

An Efficient and Effective Methodology to Control Turn-On Sequence of Power Switches for Power Gating Designs

Ya-Ting Shyu, Jai-Ming Lin, *Member, IEEE*, Che-Chun Lin, Chun-Po Huang,
and Soon-Jyh Chang, *Member, IEEE*

Abstract—As technology advances, power consumption becomes a big challenge in modern very large-scale integration designs. To resolve this problem, power-gated technology has been widely adopted in circuit designs. Since the turn-on sequence of power switches has a great impact on the rush current, wake-up, and sequence times of a power gating design, this paper proposes a methodology to construct a hybrid routing structure to connect power switches. Our hybrid routing structure can induce less rush current and satisfy timing constraints because a better daisy chain is constructed. To find members of the daisy chain, an integer linear programming algorithm is used to pick up suitable power switches. To determine suitable depth of a daisy chain, we propose a model for a power gating design and induct precise equations to estimate voltage and rush current equations according to the model. All of our experiments are based on industrial designs and measured by vendor tools. The experimental results demonstrate the efficiency and effectiveness of our design methodology.

Index Terms—Hybrid routing structure, low-power design, power gating, rush current.

I. INTRODUCTION

DUE TO prevalence of battery-powered portable devices, energy efficiency is a great concern in modern very large-scale integration designs. In CMOS technology, the power consumption of a design usually comes from dynamic power and is proportional to the frequencies and activities of the circuits. When chip performance increases, it will consume more dynamic power. However, according to the shrinking of feature size, static power consumption has gained much attention recently because of the significant increase of leakage current. To resolve this problem, many approaches have been developed including power gating, body biasing, dynamic voltage scaling, clock gating, etc.

Manuscript received February 13, 2015; revised August 7, 2015 and November 5, 2015; accepted January 7, 2016. Date of publication February 8, 2016; date of current version September 7, 2016. This work was supported by the National Science Council of Taiwan under Grant 102-2221-E-006-263-MY3. This paper was recommended by Associate Editor A. Srivastava.

Y.-T. Shyu, J.-M. Lin, C.-P. Huang, and S.-J. Chang are with the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan (e-mail: kkttkkk@ssc.as.ncku.edu.tw; jmlin@ee.ncku.edu.tw; gppo@ssc.as.ncku.edu.tw; soon@ee.ncku.edu.tw).

C.-C. Lin is with Himax Technologies, Tainan, Taiwan (e-mail: harvey3810996@hotmail.com).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCAD.2016.2523916

Power-gating is one of the most effective methods in reducing leakage power. To achieve high performance without creating serious leakage power, circuit functions are implemented using low-Vt transistors while high-Vt transistors are used to implement always-on circuits such as power switches [10], [11], [16]. This architecture contains two kinds of power meshes, which are the global mesh (named global VDD) and local mesh (named virtual VDD). The power lines of the low-Vt cells are connected together by virtual VDD, and the circuit functions using low-Vt devices are considered as a power-gated domain. The global and local meshes are connected together by power switches. By operating these power switches, the power supply of a power-gated domain cut off in idle mode and resume the power supply when in the active mode. The activities of the power switches are controlled by the control logic.

The power gating design can be implemented by two kinds of structures, one is the fine-grain structure and the other is the coarse-grain structure. For the fine-grain structure in [17], an independent sleep transistor is provided to each gate in the circuit for power gating. In [8], all logical cells are separated into different clusters and the power in each cluster can be turned down separately by controlling the respective power switch. The fine-grain structure needs to be designed more carefully because it has to divide a design into several clusters according to their geometrical locations and insert a logic control circuit into each cluster to control a power switch. Thus, [2] proposes a distributed sleep transistor network to handle a power gating design, which is also considered a coarse-grain structure. All logical cells in a power-gated domain are connected together by one power network and several power switches are placed into the power network for power gating (see Fig. 1). Because this structure can be implemented more easily, coarse-grain structure is supported by current commercial tools.

Fig. 1(a) shows the schematic of the coarse-grain structure. There are two kinds of power meshes: 1) global VDD and 2) virtual VDD. These two power meshes are connected together by a set of power switches. The power-gated logic represents the circuit in a power-gated domain, and the logic cells in the power-gated domain are connected to the mesh of the virtual VDD. They can be powered down by turning off the power switches. Then, the leakage power in the power-gated logic is reduced. Fig. 1(b) shows a power gating design in 3-D view.

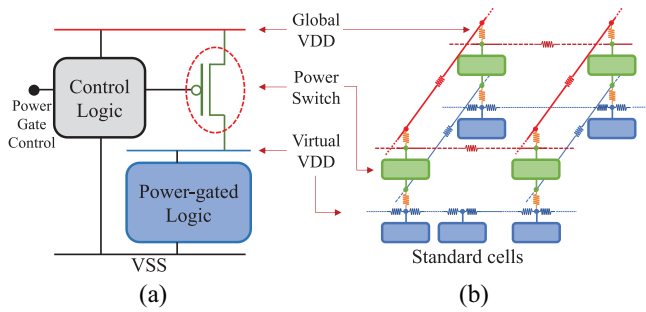


Fig. 1. (a) Schematic of a power gating design. (b) Power gating design in 3-D view.

At the moment the power switches are turned on, transient current will pass through power switches and flow into a power-gated domain. This is called rush current. Huge rush current may lead to large IR-drop in nearby always-on blocks and reduce chip performance. It even may lead to functional errors if the drop value is too large. Therefore, reducing maximum rush current is an important concern in power gating design.

While power switches are turned on in sequence, it takes some time before the virtual VDD can be charged to a stable voltage value. This duration is labeled the wake-up time. Before the power-gated domain is activated, all power switches have to be turned on. This time is considered the sequence time. Wake-up and sequence times are affected by the power switch driving capability and the capacitance loaded in the power mesh of the virtual VDD. The times can be decreased if the driving strength of a power switch is increased. However, better power driving capability may induce large rush current. Hence, there exists a tradeoff between rush current and wake-up (and sequence) time in power gating design. If a power-gated domain is turned on too fast, it may induce large rush current. However, circuit performance is degraded, if it is turned on too slow. Thus, a proper turn-on sequence of power switches will not violate the maximum rush current constraint and also satisfy the wake-up and sequence time constraints.

A. Previous Work

The performance of the power gating design is greatly impacted by the placement and connection sequence of the power switches. The research of power-gated technology can be divided into two major categories according to the active and ramp-up working modes. In the active mode, the normal functions need to work correctly. Thus, the related research [5], [8], [9], [17] focus on inserting the proper number of power switches at suitable locations to satisfy design constraints. Research about power switch signal network synthesis and delay buffer insertion were proposed by [3], [6]–[8], [12]–[14], and [17]–[19]. Some papers research the structure of customized power switches [2], [8], [14], and other papers discuss the custom power switch turn-on scheduling [1], [4], [13], [19]. The methodologies of multiphase power switch turn-on chain routing were proposed in [13] and [15].

The major concern in ramp-up mode is how to reduce rush current and satisfy wake-up and sequence time constraints, which are mainly impacted by the power switch turn-on

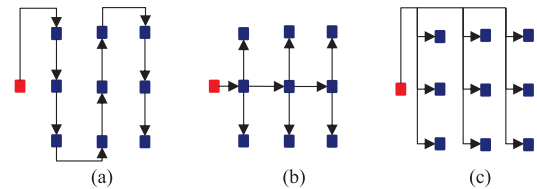


Fig. 2. Three kinds of routing structures to connect power switches. (a) Daisy chain. (b) Fishbone. (c) High fan-out.

sequence. Recently, Chen *et al.* [13] proposed a framework to schedule the power-up sequence of the power switches. This approach minimizes power ramp-up time while limiting the inrush current. However, Chen *et al.* [13] used a look-up table to describe the behavior of a power switch. Since the look-up table is noncontinuous, it cannot be used in mathematic analysis directly. To facilitate analysis, they assume that current flowing through a power switch is constant during a time interval.

B. Our Contribution

This paper proposes a methodology to construct a hybrid routing structure to connect power switches. Our hybrid routing structure can alleviate rush current without violating wake-up (sequence) time constraint because of the following reasons.

- 1) We propose a simplified model for the power gating design and induct equations to estimate voltage and transient current according to the model. In order to get accurate values, we introduce the power switch transient coefficient and power switch on-resistance variance coefficient into the equations.
- 2) The daisy chain determines the quality of the hybrid routing structure. A delicate procedure is used to construct a better daisy chain.
 - a) An integer linear programming (ILP) algorithm is used to pick up suitable power switches to constitute the daisy chain.
 - b) The proper depth of a daisy chain can be determined by the associated equations.
- 3) To enhance performance and avoid design rule check violations, we use the distributed fishbone structure to turn on the power switches which are not used in a daisy chain.
- 4) The experimental results demonstrate that our approach obtains better results of rush current and sequence time than other routing structures in real designs.

II. HYBRID ROUTING METHODOLOGY

The turn-on sequence of power switches has great impact on rush current and wake-up (and sequence) time in power gating designs, and there exists a tradeoff between these factors. In order to consider them at the same time, this paper uses a hybrid routing structure to connect power switches.

Three kinds of basic routing structures are commonly used to connect the control signals of power switches, which are daisy chain, fishbone, and high fan-out structures (Fig. 2). Each routing structure has its advantages and disadvantages. Among these routing structures, the daisy chain structure is

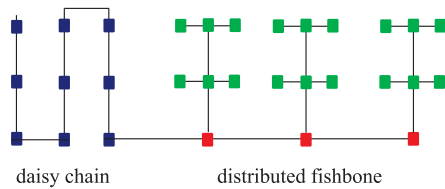


Fig. 3. Hybrid routing structure, which includes the daisy chain and distributed fishbone structures.

most effective in mitigating rush current. However, it results in longer wake-up and sequence times, which degrade chip performance. On the contrary, fishbone and high fan-out structures produce shorter wake-up and sequence times, but have larger rush current which may cause serious IR-drop.

