph data structure to implicitly explore alternative logic decompositions during library bindings. This technique helps in finding better matching, at the price of increased computational effort for power estimation. A pre-computation technique [ix] decreases the power and increases the multiplication process speed of the logic gates. In this technique all multiplication steps related with the accumulation and the generation of the partial products are eliminated at the end of the multiplication method. Such method reduces the clock and transition activity of the gates.

Retiming is the well-known optimization technique in the synchronous sequential logic circuits. In this technique, the necessary clock period is minimized by repositions of the flip-flops [x]. The proposed algorithm for register retiming and minimum delay is Polynomial time-based algorithms [xi]. In synchronous logic circuits the transitions activity at the outputs of flip flops is less than the transitions activity at the flip-flop inputs. A large number of transitions activities are filtered out by the clock at the flip-flops inputs. The power consumption of sequential circuits is effectively reduced by retiming technique.

Boolean optimizations are more general, powerful, computationally intensive, and the propagation of don't cares are compared to the algebraic transformations. When especially targeting power, don't cares conditions are more complex, challenging issue and its simplification is a local problem. To restrict don't cares for optimization to a subset named power relevant don't cares [xii] that ensures the transitions activity in the fan-out of the optimized node does not increase. Power dissipation can be minimized through the logic restructuring techniques to prevent high switching frequencies from propagating the logic gates during their unwanted values.

Finite state machine (FSM) in logic level assigned codes that reduce the number of bits in the logic for similar state transitions [xiii]. State encoding technique namely weighted transition activity targets a power cost function [xiv]. If there are a large number of transitions between two states then the two states must be provided uni-distant codes to reduce the switching activity at the output of the flip-flops. Hence, the complexity of the logic circuit may increase the number of states that should not be avoided. Encode state transition graph technique [xv-xvi] produces the two-level and the multi-level implementations with the minimum power requirements. A method to re-encode sequential circuits at the logic level optimizes the power dissipation in [xvii]. Other techniques to encode to reduce the switching activities in the data-path logic

and minimize the switching on buses have been proposed in [xvii-xviii].

There are several power saving techniques that can be used both in dynamic and static logic circuits. Dynamic logic circuits can be more than 50% faster than static logic circuits. Static circuits are slower due to its twice loading capacitance, high threshold voltages, and the slow P-type transistors. Dynamic logic can be harder to design and may be the only choice to increase the processing speed. Most processors run at giga-hertz that requires dynamic circuits and some manufacturers for power reduction such as Intel processors completely switched to the static circuits. Power reduction not only extends the running time with the batteries, but it also reduces the thermal design requirements to minimize the size of the heat-sink, fans, etc. In dynamic logic, clock gating and asynchronous methods are more effective power optimization techniques. Circuit level design techniques that vary widely with delay and power dissipation, such as static CMOS, NP Domino or NORA logic, Cascode Voltage Switch Logic (CVSL), Push-Pull Pass Transistor Logic (PPL), Differential Cascode Voltage Switch Logic (DCVSL), Pseudo-NMOS, Complementary Pass Transistor Logic (CPL), Pseudo-NMOS etc. CMOS transistors are suitable for building low-power circuits however, CMOS gate suffer other three power components discussed in section II.

Low-power dynamic logic is useful for the reduction of switching activities, reduction of the short-circuit power consumption and minimization of the internal capacitances. In the preceding cycle, each time power is consumed during the pre-charge phase and output of the discharged capacitors [xix]. Some approaches developed two and four-phase clock strategies to handle the problem of CVSL [xx]. Some further modified techniques used basic domino logic methods e.g. NP-CMOS Domino and NORA logic [xxi-xxiii]. Power consumption in Single Rail Pass-Transistor Logic (SPL) circuit is sensitive to the low-voltage operations. The NMOS pass transistor logic has better low voltage performance than standard CMOS circuits. For low threshold voltages the sub-threshold leakage may become serious problem for both SPL and CMOS low power design circuits [xxiv]. In [xxiv], using the same conditions, the delay is worse and significantly greater for low supply voltages. Pass-transistor logic can achieve significant power reduction if the problem of threshold voltage drop and static power of the inverter output is properly addressed. CPL-based full adder implemented with transmission gates saves half of power compared to the standard CMOS adder [xxv]. Booth algorithm based multiplier saves 18% power and speed increases more than 30%.