This paper uses a hybrid routing structure to connect power switches, which starts with a daisy chain followed by a distributed fishbone structure. See Fig. 3 for our hybrid routing structure. Since the daisy chain structure turns on power switches in sequence, it induces smaller rush current than the other two structures. Once virtual VDD is charged close to global VDD by the daisy chain, it does not induce large rush current even though the remaining power switches are turned on quickly. Hence, the remaining power switches are connected by the distributed fishbone structure to reduce wake-up and sequence times.

The most difficult part in building a hybrid routing structure is determining the suitable number of power switches in a daisy chain which is considered the depth of the daisy chain. Increasing the depth of a daisy chain leads to longer wake-up and sequence times. On the contrary, it may result in the large rush current if the remaining power switches are quickly turned on by the distributed fishbone. Therefore, the major contribution of this paper is to propose an efficient and effective procedure to determine the precise depth of a daisy chain. Thus, power gating designs using our hybrid routing structures can satisfy the timing constraints without inducing serious rush current.

III. MODEL AND EQUATIONS FOR DAISY CHAIN

This section first proposes a simplified model for power gating designs. The model enables us to induct equations to predict transient voltage and rush current in a virtual VDD. Based on the equations, we can propose an efficient and effective procedure to predict a suitable number of power switches for a daisy chain without circuit level simulation.

A. C1C2 Model With Leakage Resistance

To compute rush current efficiently for a power gated circuit, this section introduces the C1C2 model with leakage resistance derived from the C1RC2 model with leakage resistance [16] to simplify circuit structure. The power network in the coarse grain structure consists of two layers of power meshes, which are interconnected by a set of power switches. The top layer power mesh obtains power from power pads and the bottom layer power mesh provides power for standard cells as shown in Fig. 4(a), where VDD, VDD_OFF, and VSS denote the global VDD, the virtual VDD, and ground, respectively. Fig. 4(b) shows the power network of Fig. 4(a) in 3-D.

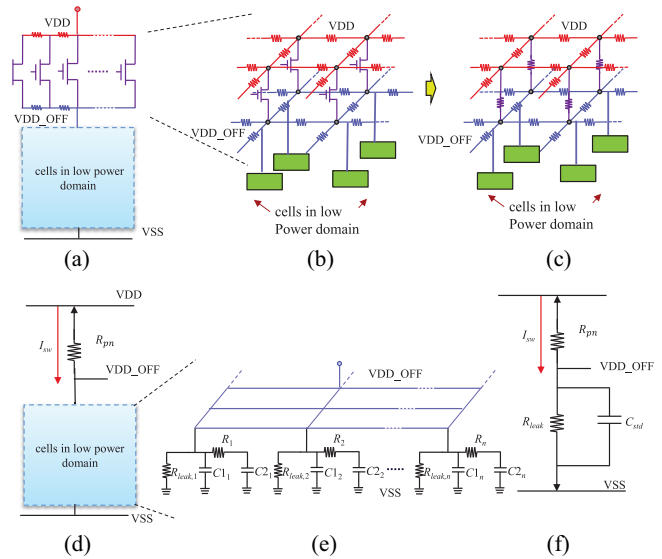


Fig. 4. Our model for a coarse-grain structure. (a) Coarse-grain structure. (b) Power network in 3-D view. (c) Resistance network of the power network. (d) Simplified coarse-grain structure from replacing the power network with the associated equivalent resistance. (e) C1RC2 model with leakage power for standard cells. (f) C1C2 model with leakage resistance.

Since a power switch behaves like a resistor during ramp-up, the whole power network can be transformed into a 3-D resistor network like Fig. 4(c). If the equivalent resistance of the resistor network can be found, all standard cells, which originally connected to the bottom layer power mesh, are connected to one terminal of the resistor in parallel. Since the wire resistance is relative smaller than the equivalent resistance of power switches and standard cells, for simplification, we assume that the wire resistance is 0Ω . After transferring the wire resistor into a short circuit, the structure in Fig. 4(c) can be simplified into the structure in Fig. 4(d). Fig. 4(d) shows the simplified structure of Fig. 4(a), where R_{pn} denotes the equivalent parallel resistance of all power switches.

A power gating design may contain millions of standard cells. Due to capacity limit and runtime considerations, it is impossible to simulate the associated netlist at the circuit level. Hence, Sreekumar and Ravichandran [16] proposed to use the C1RC2 model with leakage resistance R_{leak} to represent each standard cell in ramp-up mode. The C1RC2 model contains two parasitic capacitances (denoted by $C1$ and $C2$) and one internal resistance (denoted by R), where $C1$ connects to the power terminal and $C2$ connects to the output terminal. Let $C1_i$, R_i , $C2_i$, and $R_{leak,i}$ represent the components of the i th standard cell in the C1RC2 model with leakage resistance. Fig. 4(e) shows the resulting RC network if all cells in a low-power domain of Fig. 4(d) are replaced by the C1RC2 model.

The transient behavior of a circuit is determined by its dominant pole according to the theory of frequency response. The charge and discharge speed of capacitors in an RC circuit depend on the RC time constant in the time domain. Let τ_{domint} denote the RC time constant. τ_{domint} is obtained by summing the RC-products for each standard cell in the circuit. Because R_i is far less than $R_{leak,i}$, τ_{domint} can be approximated by $(R_{pn} \parallel R_{leak}) \times (\sum_{i=1}^n (C1_i + C2_i))$, where n is the number

of cells and R_{leak} is the equivalent parallel resistance of all the $R_{\text{leak},i}$. The resistance R can be discarded from the C1RC2 model, and the resulting model is termed the C1C2 model with leakage resistance.

Based on C1C2 model with leakage resistance, all standard cells are represented by a set of capacitances and resistances parallel connecting to one terminal of R_{pn} . Let C_{std} and R_{leak} , respectively, denote the capacitance and leakage resistance induced by the standard cells. C_{std} is equal to the summation of the $C1_i$ and $C2_i$ of the cells [i.e., $C_{\text{std}} = \sum_{i=1}^n (C1_i + C2_i)$] while R_{leak} is calculated by the equation $\text{VDD}^2/P_{\text{leak}}$, where VDD denotes the supply voltage value of the global VDD and P_{leak} denotes the total leakage power of the cells. Based on the above description, an RC network power gating design is shown in Fig. 4(f).

Because the value of R_{pn} varies according to the on/off condition of each power switch, the turn-on sequence of power switches has a great influence on rush current and wake-up time.

B. Equations for Estimating Voltage

Based on our model, this section deduces equations to estimate transient voltage value and rush current in power gated designs. Given a set of power switches which are in a stable power-off state, we first deduce an equation to compute the transient voltage of a power switch after it is turned on. Next, according to this equation, more general equations which can estimate the transient voltage and current of power switches in a daisy chain are proposed. It is assumed that each power switch, denoted by s_i , is triggered by the ON signal at time T_{s_i} and $T_{s_i} < T_{s_{i+1}}$.

Let $\text{VDD_OFF}(t)$ denote the voltage of virtual VDD at time t , R_{pn} denotes the equivalent resistance of the power switch network, and VDD (VSS) denotes the voltage of global VDD (ground). The first power switch s_1 is turned on at time T_{s_1} . When $t < T_{s_1}$, $\text{VDD_OFF}(t)$ is equal to VSS . When $t \geq T_{s_1}$, according to our model and Kirchhoff's current law (KCL), $\text{VDD_OFF}(t)$ must satisfy the following equation:

$$\frac{\text{VDD} - \text{VDD_OFF}(t)}{R_{\text{pn}}} = C_{\text{std}} \times \frac{d(\text{VDD_OFF}(t) - \text{VSS})}{dt} + \frac{\text{VDD_OFF}(t) - \text{VSS}}{R_{\text{leak}}} \quad (1)$$

where C_{std} and R_{leak} , respectively, denote the parasitic capacitance and leakage resistance of standard cells. Since (1) is a noncontinuous first order differential equation, it can be solved by the standard differential equation solving method. The particular solutions are shown in the following (see the Appendix for the detailed derivation process):

$$\text{VDD_OFF}(t) = V_{\text{final}} + (V_{\text{init}} - V_{\text{final}}) \times e^{\frac{-(t-T_p)}{R_{\text{total}} \times C_{\text{std}}}} \quad (2)$$

where V_{init} and V_{final} denote the voltages of virtual VDD node just before s_1 is turned on and after s_1 is turned on for a long period (i.e., at the moment, s_1 acts as resistor and capacitor C_{std} acts as an open circuit), respectively. T_p denotes the time when s_1 is turned on, and R_{total} denotes the equivalent parallel

resistance of the resistors link to virtual VDD node. The values V_{init} , V_{final} , T_p , and R_{total} in (2) are shown in the following:

$$\begin{cases} V_{\text{init}} = \text{VSS} \\ V_{\text{final}} = \text{VDD} \times \frac{R_{\text{leak}}}{R_{\text{leak}} + R_{\text{pn}}} + \text{VSS} \times \frac{R_{\text{pn}}}{R_{\text{leak}} + R_{\text{pn}}} \\ R_{\text{total}} = R_{\text{pn}} \parallel R_{\text{leak}} \\ T_p = T_{s_1} \end{cases} \quad (3)$$

Since only s_1 acts as the resistor and the other power switches act as open circuits after T_p , the value of R_{pn} is equal to $R_{s_1,\text{on}}$, where $R_{s_1,\text{on}}$ denotes the on-resistance of s_1 . This value can be obtained from library.