PPL is another good choice for low-power design. For example 40-stage full adder is a power-delay product of 60%, CMOS multiplier of 42%, CPL of 63% and SPL of 78%. Generally Pseudo-NMOS is not

considered for low-power operation. This logic style can reduce power consumption only for complex logic function switching at high frequencies where savings to the dynamic power component due to reduced capacitances are dominant. DCVSL over conventional CMOS is faster switching due to reduced output capacitances. Compared to Pseudo-NMOS the DCVSL exceeds in that there is no static power consumption; however, the current during switching increases due to the large pull-up transistor. DCVSL logic style is suitable for implementing high fan-in gates and aims at reducing power consumption by limiting voltage swings in the internal nodes in the evaluation NMOS tree without any performance degradation. However, this style has some problems in different cases.

Domino circuit is normally used in high-speed applications. Domino circuit has faster speed but also have higher power dissipation. The low power dynamic logic gates works effectively at low supply voltages. The NMOS transistors are used instead of PMOS pull up transistors which reduce the voltage swing at the input of circuit.

Dynamic Latches are the simplest and most efficient timing circuits. There are different types of dynamic latch styles. According to simulation results in [xxvi], the non pre-charged true single phase clocked latch and 9T flip-flops dissipates lowest amount of power. The low-power dynamic latch can improve considerably their speed if they are designed without complementary outputs. A valuable comparison of the performance and power characteristics of flip-flops can be found in [xxvii].

D. Transistor Level

Transistor-level optimization and analysis in digital design is becoming critical for achieving possible unique solution of power, performance and area. Among several methods, transistor sizing is most common approach for various circuit optimization purposes. Most of the research work on transistor scaling has been on channel widths improvements of the transistor on the given circuit critical path delay. In [xxviii], the authors propose a novel method to incorporate into the transistor-level power optimization tools. The functional units are modeled by the graphical representation of the series parallel transistors. Then power budget are accurately allocate by using power budget distribution algorithm. The functional units' power consumption is optimized using these power budgets by transistor sizing technique.

In constant field scaling technique [xxix] a constant gate oxide field of the transistor is used to optimize the transistor geometric features and maintains the silicon doping level. This technique remains the power density constant and power dissipation scales as k^2 , while speed increases as k , Transistor size optimization can be performed with two types of algorithms:

- Algorithm that reduce the power dissipation by reducing the size of the gates and satisfies the timing constraint.
- Algorithm that performs sizing on each transistor for power optimization. The process may be completed, if power optimized layout satisfies the design constraints otherwise, power minimal size is applied to transistor sizes until the timing and delay constraints are required [xxx].

Advanced encryption standard (AES) technique [xxxii] is introduced in each stage of AES to optimize the low-power dissipation using multi threshold CMOS technique. The dynamic and sub-threshold power is reduced with different delays of the signals. The leakage current is minimized by implementing the most of the design part with low-power transistors with low speed. While the portion of design required high performance is implemented with faster transistors but high leakage current. This technique not only optimizes the leakage power but also allow propagation delay for the critical and non-critical paths closer. 10% power reduction can be made with AES technique of the system having throughput of higher than 18 GB/s.

Circuit parallelization is a well-known technique in which throughput of logic blocks on the critical path is maintain at a reduced voltage V_{DD} . It can be obtained with N number of blocks in parallel clocked at $\frac{f}{N}$. Each block can calculate its result in a given time slot N time longer and provided at a reduced V_{DD} shown in (11).

$$P = C \cdot \frac{f}{N} \cdot V_{DD}^2 \quad (11)$$

Power dissipation is minimizes without reducing the V_{DD} . But, some factors have to consider such as results of the logic blocks and logic block repetition.