Based on (2), the transient voltage of the virtual VDD node after a new power switch s_i is turned on in the daisy chain can be computed. Let $\text{VDD_OFF}(t, i)$ denote the transient voltage of the virtual VDD during time period $[T_{s_i}, T_{s_{i+1}})$ (after s_i is turned on and before s_{i+1} is turned on), and it can be expressed as follows:

$$\text{VDD_OFF}(t, i) = V_{\text{final}}(i) + (V_{\text{init}}(i) - V_{\text{final}}(i)) \times e^{\frac{-(t-T_p(i))}{R_{\text{total}}(i) \times C_{\text{std}}}} \quad (4)$$

where $V_{\text{init}}(i)$, $V_{\text{final}}(i)$, $T_p(i)$, and $R_{\text{total}}(i)$ are the following:

$$\begin{cases} V_{\text{init}}(i) = \text{VDD_OFF}(T_{s_i}, i - 1) \\ V_{\text{final}}(i) = \text{VDD} \times \frac{R_{\text{leak}}}{R_{\text{leak}} + R_{\text{pn}}(i)} + \text{VSS} \times \frac{R_{\text{pn}}(i)}{R_{\text{leak}} + R_{\text{pn}}(i)} \\ R_{\text{total}}(i) = R_{\text{pn}}(i) \parallel R_{\text{leak}} \\ T_p(i) = T_{s_i} \end{cases} \quad (5)$$

$V_{\text{init}}(i)$ denotes the initial transient voltage value of the period $[T_{s_i}, T_{s_{i+1}})$, which is equal to the final transient voltage value of the previous period $[T_{s_{i-1}}, T_{s_i})$. The major difference between (3) and (5) lies with R_{pn} and $R_{\text{pn}}(i)$. After a new power switch s_i is turned during the next time period $[T_{s_i}, T_{s_{i+1}})$, there exist i power switches (i.e., s_1 to s_i) in the stable power-on state. Let $R_{\text{pn}}(i)$ denote the equivalent resistance of the power network at period $[T_{s_i}, T_{s_{i+1}})$. $R_{\text{pn}}(i)$ is equivalent to the resistance of parallel power switches from s_1 to s_i according to our model. Hence, $V_{\text{final}}(i)$ and $R_{\text{total}}(i)$ in (5) are identical to those in (3) except the equivalent resistance of the power network is updated as $R_{\text{pn}}(i)$.

After a new power switch s_i is turned on in the daisy chain, the associated on-resistance does not always stay in $R_{s_i,\text{on}}$ since its on-resistance varies as the voltage across its two terminals VDD and VDD_OFF vary. The voltage value in VDD_OFF changes when power switches are turned on. The on-resistance variance has an influence on the accuracy of the transient voltage estimation. However, it is omitted in the previous equations. To make the model of on-resistance variance more precise, we take the terminal voltage of VDD_OFF into consideration. Let $R'_{s_i,\text{on}}$ denote the incomplete on-resistance of s_i when the cross voltage of s_i is equal to $\text{VDD} - \text{VSS}$, and $R_{s_i,\text{on}}$ denotes the complete on-resistance of s_i when the cross voltage of s_i is about 0 V (i.e., 0.01 V). Since the relationship between the on-resistance and the voltage of VDD_OFF is about linear when the transistor is operating in linear and

saturation region, the on-resistance of s_i can be modeled more precisely by the following function $R_{s_i, \text{on}}(v)$:

$$R_{s_i, \text{on}}(v) = R'_{s_i, \text{on}} - R_v \times v \quad (6)$$

where

$$R_v = \frac{(R'_{s_i, \text{on}} - R_{s_i, \text{on}})}{(\text{VDD} - \text{VSS})} \quad (7)$$

R_v denotes the power switch on-resistance variation coefficient. Since $R_{\text{pn}}(i)$ is influenced by $R_{s_i, \text{on}}(v)$ in (5), $R_{\text{pn}}(i)$ should be replaced by $R_{\text{pn}}(i, v)$. $R_{\text{total}}(i)$ and $V_{\text{final}}(i)$ are associated with $R_{\text{pn}}(i)$. They should be replaced by $V_{\text{final}}(i, v)$ and $R_{\text{total}}(i, v)$. Note that $R_{s_i, \text{on}}$ and $R'_{s_i, \text{on}}$ can be obtained from library. More precise $\text{VDD_OFF}(t)$ is shown as follows:

$$\text{VDD_OFF}(t) = \begin{cases} \text{VSS}, & \text{if } 0 \leq t < T_{s_1} \\ \text{VDD_OFF}(t, i, v), & \text{if } T_{s_i} \leq t < T_{s_{i+1}}. \end{cases} \quad (8)$$

The value of $\text{VDD_OFF}(t)$ should be calculated for each time period $[T_{s_i}, T_{s_{i+1}})$ because $\text{VDD_OFF}(t)$ is discontinuous at each time point $t = T_{s_i}$.

C. Equations for Estimating Rush Current

This section deduces an equation to estimate the total current through the power switches. Let $I_{\text{VDD}}(t)$ denote current flowing through power switches at time t . According to our model, $I_{\text{VDD}}(t)$ can be computed by the following equation:

$$I_{\text{VDD}}(t) = \frac{\text{VDD} - \text{VDD_OFF}(t)}{R_{\text{pn}}(t)}. \quad (9)$$

In the previous section, the transient voltage obtained from (2) is based on the assumption that each power switch s_i is turned on immediately at time T_{s_i} . The value is accurate enough under the assumption. However, it will lead to an intolerable error in computing $I_{\text{VDD}}(t)$ because it takes time to change a power switch from the power-off state to the power-on state in an actual circuit (i.e., the resistance gradually decreases from ∞ to $R'_{s_i, \text{on}}$). To resolve this problem, we introduce a function to model the change of resistance associated with s_i to get a more precise current value.

The change of resistance associated to s_i is modeled by the following function:

$$R_{s_i}(t, v) = \frac{R_{s_i, \text{on}}(v)}{c_{s_i}(t)} \quad (10)$$

where $R_{s_i, \text{on}}(v)$ denotes the on-resistance of s_i when the voltage of VDD_OFF is equal to v , and $c_{s_i}(t)$ denotes power switch transient coefficient. $c_{s_i}(t)$ is determined depending upon the switching behavior of transistor, and it is computed by the following equation:

$$c_{s_i}(t) = \begin{cases} 0, & 0 \leq t < T_{s_i} \\ (t - T_{s_i})/t_{\text{on}}, & T_{s_i} \leq t < T_{s_i} + t_{\text{on}} \\ 1, & T_{s_i} + t_{\text{on}} \leq t < \infty. \end{cases} \quad (11)$$

To get a precise value from $R_{s_i}(t, v)$, $c_{s_i}(t)$ should be an exponential function. We use a linear function to simplify computation. A power switch is in the stable power-off state during time interval $[0, T_{s_i})$. During time interval $[T_{s_i}, T_{s_i} + t_{\text{on}})$, the value $c_{s_i}(t)$ increases linearly from 0 to 1 and the value $R_{s_i}(t)$

decreases from ∞ to $R_{s_i, \text{on}}$, where t_{on} denotes the required time for the change of the resistance. This is considered the transient time period of a power switch. After time $T_{s_i} + t_{\text{on}}$, the resistance of s_i becomes $R_{s_i, \text{on}}$. Therefore, $R_{\text{pn}}(t)$ in (9) can be represented as

$$R_{\text{pn}}(t, v) = \left(\sum_{i=1}^K \frac{1}{R_{s_i}(t, v)} \right)^{-1} = \left(\sum_{i=1}^K \frac{c_{s_i}(t)}{R_{s_i, \text{on}}(v)} \right)^{-1} \quad (12)$$

where K denotes the total number of power switches in a power network. By replacing $R_{\text{pn}}(t)$ in (9) with (12), $I_{\text{VDD}}(t)$ can be expressed as follows:

$$I_{\text{VDD}}(t) = \frac{\text{VDD} - \text{VDD_OFF}(t)}{\left(\sum_{i=1}^K \frac{c_{s_i}(t)}{R_{s_i, \text{on}}(\text{VDD_OFF}(t))} \right)^{-1}}. \quad (13)$$

Let I_{rush} denotes the rush current in a power network. It can be computed by the following equation:

$$I_{\text{rush}} = \mathbf{Max}\{I_{\text{VDD}}(T_{s_i}), 1 \leq i \leq K\} \quad (14)$$

where K represents the number of power switches in the power network.

IV. ALGORITHM FOR DETERMINING THE DEPTH OF DAISY CHAIN

The depth of the daisy chain has an important impact on wake-up time and rush current. This section shows a procedure to determine the suitable depth for a given daisy chain. According to the equations shown in the last section, the transient voltage and rush current in the virtual VDD can be estimated when a new power switch in a daisy chain is turned on. The suitable number of power switches can be found if a stable voltage in the virtual VDD is given.

Given a daisy chain which contains many power switches and connected in serial, this section shows a procedure to prune unnecessary power switches whose turn-on timing has scarcely no impact on rush current.

Before illustrating our procedure, we define the following notations.

- 1) Let L denote a daisy chain, and L is a sequence $s_1, s_2, \dots, s_{|L|}$, where s_i denotes a power switch.
- 2) Let $|L|$ denote the number of power switches in L .
- 3) Let δ_{s_i} denote the control signal delay from s_i to s_{i+1} .
- 4) Let T_{s_i} denote the turn on time of s_i , and $T_{s_i} = T_{s_{i-1}} + \delta_{s_{i-1}}$.