The most commonly used circuit-level dynamic power optimization technique is clock gating. Switching activity in a block which is not used can be eliminated by stopping the clock signals at that block. Transistor-level multiple voltage usage in CMOS circuits produces significant amount of leakage power when the low voltage gates drive high voltage gates. In this case, High voltage gate's PMOS transistor is not turned off at low voltage produced by low voltage gates. To handle this problem, clustered voltage scaling (CVS) [xxxii-xxxiii] and Module level voltage scaling (MLVS) are proposed in [xxxiv-xxxv]. In CVS technique, low voltages are assigned to the gates to produce clusters. In MLVS approach, dual supply voltage assign to large blocks of the circuit. Both techniques limit the power optimization by introducing low voltage assignment process. For the placement of high and low voltage transistors in dual voltage circuits it is necessary to understand its effect on the leakage current. To solve this issue, voltage level circuits are constructed to handle the leakage problem [xxxvi]. Voltage level converters convert a low to high voltage

without increase of the leakage current. Further, CVS and MLVS techniques handle when no low voltage gate drives a high voltage gate. Both techniques consist of additional constraints to the dual voltage assignment process that reduces upto 8% of the total power dissipation in the circuit [xxxvi].

Reverse body bias (RBB) technique [xxxvii] increases the threshold voltages of transistors during an idle state. However, the performance of the RBB decreases as the V_{th} values are lowered or the channel lengths becomes smaller. 80% of the static power in CMOS circuits can be optimized by using multiple threshold voltages without decreasing the performance of the circuit [xxxviii]. A change of static logic provides a layout optimization for low-power by reducing the parasitic capacitances. Transistors are designed in series between output and the power supply. Nanotechnologies have been used with the limited number of standard cells for standard libraries. Such techniques achieve improved speed and low-power consumption compared to the conventional libraries [xxxix].

Several low-swing voltage technique have be introduced such as: conventional level converter (CLC), capacitive-coupled lever converter (CCLC), differential interconnect (DIFF), pseudo-differential interconnect (PDIFF), and pulse-controlled driver (PCD) etc. The most commonly used technique for power optimization on a long interconnect wire is low-swing voltage technique in which swing of the voltage is reduces on the wire. Power savings can be achieved with the minimum driver size due to the current delivered to the load capacitance by the driver in a certain time is smaller than the full-swing case. Second the charge required for charging and discharging of load capacitance is smaller. The dynamic energy E_{dyn} of interconnect wire in one cycle can explained in (12).

$$E_{D_{yn}} = \alpha \cdot C_L \cdot V_{Swing} \cdot V_{DD} \quad (12)$$

where V_{swing} is the voltage across the wire and α is the switching activity of the signal.

E. Layout/Physical Level

Layout or physical level is an intermediate between the gate level and the geometric design of the electronic system. There are numerous layout design styles are used to place, route, partition and resize transistors. In zero-delay or glitch free models, the transition activity of the transistors remains unchanged during layout optimization and hence power reduction can be made through netlist partitioning, transistor sizing, and transistor reordering, routing and gate placement. In with glitch models, designing layout are more complicated and glitch activity may effected by layout optimization in various ways so glitch activity cannot be modelled accurately.

A large number of low power layout optimization

methods have been developed such as wire and buffer resizing, re-mapping and local restructuring. The transistor re-sizing in layout minimizes the short-circuit power dissipation and the parasitic capacitances. The optimal size for the transistor that drives the output should be larger than the minimum size. In [xxxx], power-delay optimal sizes algorithm runs until the power-minimal layout satisfies the delay constraints. Several authors [xxxxi-xxxxii] demonstrated 15-20% power reduction can be obtained through the transistor re-sizing techniques. The relation between power dissipation, signal delay, interconnect load and driver size is described by optimal buffer size technique [xxxxiii]. A layout interconnection propagation delay is reduced significantly by wire sizing and driver sizing. Power dissipation increases when the size of wire increases because of increase in driver load. Interconnect delay can be improved by using wire and driver sizing but at the cost of small increase of power dissipation. Another technique [xxxxiv] which avoids the monotonicity of the propagation delay model by using an optimal gate and wire sizing on convex programming. Such approach can find the gate size and wire width.

In digital systems, clock distribution network dissipates significant amount of power. Several clock routing algorithm techniques have been proposed. Among those, wire sizing and wire elongation [xxxxv] proposed the chain of drivers at the zero-skew. The non-zero skew clock routing approach was introduced for sized to find a prescribed skew bound [xxxxvi]. In another technique [xxxxvii], buffers were introduced at internal points in the clock tree for satisfying the source-sink path delay constraints and minimizing the area of the clock network. Due to the buffer insertion method for the partition of large clock tree into small number of sub-trees with the minimum wire widths that results 60-70% power savings in the clock tree compared to the single driver approach. The low power clock routing minimizes the load on the clock drivers to meet a tolerable clock skew.

With small supply voltage the noise margins are diminished and the power distribution can have large effect on speed of the digital circuits. The wire sizing topology in power distribution networks (PDN) was proposed in [xxxxviii]. The main goal was to reduce the layout area while limiting the average current density to avoid the voltage drops and the electron-migration induction problems. This approach observes that when the two sinks do not draw currents at same time, narrow wires can be used for power distribution. Thus reduction can be made through layout area up to 30% compared to the star routing scheme.

V. CONCLUSION

Power has to be addressed in all levels of abstraction through the use of different EDA/CAD

tools. Based on the previous research, we discussed efficient techniques for the power optimization in digital electronic system. While system to layout level commercial tools is available but still many of the optimization techniques require manual interaction by the designer. For detailed discussions on different aspects of power consumption, the reader is referred to given below references.

REFERENCES

- [i] C. Yang, J. Chen, and T. Kuo, "An approximation algorithm for energy-efficient scheduling on a chip multiprocessor," In proc. of the Design, Automation and Test in Europe conference, vol. 1, pp. 468-473, March 2005.
- [ii] Y. A. Durrani, and T. Riesgo, "High-level power analysis for intellectual property-based digital systems," Springer Circuits, Systems & Signal Processing, vol. 33, no. 6, pp. 10531-1051, Dec. 2013.
- [iii] C. Nagendra, R. Owens, and M. Irwin, "Power-delay characteristics of CMOS adders," IEEE Trans. VLSI systems, vol.2, pp. 377-381, Sept. 1994.
- [v] A. Bellaouar, and M. Elmasry, "Low-Power Digital VLSI Design Circuits and Systems", Kluwer Academic Publishers, 1995.
- [iv] Y. A. Durrani, and T. Riesgo, "High-level power Analysis for IP-based digital systems," Journal of Low Power Electronics, American Scientific Publisher, vol. 9, no. 4, pp. 435-444, Dec. 2013.
- [vi] Y. A. Durrani, and T. Riesgo, "Power estimation for intellectual property-based digital systems at architectural level" Elsevier Journal of King Saud University-Computer and Information Sciences, vol. 26, no. 3, pp. 1319-1578, 2014.
- [vii] K. Roy and S. Prasad, "SYCLOP: Synthesis of CMOS Logic for Low Power Applications" In proc. of the Int'l Conference on Computer Design: VLSI in Computer and Processors, pp. 464-467, Oct. 1992.
- [viii] S. Iman, and M. Pedram, "Logic Synthesis for Low Power VLSI Designs," Kluwer Academic Publishers, 1998.
- [ix] Y. Liu, and S. Furber, "The design of a low power asynchronous multiplier," in Proc. ISLPED, pp. 301-306, 2004.
- [x] L.-F. Chao and E. H.-M. Sha, "Scheduling data-flow graphs via retiming and unfolding," IEEE Transactions on Parallel and Distributed Systems, vol.8, pp.1259-1267, 1997.
- [xi] T. Soyata, E. Friedman, and J. Mulligan, "Incorporating interconnect, register, and clock distribution delays into the retiming process," IEEE Transaction on CAD of