Power switches in L are turned on in sequence and s_i is turned on at time T_{s_i} . During $[T_{s_i}, T_{s_{i+1}})$, the two layers of power meshes are connected by i power switches (i.e., s_1 to s_i). The voltage of the virtual VDD and current flowing through power switches has to be recalculated in each time interval. The virtual VDD at time t is denoted by $\text{VDD_OFF}(t)$, and the current flowing through power switches at time t is denoted by $I_{\text{VDD}}(t)$. To obtain values of $\text{VDD_OFF}(t)$ and $I_{\text{VDD}}(t)$ at each time t , we summarize two theorems in (15) and (16) from Section III. According to (8) in Section III-B, $\text{VDD_OFF}(t)$ of interval $[T_{s_i}, T_{s_{i+1}})$ can be computed by Theorem 1.

Theorem 1:

$$\text{VDD_OFF}(t) = V_{\text{final}} + (V_{\text{init}} - V_{\text{final}}) \times e^{\frac{-(t-T_{s_i})}{R_{\text{total}} \times C_{\text{std}}}}$$

where

$$\left\{ \begin{array}{l} V_{\text{final}} = \text{VDD} \times \frac{R_{\text{leak}}}{R_{\text{leak}} + R_{\text{pn}}(i, V_{\text{init}})} \\ \quad + \text{VSS} \times \frac{R_{\text{pn}}(i, V_{\text{init}})}{R_{\text{leak}} + R_{\text{pn}}(i, V_{\text{init}})} \\ V_{\text{init}} = \text{VDD_OFF}(T_{s_i}) \\ R_{\text{total}} = R_{\text{pn}}(i, V_{\text{init}}) \parallel R_{\text{leak}} \\ R_{\text{pn}}(i, V_{\text{init}}) = \left(\sum_{j=1}^i \frac{1}{R_{s_j, \text{on}}(V_{\text{init}})} \right)^{-1} \end{array} \right.$$

Based on (14), the max current $\text{MAX } I_{\text{VDD}}(t)$ flowing through power switches at the interval $[T_{s_i}, T_{s_{i+1}})$, which is denoted by I_{MAX} , can be computed by Theorem 2.

Theorem 2:

$$I_{\text{MAX}} = \text{MAX}\{I_{\text{VDD}}(T_{s_i}), I_{\text{VDD}}(T_{s_{i+1}})\}, \forall i$$

where $I_{\text{VDD}}(t) = \frac{\text{VDD} - \text{VDD_OFF}(t)}{R_{\text{pn}}(t)}$ and

$$R_{\text{pn}}(t) = \left(\sum_{i=1}^K \frac{c_{s_i}(t)}{R_{s_i, \text{on}}(\text{VDD_OFF}(t))} \right)^{-1}.$$

Based on Theorems 1 and 2, the procedure to find a required number of power switches in L is shown in Fig. 5. If we are able to charge the virtual VDD to a suitable voltage value by a proper amount of power switches in a daisy chain, the maximal rush constraint will not be violated even if all remaining power switches are turned on at the same time. The voltage value is defined as V_{limit} . In other words, V_{limit} is the parameter for robustness. It can help prevent too large rush current if appropriate value is assigned to V_{limit} . The recommended value for V_{limit} is about 85%–90% of VDD by experience. Since the power switches in L are turned on in sequence, Fig. 5 uses an iterative method to compute the maximum rush current and transient voltage when a new power switch s_i , $i = 1$ to $|L|$, is turned on. The output is a daisy chain, which is denoted by L_D . L_D is a subsequence of L (i.e., $L_D \subseteq L$).

First, T_{s_1} and I_{max} are initialized as zero in lines 1 and 2 and the initial transient voltage for the virtual VDD is set to VSS in line 3. Let V_{last} and I_{last} denote the transient voltage and current in the last iteration. Lines 4 and 5 set V_{last} and I_{last} to VSS and zero, respectively. Then, the procedure iteratively selects a new power switch s_i from L , $i = 1$ to $|L|$, and computes the corresponding current and transient voltage when s_i is turned on at $[T_{s_i}, T_{s_{i+1}})$ (see lines 6–24). The time period in each iteration is computed first. Let T_{start} and T_{end} denote the start and end of current time period, and their values are set in lines 8 and 9, respectively. When a new power switch s_i is turned on, the equivalent resistance $R_{\text{pn}}(T_{s_i})$ of power network for current calculation has to be updated (see line 10). Then, the current voltage value is computed according $R_{\text{pn}}(T_{s_i})$ and V_{last} in line 11, and the maximum rush current is updated in line 12. Next, the transient voltage $\text{VDD_OFF}(T_{\text{end}})$ at the end of the time period is computed. To compute the value in line 17, the required parameters including V_{final} , $R_{\text{pn}}(i, V_{\text{init}})$, V_{init} , and R_{total} are calculated in lines 13–16. If $\text{VDD_OFF}(T_{\text{end}})$ is

Determine_Depth_of_DC ($L, V_{\text{limit}}, L_D, I_{\text{rush}}$)
// L denotes a list of power switches
// V_{limit} denotes the required voltage in virtual VDD
// L_D denotes a list of power switches which achieve the required voltage
// I_{rush} denotes the rush current
// T_{start} denotes the start time of time interval $[T_{s_i}, T_{s_{i+1}})$
// T_{end} denotes the end time of time interval $[T_{s_i}, T_{s_{i+1}})$

1. Set $T_{s_1} = 0$
2. Set $I_{\text{max}} = 0$
3. Set $\text{VDD_OFF}(T_{s_1}) = \text{VSS}$
4. Set $V_{\text{last}} = \text{VSS}$
5. Set $I_{\text{last}} = 0$
6. **For** $i = 1$ **To** $|L|$ **Do**
7. $T_{s_{i+1}} = T_{s_i} + \delta_{s_i}$
8. $T_{\text{start}} = T_{s_i}$
9. $T_{\text{end}} = T_{s_{i+1}}$
10. **Comp_Equi_R_of_Cur_Network**($R_{\text{pn}}(T_{\text{start}})$)
11. $I_{\text{VDD}}(T_{\text{start}}) = \frac{\text{VDD} - V_{\text{last}}}{R_{\text{pn}}(T_{\text{start}})}$
12. $I_{\text{max}} = \text{MAX}\{I_{\text{max}}, I_{\text{VDD}}(T_{\text{start}})\}$
13. $V_{\text{init}} = V_{\text{last}}$
14. **Comp_Equi_R_of_on_switches**($R_{\text{pn}}(i, V_{\text{init}})$)
15. $V_{\text{final}} = \text{VDD} \times \frac{R_{\text{leak}}}{R_{\text{leak}} + R_{\text{pn}}(i, V_{\text{init}})} + \text{VSS} \times \frac{R_{\text{pn}}(i, V_{\text{init}})}{R_{\text{leak}} + R_{\text{pn}}(i, V_{\text{init}})}$
16. $R_{\text{total}} = R_{\text{pn}}(i, V_{\text{init}}) \parallel R_{\text{leak}}$
17. $\text{VDD_OFF}(T_{\text{end}}) = V_{\text{final}} + (V_{\text{init}} - V_{\text{final}}) \times e^{\frac{-\delta_{s_i}}{R_{\text{total}} \times C_{\text{std}}}}$
18. **If** ($\text{VDD_OFF}(T_{\text{end}}) \geq V_{\text{limit}}$) **Then**
19. $L_D = [s_1, \dots, s_i]$
20. $I_{\text{rush}} = I_{\text{max}}$
21. **Break**
22. **End if**
23. $V_{\text{last}} = \text{VDD_OFF}(T_{\text{end}})$
24. **End For**

Fig. 5. Procedure for determining depth in a daisy chain.

larger than V_{limit} , the procedure stops (lines 18–22). The procedure will return the first i power switches in L as the member of L_D (line 19), and the maximum rush current induced by these power switches is updated (line 20). Finally, V_{last} is updated to $\text{VDD_OFF}(T_{\text{end}})$ in line 23.

V. METHODOLOGY FOR DETERMINING TURN-ON SEQUENCE OF POWER SWITCHES

This section shows our methodology to determine a power switch turn-on sequence using a hybrid routing structure. See Fig. 6 for our flow. There are mainly two stages in the flow. First stage is construction of daisy chain and the second stage is construction of distributed fishbone. In the first stage, there are three steps for constructing the daisy chain. First, ILP select a set of power switches from layout to constitute a daisy chain. For each set of daisy chain, there are eight routing structures for selection to connect power switches. The proper depth and rush current of daisy chain for each routing structure are calculated through the algorithm described in Section IV. The run times of constructing daisy chain is N_T , and N_T is determined by user. There are $N_T * 8$ solutions will be generated

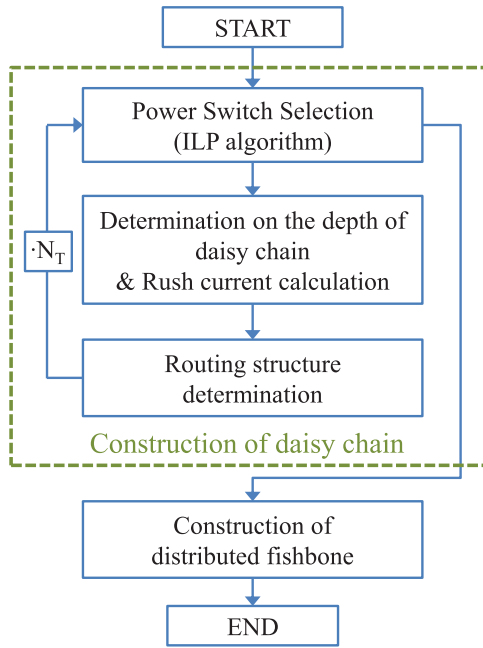


Fig. 6. Design flow of the methodology.

and we will pick the best solution from them. After finishing the construction of daisy chain, the remaining switches are constructed in the distributed fishbone.

In this section, we will detail each stage of our methodology. The first section described how to pick up a set of power switches by ILP algorithm. Then, the proposed algorithm to determine the connection sequence of the power switches are detailed in the next section. The last section describes the approach to connect the remaining power switches using the distributed fishbone structure.

A. Selection of Power Switches

This section shows an ILP-based approach to select power switches from the layout to constitute a daisy chain.

Suppose the number of power switches required by a daisy chain is determined. The ILP algorithm tries to select power switches from layout, which can induce less rush current. The rush current flowing through a power switch s_i is impacted by the following factors.

- 1) The linear regression resistance (denoted by $R_{s_i,on}$) of s_i .
- 2) The leakage power distribution (denoted by $P_{leak,i}$) and the density of standard cells (denoted by $D_{std,i}$) near s_i .

Hence, the following ILP is formulated, which consists of two terms for considering the two issues:

$$\begin{aligned}
 & \text{Minimize } \sum_{i=1}^{N_T} X_i \times \frac{1}{R_{s_i,on}} + \sigma \times P_{leak,i} \times D_{std,i} \\
 & \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \frac{1}{1 + \frac{N_{ran,i}}{N_{iter}}} \\
 & \text{s.t.} \\
 & \quad 1) \sum_{i=1}^{N_T} X_i = N_D \\
 & \quad 2) X_i = 0 \text{ or } 1
 \end{aligned} \tag{15}$$

where σ is a user specified parameter which gives the weight for the second term.

The first term tries to select power switches with large resistance. Power switches with large on-resistance values should be turned on early because this slows down the increasing speed of leakage current. Slower increasing speed of leakage current before t_{rush} will result in smaller rush current, where t_{rush} denotes the time that rush current occurs. The second term picks power switches with small $P_{leak,i}$ and $D_{std,i}$ to lower the number of standard cells fell under influence of rush current. Let N_T denote total number of power switches in the layout.

The traditional ILP formulation obtains an unique solution for each layout. Since the distribution of power switches is not considered in the object function, the solution obtained through ILP is not good enough. However, it is not easy to formulate this issue into the objective function. Therefore, two variables $N_{ran,i}$ and N_{iter} are used in the objective function to perturb the solution. In order to explore other possible solutions, the ILP algorithm is applied several times. Thus, N_{iter} starts from 1 and increases by 1 at every iteration. $N_{ran,i}$ is a random number and its value is between 0 and an user specified upper bound. A different solution set is obtained when the values of $N_{ran,i}$ and N_{iter} are changed. After a certain number of iterations, the value $1 + N_{ran,i}/N_{iter}$ will approach 1 and the iteration will stop since a fixed solution set is obtained (i.e., $N_{ran,i}/N_{iter} \approx 0$).

The first constraint restricts that the number of picked switches should be equal to N_D . Let N_D is the specified depth of a daisy chain. The second constraint restricts X_i equal to 1 or 0.

B. Construction of Daisy Chain

Based on the ILP-based approach presented in the last section, this section shows the algorithm to construct a daisy chain. The pseudo-code is shown in Fig. 7.

In order to satisfy the wake-up time constraint, the maximum depth of a daisy chain is determined first (line 1). Let N_D denote the maximum depth of a daisy chain which is computed by the equation $T_{wake_up}/\delta_{s_{min}}$, where T_{wake_up} denotes required wake-up time and $\delta_{s_{min}}$ denotes the minimal signal delay passing through a power switch. In line 1, the maximum depth of a daisy chain is computed roughly without the consideration of wiring delay. Usually, there exist a lot of power switches in a power domain. When our methodology constructs a daisy chain, we will pick neighboring power switches in layout as contiguous power switches from a given daisy chain picked up by ILP algorithm. Let L_{best} denote the best daisy chain and I_{best} denote the induced rush current of L_{best} . I_{best} is initialized as infinite (see line 2). The procedure is repeated N_T times, where N_T is a user specified parameter (see line 3). In each iteration, a set of power switches is picked from the layout to form a daisy chain. The ILP-based procedure illustrated in Section V-A is applied (see line 4). For each set S of power switches, they are connected by one of eight possible directions (see line 6). The resulting sequence is recorded in a list L_i , $i = 1 \dots 8$. Fig. 8 shows the eight possible routing directions of a daisy chain, which includes lower_LV, lower_LH, lower_RV, lower_RH, upper_LV, upper_LH, upper_RV, and

```

Build_a_Daisy_Chain ( $T_{wake\_up}$ ,  $N_T$ ,  $V_{limit}$ ,  $T_s$ ,  $L_{best}$ )
//  $N_T$  denotes the iteration number specified by a user
//  $V_{limit}$  denotes the stable voltage in virtual VDD
//  $T_{wake\_up}$  denotes the wake-up time constraint
//  $\delta_{s_{min}}$  denotes the smallest time delay of all the power switch
types
//  $L_{best}$  records a list of selected power switches

1. Set  $N_D = T_{wake\_up} / \delta_{s_{min}}$ .
2. Initialize the minimal rush current  $I_{best} = \infty$ .
3. For  $N_{iter} = 1$  To  $N_T$  Do
4.   Selection_of_Power_Switches( $N_D$ ,  $N_{iter}$ ,  $S$ )
5.   For  $i = 1$  To 8 Do
6.      $L_i = \text{Build\_DC}(i, S)$ 
7.     Determine_Depth_of_DC( $L_i$ ,  $V_{limit}$ ,  $L_D$ ,  $I_{rush}$ )
8.     if ( $I_{rush} < I_{best}$ ) or
        ( $I_{rush} == I_{best}$ ) and ( $D(L_D) < D(L_{best})$ )
9.       Then
10.         $I_{best} = I_{rush}$ 
11.         $L_{best} = L_D$ 
12.         $N_D = D(L_D)$ 
13.     End if
14.   End For
15. End For
    
```

Fig. 7. Algorithm for constructing a daisy chain from layout for the proposed hybrid routing structure.

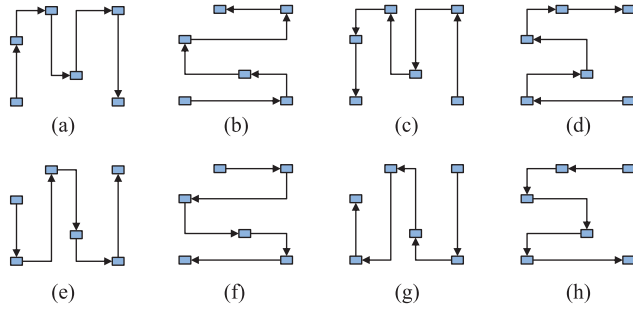


Fig. 8. Eight possible routing directions for a daisy chain. (a) Lower_LV. (b) Lower_LH. (c) Lower_RV. (d) Lower_RH. (e) Upper_LV. (f) Upper_LH. (g) Upper_RV. (h) Upper_RH.

upper_RH, where the first letter denotes (R)ight/(L)eft and the second letter denotes (H)orizontal/(V)ertical. For each L_i , the actual members in L_i which are able to charge the virtual VDD to the required voltage V_{limit} is determined by the procedure in Fig. 5 in Section IV, and a new daisy chain L_D and its induced rush current I_{rush} are returned (see line 7). Then, the best solution is updated if one of the following two situations is satisfied (see lines 8–13): 1) I_{rush} of L_D is less than I_{best} and 2) I_{rush} is equal to I_{best} and its depth [denoted by $D(L_D)$] is smaller than the depth [denoted by ($D(L_{best})$)] of L_{best} .

C. Construction of Distributed Fishbone

Since the voltage in the virtual VDD is charged to the VDD by the daisy chain constructed by the last subsection, the remaining power switches are connected by the fishbone structure to reduce wake-up (or sequence) time. However, if the number of power switches in the daisy chain is too small, the last power switch in the daisy chain may have a low-slew

rate once the fishbone has large output load. To resolve this problem, the remaining power switches are connected by the distributed fishbone structure.

The procedure to build a distributed fishbone structure is shown in the following: first, the remaining power switches are partitioned into K parts based on their horizontal coordinates, and one power switch is selected as the root of the power switches in each part. Then, the power switches in each part are connected by the fishbone structure. Finally, the last power switch in the daisy chain is connected to these K power switches. To ensure that the daisy chain is not impacted by the distributed fishbone, K should be much less than the number of the power switches in the daisy chain. Because K power switches are used to share the loading of the remaining power switches instead one, violation of slew rate can be avoided.

VI. EXPERIMENTAL RESULTS

This section shows our experimental results. The proposed methodology was implemented in the C++ programming language and compiled by g++ 4.6.2. The program was run on a CentOS 5.1 machine having an Intel XEON E5520 CPU and 62 GB memory. Our experiment can be divided into two parts. The first part demonstrates the accuracy of the equations from our model. Then, the second part shows effectiveness and efficiency of our methodology.