- integrated circuits and systems, vol.16, no.1, pp. 105-120, Jan. 1997.
- [xii] M. Pedram, "Power Estimation and Optimization at the Logic Level," *Intl. Journal of High-Speed Electronics and Systems*, vol.5, no.2, pp. 179-202, 1994.
- [xiii] C-Y. Tsui, M. Pedram, and C-h. Chen, A. Despain, "Low power state assignment targeting two- and multi-level logic implementations," In proc. of the IEEE International Conference on Computer Aided Design, pp. 82-87, Nov. 1994.
- [xiv] P. Ashar, S. Devdas, and A. R. Newton, "Sequential Logic Synthesis," Kluwer Academic Publishers, Boston, 1991.
- [xv] K. Roy, and S. Prasad, "SYCLOP: Synthesis of CMOS Logic for Low Power Applications," In proc. of the Int'l Conference on Computer Design: VLSI in Computer and Processors, pp. 464-467, Oct. 1992.
- [xvi] H-S. Jung, and M. Pedram "Uncertainty-aware dynamic power management in partially observable domains," *IEEE Trans. on VLSI Systems*, vol. 17, no. 7, pp. 929-942, Jul. 2009.
- [xvii] G. D Hachtel, M. Hermida, A. Pardo, M. Poncino, and F. Somenzi, "Re-Encoding Sequential Circuits to Reduce Power Dissipation," In proc. of the Int'l Conference on Computer Aided Design, pp. 70-73, Nov. 1994.
- [xviii] M. Stan and W. Burleson. "Limited-weight codes for low power I/O," In proc. of Int'l Workshop on Low Power Design, pp. 209-214, April 1994.
- [xix] J. M. Rabaey, "Digital Integrated Circuits: A Design Perspective", Prentice Hall Publisher, 1996.
- [xx] N. H: E. Weste, and K. Eshraghain, "Principles of CMOS VLSI Design," Addison Wesley Publisher, 1994.
- [xxi] C. M. Lee, and E. W. Szeto, "Zipper CMOS," *IEEE Circuits and Systems Magazine*, pp. 10-16, May 1986.
- [xxii] R. Krambeck et al., "High-Speed Compact Circuits with CMOS," *IEEE Journal of Solid State Circuits*, vol. SC-17, no.3, pp. 614-619, June 1982.
- [xxiii] N. Gonclaves, and H. J. DeMan, "NORA: a race free dynamic CMOS technique for pipelined logic structures," *IEEE Journal of Solid State Circuits*, vol. SC-118, no. 3, pp. 261-266, June 1983.
- [xxiv] K. Yano, Y. Sasaki, K. Rikino, and K., Seki, "Top-Down Pass-Transistor Logic Design," *IEEE Journal of Solid State Circuits*, vol. 31, no. 10, pp. 792-803, Oct. 1996.
- [xxv] I. S. Abu-Khater, A. Bellaouar, and M.I. Elmasry, "Circuit Techniques for CMOS Low-Power High-Performance Multipliers," *IEEE Journal of Solid-State Circuits*, vol. 31, no. 10, pp. 1535-1546, October 1996.
- [xxvi] C. Svensson, and J. Yuan, "Latches and Flip-Flops for Low Power Systems," *Low Power CMOS Design*, IEEE Press, 1998.
- [xxvii] V. Stojanovic, and V. G. Oklobdzija, "Comparative Analysis of Master-Slave Latches and Flip Fops," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 4, pp. 536-548, April 1999.
- [xxviii] V. Sundararajan, S. S. Sapatnekar, and K. K. Parhi, "MINFLOTRANSIT: Min-Cost Flow Based Transistor Sizing Tool," *Design Automation Conference*, pp. 453-460, 2000.
- [xxix] S. Borkar, "Design Challenges of Technology Scaling," *IEEE Micro*, vol. 19, no.4, pp. 23-29, July 1999.
- [xxx] M. Borah, R. M. Owens, and M. J. Irwin, "Transistor sizing for low power CMOS circuits," *IEEE Transactions on Computer Aided Computer-Aided Design of Integrated Circuits and Systems*, vol. 15, no. 6, pp. 665-671, June 1996.
- [xxxi] A. Alma'aitah, and Z. Abid, "Area efficient high throughput sub-pipelined design of the AES in CMOS 180nm," *5th International Design and Test Workshop*, pp.31-36, 2010.
- [xxxii] K. Usami, and M. Horowitz, "Clustered voltage scaling technique for low-power design," *Low Power Design Symposium*, pp. 3-8, 1995.
- [xxxiii] M. Donno, L. Macchiarulo, A. Macii, E. Macii, and M. Poncino, "Enhanced clustered voltage scaling for low power," *GLSVLSI'02*. In proc. of the 12th ACM Great Lakes Symposium on VLSI, 18-20, pp. 18-23, April 2002.
- [xxxiv] J.-M. Chang, and M. Pedram, "Energy minimization using multiple supply voltages," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 5, pp. 436-443, 1997.
- [xxxv] C. Chen, A. Srivastava, and M. Sarrafzadeh, "On gate level power optimization using dual-supply voltages," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 9, pp. 616-29, 2001.
- [xxxvi] S. H. Kulkarni, and D. Sylvester, "Fast and Energy-Efficient Asynchronous Level Converters for Multi-VDD Design," *IEEE International SoC Conference*, pp. 169-172, 2003.
- [xxxvii] G. Sery, S. Borkar, and V. De, "Life is CMOS: why chase the life after," In proc. of *Design Automation Conference*, pp. 78-83, 2002.