A. Waveform Comparison With HSPICE Simulation Results

To show feasibility of our model, this section compares the results estimated from our equations with the simulation results from HSPICE. We construct a daisy chain composed of K power switches. Assume the first power switch is turned on at time T_s . Each power switch s_i is turned on at time $T_s + \delta \times (i - 1)$, where i is the index of power switch in the daisy chain and δ denotes the time delay between s_i and s_{i+1} . The transient voltage and total current are computed after a new power switch is turned on according to the equations shown in Section III, and they are compared with simulation results from HSPICE.

HSPICE is performed based on the model in Fig. 4(e). To make the simulation results closer to a real circuit, the rising time of the power switch control signal is set as 50 ps. The parameters used in HSPICE are as follows.

- 1) The leakage resistor $R_{leak} = 500 \Omega$.
- 2) The capacitance of standard cells $C_{std} = 5000 \text{ pF}$.
- 3) The on-resistance of power switches $R_{on} = 205 \Omega$.
- 4) The incomplete on-resistance of power switches $R'_{on} = 583 \Omega$.
- 5) $T_s = 3 \text{ ns}$.
- 6) $\delta = 50 \text{ ps}$.
- 7) $K = 1600$.

Fig. 9 shows the resulting waveforms of transient voltage and current from our equations and from HSPICE, respectively. The x -axis of the figure denotes simulation time while the y -axis denotes transient voltage and current in Fig. 9(a) and (b). The dashed line represents the values calculated from our equations while the solid line denotes the results

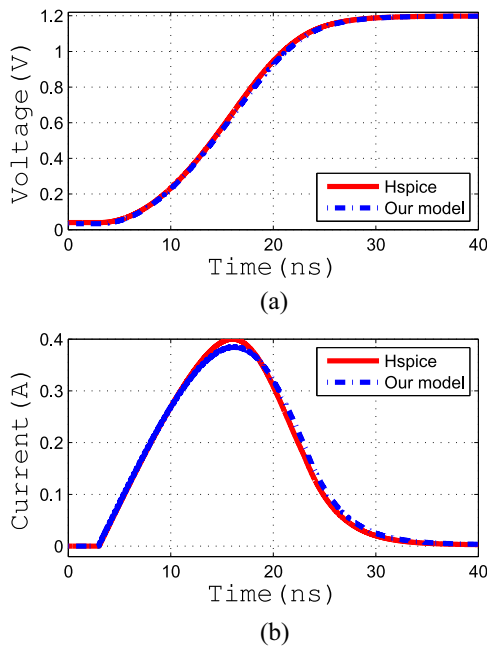


Fig. 9. (a) Voltage values estimated from our equation and from HSPICE. (b) Current values estimated from our equation and from HSPICE.

of HSPICE. Fig. 9(a) reveals the largest variation of voltage values estimated from our equations versus those obtained from HSPICE is smaller than 4%. And the peak current in our equation is 3% smaller than that of HSPICE in Fig. 9(b). Also, to verify the variance of the model accuracy, we perform the Monte Carlo simulation of switch types and turn-on delay. In the actual circuit, there might be several types of power switches for selection. And the turn-on delays between each pair of contiguous power switches are different. To perform the Monte Carlo simulation, we randomly generate 50 test sets. Each test set assigns the type and the turn-on delay of power switches. There are three types of power switch for selection. The R_{on} and R'_{on} of the types I power switch are 205 and 583 Ω . The R_{on} and R'_{on} of the types II power switch are 288 and 983 Ω . The R_{on} and R'_{on} of the types III power switch are 91 and 241 Ω . And the value of turn on delay varies between 40 and 200 ns. For each test case, we compare the simulation results obtained from our model and from HSPICE simulation. Let R_{senti} and W_{senti} denote the sensitivity ratio of rush current and wake-up time, which are calculated by the following equations:

$$R_{senti}(\%) = \frac{R_{model} - R_{HSPICE}}{R_{HSPICE}} \times 100\% \quad (16)$$

$$W_{senti}(\%) = \frac{W_{model} - W_{HSPICE}}{W_{HSPICE}} \times 100\% \quad (17)$$

where R_{model} and R_{HSPICE} denote the rush currents calculated from our equation and extracted from HSPICE, and W_{model} and W_{HSPICE} denote the wake-up time calculated from our equation and extracted from HSPICE. The mean and standard deviation of W_{senti} are 1.19% and 0.114%. The mean and standard deviation of R_{senti} are -1.59% and 0.3579%.

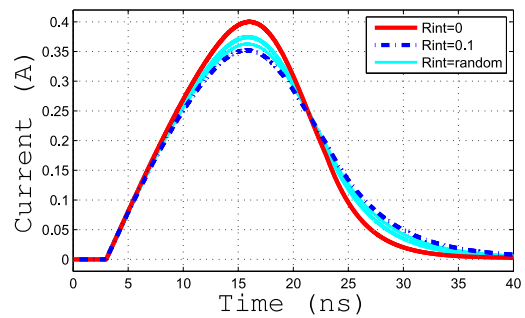


Fig. 10. Transient current after the internal resistance of the power lines is considered.

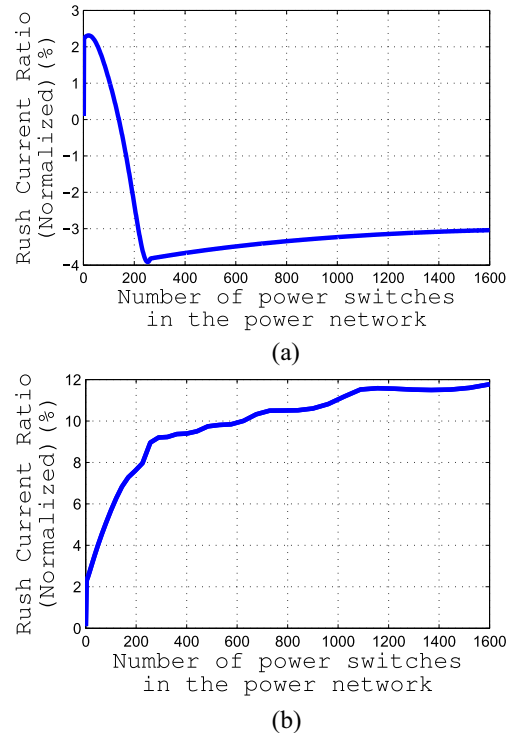


Fig. 11. Rush current after each power switch in a daisy chain is (a) turned on and (b) turned on while $R_{int} = 0.1 \Omega$.

TABLE I
INFORMATION OF DESIGNS

Circuit	# of std. cells	Total power(mw)	Leakage power(mw)	Global VDD(V)	Stable virtual VDD(V)
Cir.A	1595662	268.8	0.291	1.08	1.047
Cir.B	1632538	446.3	56.396	1.08	1.069
Cir.C	2540941	1307.1	0.15	1.32	1.27

In fact, our peak current is larger than that of a real circuit if the internal resistance of power lines is considered. Fig. 10 shows the rush current when the resistance of power lines is considered in HSPICE. The architecture of the circuit after with the consideration of internal wire resistance is the same as the architecture in Fig. 4(c). But in actual circuits, the resistance values of wire resistances are different. To model the circuits more accurately, we perform the Monte Carlo simulation of wire resistances. We varies the value of wire resistance between 0.05 and 0.1 Ω . The run time is 50. In Fig. 10, the red line represents the results if the resistance of

TABLE II
COMPARISON OF RUSH CURRENT, WAKE-UP TIME, AND SEQUENCE TIME OF DIFFERENT ROUTING STRUCTURES

Circuits	Number of power switches	Daisy chain			Fishbone			Hybrid structure			
		Rush current (mA)	Wake-up time (ns)	Sequence time (ns)	Rush current (mA)	Wake-up time (ns)	Sequence time (ns)	Rush current (mA)	Wake-up time (ns)	Sequence time (ns)	Run time (s)
Cir.A	4874	427	102.59	994.25	2122	35.01	67.02	309	111.51	433.10	129.2
Cir.B	10595	427	96.67	1991.69	1930	30.70	98.53	208	107.54	600.15	288
Cir.C	10845	2101	53.36	798.63	19270	10.48	22.08	1478	78.05	114.35	348
normalized	-	1.48	0.85	3.30	11.69	0.26	0.16	1	1	1	-

power lines is ignored while the blue line represents the conditions where the resistance of power lines are set at 0.1Ω . And the Monte Carlo simulation results are denoted by cyan lines, and all cyan lines are very centralized. It can be seen that the total current becomes smaller if the resistance of power lines is considered.

Next, we turn on power switches in the daisy chain sequentially and compare the sensitivity ratio of rush current estimated from our model versus HSPICE. The experiment parameters are the same as the ones in Section VI-A. The sensitivity ratio of rush current is calculated based on (16). The results are shown in Fig. 11, where the x -axis denotes the number of the turned-on power switches while the y -axis denotes the variation value. In Fig. 11(a), it shows the sensitivity ratio without the consideration of internal resistance. The figure reveals that the rush current ratio is smaller than 4%. However, if the internal resistance of the power lines is considered, the rush current variation ratio becomes positive. In Fig. 11(b), it shows the sensitivity ratio when the resistance of each power line is set to 0.1Ω . It can be seen that the rush current estimated by our equation can be regarded as the estimation result of rush current on worst case.

B. Test Cases Simulation Results

This section shows the effectiveness of our methodology. Three sets of test cases designed by Himax Technologies Inc. are used to test our program. Our design flow is described as follows. To get the locations of power switches, placement density, and leakage power near a power switch, we first dump layout information (i.e., in `def` and `lef` formats) from IC Compiler (ICC) [20]. The linear regression equivalent resistance of each power switch s_i and its delay δ_{s_i} are obtained from layout and library. After placement and connection of power switches are determined by our program, the results are imported back to ICC via TCL file. Finally, the rush current, wake-up time, and sequence time of circuits are measured by PrimeRail [20].

The experiments were performed on three circuits, and the information of the circuits is listed in Table I. In this experiment, we compare the ICC results of our hybrid routing structure with the two classic routing structures including daisy chain and fishbone. We do not discuss high fan-out structure because the driving strength of a signal source is usually not strong enough to drive a large number of power switches. Thus, the high fan-out structure is seldom singly used in a design. The comparison results among daisy chain,

fishbone, and our hybrid structure are listed in Table II. Columns 3–5 (6–8) show the experimental results of the daisy chain (fishbone) structure while our results are shown in columns 9–12. The design specification goals are to mitigate the rush current under 500 mA for Cir.A and Cir.B, and 2000 mA for Cir.C. The wake-up time and sequence time should be under 1000 ns. Based on the results, the daisy chain and fishbone structures cannot satisfy the rush current constraints. Note that the fishbone structure induces 19270 mA rush current in Cir.C. Although the daisy chain structure can meet rush current requirement in Cir.A and Cir.B, it cannot satisfy the rush current constraint in Cir.C. Moreover, the daisy chain structure induces longer sequence time. Note that the daisy chain violates sequence time in Cir.B. Compared to the daisy chain structure, our hybrid routing structure not only obtains smaller sequence time, but also induces smaller rush current. Since serious rush current usually occurs early, turning on the power switches which have larger equivalent resistance and smaller leakage current density around them in the early part of daisy chain are beneficial for suppressing the rush current. As described in Section V-A, power switches with larger on-resistance are turned on earlier in the sequence, resulting in slower increasing speed of leakage current and lower rush current. Because the power switches in our daisy chain are selected by the ILP algorithm (see Section V-A), our hybrid routing structure induces smaller rush current than the daisy chain structure. Besides, since remaining power switches are turned on quickly by the distributed fishbone, the sequence time of the hybrid structure is much smaller than daisy chain. Compared to the fishbone structure, rush current induced by our hybrid routing structure is significantly smaller than that generated by the fishbone structure. Though the wake-up and sequence times of our hybrid structure are larger than the fishbone structure, they still satisfy the timing constraints. The execution time of each test case is listed in column 12. It can be seen that our program can generate hybrid structure within few minutes even for large test case which contains thousands of power switches.

Figs. 12–14 show our hybrid structures for Cir.A, Cir.B, and Cir.C, respectively. The left side is connected by a daisy chain structure, the middle side is connected by fishbone structure, and the right side is connected by our hybrid structure. The white points are the power switches, the lines indicate the routing structure, and the yellow arrows on lines mean the routing order and directions of power switches.

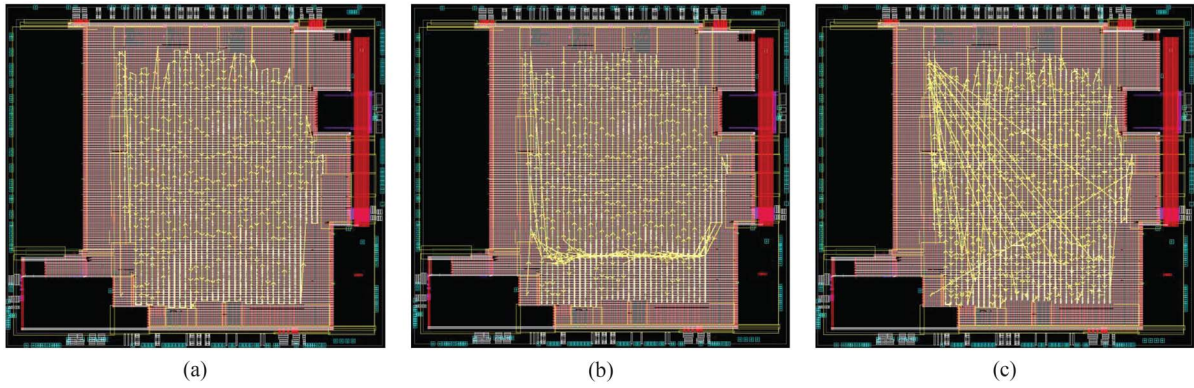


Fig. 12. Signal routing results of (a) daisy chain structure, (b) fishbone structure, and (c) our hybrid structure of Cir.A shown in ICC.

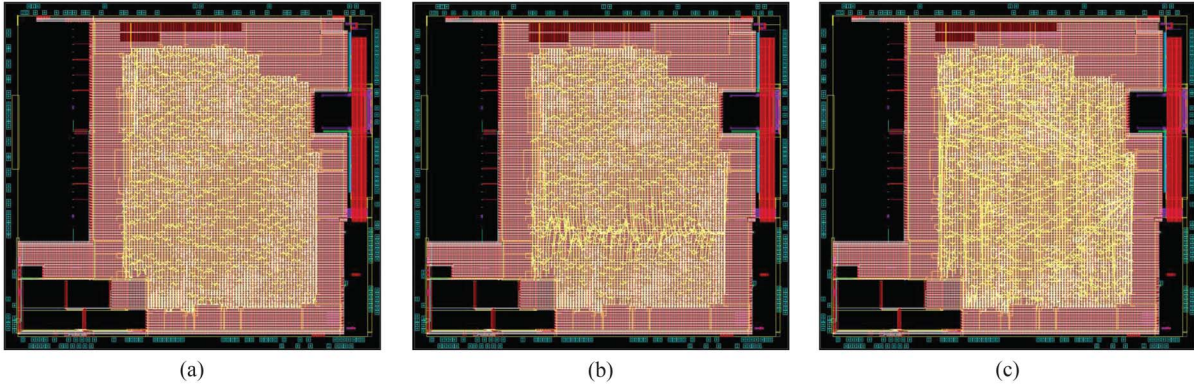


Fig. 13. Signal routing results of (a) daisy chain structure, (b) fishbone structure, and (c) our hybrid structure of Cir.B shown in ICC.

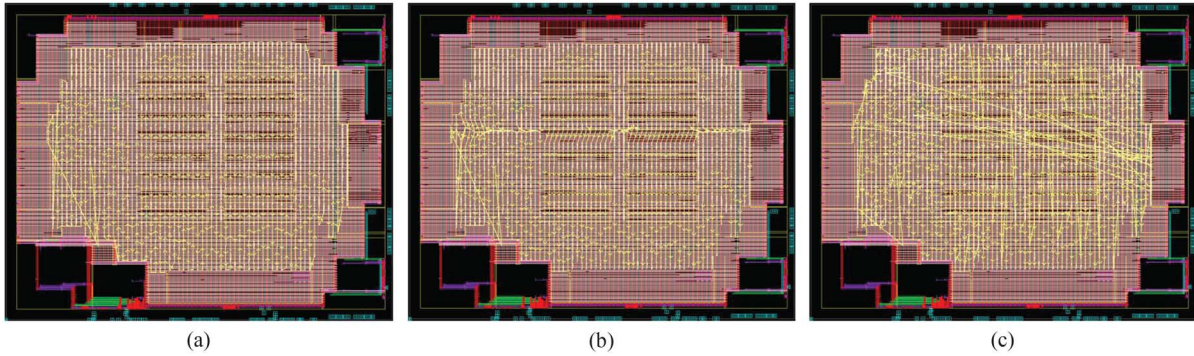


Fig. 14. Signal routing results of (a) daisy chain structure, (b) fishbone structure, and (c) our hybrid structure of Cir.C shown in ICC.

VII. CONCLUSION

This paper proposed a methodology to build a hybrid routing structure to connect power switches for power gating design. Our hybrid routing structure induces less rush current and satisfies timing constraints because a better daisy chain is built. To determine depth of a daisy chain, we have proposed a simplified model for power gating design and produced precise equations to estimate voltage and transient current. The experimental results demonstrated the stability and effectiveness of our method. All of our experiments are based on industrial designs and measured by vendor tools.

APPENDIX

This section derives the solution [see (2)] for (1). Shown in Fig. 15(a) is the simplified power network based on the

C1C2 model with leakage resistance, where R_L denotes the leakage resistance and C_{std} denotes the parasitic capacitance. R_t represents the equivalent parallel resistance of the power switches which are in the stable power-on state, and s denotes a single power switch which is not turned on. The node voltage V_x corresponds to a virtual VDD node. Assume the power switch s is turned on at time T_p .

The resistance of s at time t is denoted as $R_s(t)$ and the value of $R_s(t)$ can be divided into two states as follows:

$$R_s(t) = \begin{cases} \infty, & 0 \leq t < T_p \\ r_{on}, & T_p \leq t < \infty. \end{cases}$$

At the time interval $[0, T_p)$, the power switch is turned off and its resistance becomes infinite to denote an open circuit [see Fig. 15(b)]. At time T_p , the power switch is turned on and let r_{on} denote the on-resistance of the power switch in the

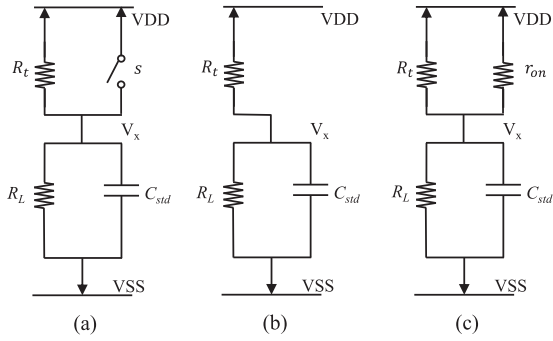


Fig. 15. (a) C1C2 model with leakage resistor and power switch s . Equivalent circuit where s acts as an (b) open circuit and (c) resistor.

stable power-on state [see Fig. 15(c)]. In the actual circuit, the power switch takes time t_{on} to be turned on completely from power-off state to power-on state and is termed the transient state. For simplification, the transient state is not considered in the derivative process.