- [xxxviii] L. Wei, Z. Chen, K. Roy, M. C. Johnson, Y. Ye, and V. K. De, "Design and optimization of dual-threshold circuits for low-voltage low-power applications," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 7, pp. 16-24, 1999.
- [xxxix] J. M. Masgonty, S. Cserveny, C. Arm, P.-D. P. fister, and C. Piguet. "Low-Power Low-Voltage Standard Cell Libraries with a Limited Number of Cells," In proc. of 11th Int. Workshop on Power and Timing Modeling, Optimization and Simulation, Sept. 2001.
- [xxxx] M. Borah, R. M. Owens, and M. J. Irwin. "Transistor sizing for minimizing power consumption of CMOS circuits under delay constraint," In proc. of the 1995 International Symposium on Low Power Design, pp. 167-172, April 1995.
- [xxxxi] H-R. Lin, and T-T. Hwang, "Power reduction by gate sizing with path-oriented slack calculation," In proc. of the 1st Asia-Pacific Design Automation Conference, pp. 7-12, Aug. 1995.
- [xxxvii] S. Sapatnekar and W. Chuang "Power versus delay in gate sizing: conflicting objectives," In proc. of the IEEE International Conference on Computer Aided Design, pp. 323-330, Nov. 1995.
- [xxxviii] A. Vittal, and M. Marek-Sadowska, "Power optimal buffered clock tree design," In proc. of the 32nd Design Automation Conference, pp. 497-502, June 1995.
- [xxxix] N. Menezes, R. Baldick, and L. T. Pileggi, "A sequential quadratic programming approach to concurrent gate and wire sizing," In proc. of the IEEE International Conference on Computer Aided Design, pp. 144-151, Nov. 1995.
- [xxxv] J. Lillis, C-K Cheng, and T-T. Y. Lin, "Optimal wire sizing and buffer insertion for low power and a generalized delay model" In proc. of the International Conference on Computer Design, pp. 138-143, Nov. 1995.
- [xxxvi] S. Pullela, N. Menezes, and L. T. Pillage, "Reliable non-zero skew clock tree using wire width minimization," In proc. of the 30th Design Automation Conference, pp. 165-170, June 1993.
- [xxxvii] J. G. Xi, and W-M. Dai, "Buffer insertion and sizing under process variations for low power," In proc. of the 32nd Design Automation Conference, pp. 491-496, June 1995.
- [xxxviii] A. Vittal, and M. Marek-Sadowska, "Power distribution topology design," In proc. of the 32nd Design Automation Conference, pp. 503-507, June 1995.