Since $R_s(t)$ is not continuous at time T_p , the transient voltage should be solved for each time interval. The derivative process is shown in the following equations.

- 1) When $0 \leq t < T_p$: Because $V_x(t)$ is in steady state within this time interval, there is no current flowing through the switches to charge C_{std} . According to superposition theorem, $V_x(t)$ can be written as

$$V_x(t) = \text{VDD} \times \frac{R_L}{R_t \parallel R_s(t) + R_L} + \text{VSS} \times \frac{R_t \parallel R_s(t)}{R_t \parallel R_s(t) + R_L}. \quad (18)$$

Since the value $R_t \parallel R_s(t)$ is approximately R_t [i.e., $R_s(t)$ is infinite in the interval], the transient voltage is determined by the following equation:

$$V_x(t) = \text{VDD} \times \frac{R_L}{R_t + R_L} + \text{VSS} \times \frac{R_t}{R_t + R_L}. \quad (19)$$

Hence, we use V_{init} to denote the stable voltage value of $V_x(t)$ because it maintains constant in $[0, T_p]$ as follows:

$$V_{init} = \text{VDD} \times \frac{R_L}{R_t + R_L} + \text{VSS} \times \frac{R_t}{R_t + R_L}. \quad (20)$$

- 2) When $T_p \leq t < \infty$: Since the power switch is turned on at T_p , the resistance of s becomes r_{on} after T_p , and current starts to flow through s . Because of the instant extra current, C_{std} is charged and the value V_x increases. However, the increased value in V_x results in slower charging speed of C_{std} [note that the charging speed is proportional to $((\text{VDD} - V_x(t))/R_t \parallel r_{on})$] and the slower charging speed makes the rising speed of V_x even slower. Finally, there is no current through C_{std} when $t \rightarrow \infty$, and V_x becomes constant. The C_{std} acts as an open circuit, and power switch s acts as the resistor. The circuit reaches stable condition. According to KCL, $V_x(t)$ must satisfy the following equation:

$$\frac{\text{VDD} - V_x(t)}{R_s(t)} = C_{std} \times \frac{d(V_x(t) - \text{VSS})}{dt} + \frac{V_x(t) - \text{VSS}}{R_L} + \frac{V_x(t) - \text{VDD}}{R_t}. \quad (21)$$

Because of the limitation in the integration time period, the time axis is shifted. To make T_p the original point, we set $t^* = t - T_{op}$, and $R_s(t) = r_{on}$ in the time interval $t^* > 0$. Hence, (21) is transformed into the following equation:

$$\frac{\text{VDD} - V_x(t^*)}{r_{on}} = C_{std} \times \frac{d(V_x(t^*) - \text{VSS})}{dt^*} + \frac{V_x(t^*) - \text{VSS}}{R_L} + \frac{V_x(t^*) - \text{VDD}}{R_t}. \quad (22)$$

Let $V_{x,tp}(t^*) = V_x(t^*) - \text{VSS}$. Equation (23) is obtained if we replace the term $V_x(t^*)$ by $V_{x,tp}(t^*) + \text{VSS}$ as follows:

$$\frac{\text{VDD} - \text{VSS} - V_{x,tp}(t^*)}{r_{on}} = C_{std} \times \frac{dV_{x,tp}(t^*)}{dt^*} + \frac{V_{x,tp}(t^*)}{R_L} + \frac{V_{x,tp}(t^*) + \text{VSS} - \text{VDD}}{R_t}. \quad (23)$$

Rearranging the above equation, we obtain (24) as follows:

$$\begin{aligned} (\text{VDD} - \text{VSS}) \times \left(\frac{1}{r_{on}} + \frac{1}{R_t} \right) &= C_{std} \times \frac{dV_{x,tp}(t^*)}{dt^*} \\ &+ \left(\frac{1}{r_{on}} + \frac{1}{R_L} + \frac{1}{R_t} \right) \times V_{x,tp}(t^*) \\ \frac{\text{VDD} - \text{VSS}}{r_{on} \parallel R_t} &= C_{std} \times \frac{dV_{x,tp}(t^*)}{dt^*} + \frac{V_{x,tp}(t^*)}{r_{on} \parallel R_L \parallel R_t}. \end{aligned} \quad (24)$$

Equation (24) is a linear differential equation with constant coefficients. According to the method of solution for linear differential equation with constant coefficients, the general solution $V_{x,tp}(t^*)$ is equal to $V_{x,N} + \alpha \times V_{x,H}(t^*)$, where $V_{x,H}(t^*)$ is the homogenous solution and $V_{x,N}$ is the nonhomogeneous solution. According to the method of solution for linear differential equation with constant coefficients, $V_{x,H}(t^*) = e^{ct^*}$, c is a constant. Reordering $V_{x,tp}(t^*) = V_{x,N} + \alpha \times e^{ct^*}$ and substituting it back to (24), the following equation is obtained:

$$\frac{\text{VDD} - \text{VSS}}{r_{on} \parallel R_t} = C_{std} \times (\alpha c \times e^{ct^*}) + \frac{V_{x,N} + \alpha \times e^{ct^*}}{r_{on} \parallel R_L \parallel R_t}. \quad (25)$$

Differentiating both sides of (25), the following equation is obtained:

$$\left(c \times C_{std} + \frac{1}{r_{on} \parallel R_L \parallel R_t} \right) \times e^{ct^*} = 0. \quad (26)$$

Because e^{ct^*} is larger than 0, the value c is obtained from (26)

$$c = \frac{-1}{(r_{on} \parallel R_L \parallel R_t) \times C_{std}}.$$

Thus

$$V_{x,H}(t^*) = e^{\frac{-t^*}{(r_{on} \parallel R_L \parallel R_t) \times C_{std}}}.$$

Taking $V_{x,H}(t^*)$ into $V_{x,tp}(t^*)$, we can get the following equation:

$$V_{x,tp}(t^*) = V_{x,N} + \alpha \times e^{\frac{-t^*}{r_{on}\|R_L\|R_t \times C_{std}}}. \quad (27)$$

Before calculating the value of $V_{x,N}$ and α , we need the boundary condition of $V_x(t^*)$ at $t^* = 0$ and $t^* = \infty$. At $t^* = 0$, the value of $V_x(t^*)$ equals to V_{init} as denoted in (20). At $t^* = \infty$, there is no current through C_{std} . C_{std} behaves as an open-circuit and s acts as a resistor. The value of $V_x(t)$ at $t^* = \infty$ can be obtained through the voltage divider rule or superposition theorem, and it is represented by V_{final} . The boundary conditions are shown in the following:

$$V_x(0) = V_{init}, \text{ and} \quad (28)$$

$$\begin{aligned} V_x(\infty) &= VDD \times \frac{R_L}{R_t\|r_{on} + R_L} + VSS \times \frac{R_t\|r_{on}}{R_t\|r_{on} + R_L} \\ &= V_{final}. \end{aligned} \quad (29)$$

Since $V_{x,tp}(t^*) = V_x(t^*) - VSS$, the following two equations are obtained if we consider the boundary conditions in (27):

$$V_{x,tp}(0) = V_{x,N} + \alpha = V_{init} - VSS \quad (30)$$

$$V_{x,tp}(\infty) = V_{x,N} = V_{final} - VSS. \quad (31)$$

The value of $V_{x,H}$ is obtained from (31). By substitution into (30), the value α is obtained as follows:

$$\begin{aligned} \alpha &= V_{init} - VSS - V_{x,N} \\ &= V_{init} - V_{final}. \end{aligned} \quad (32)$$

By taking α and $V_{x,N}$ into (27) and considering $V_{x,tp}(t^*) = V_x(t^*) - VSS$, $V_x(t^*)$ in the following can be obtained:

$$\begin{aligned} V_x(t^*) &= V_{x,tp}(t^*) + VSS \\ &= V_{final} + (V_{init} - V_{final}) \times e^{\frac{-t^*}{(r_{on}\|R_L\|R_t) \times C_{std}}}. \end{aligned} \quad (33)$$

Finally, the particular solution $V_x(t)$ within the time interval $T_p \leq t < \infty$ can be expressed by the following equation if time axis t^* is transformed back to t :

$$V_x(t) = V_{final} + (V_{init} - V_{final}) \times e^{\frac{-(t-T_p)}{(r_{on}\|R_L\|R_t) \times C_{std}}}. \quad (34)$$

The value of V_{init} and V_{final} are determined in (20) and (29).

From (34), we can conclude the operation concepts of the power switch s_i when turned on from power-off state at time T_p . When $t < T_p$, the circuit is stable (in the assumption), so the value of $V_x(t)$ is constant and independent of time variable t . This constant value is represented by V_{init} which denotes the output value of a voltage divider composed of R_t and R_L . At time $t = T_p$, s_i is triggered. After time T_p , the resistance of s_i becomes the on-resistance r_{on} . C_{std} starts to be charged by the current flowing through the power switches. The charging speed is determined by the RC-product of resistance $r_{on}\|R_L\|R_t$ and capacitance C_{std} as

shown in the exponent term of e in (34). In short, the R and C terms in RC-product denotes the equivalent capacitance and resistance link to the virtual VDD node. Finally, when $t \rightarrow \infty$, the voltage of the virtual VDD node equals V_{final} , which denotes the output value of the voltage divider is composed of $R_t\|r_{on}$ and R_L . When the power switch network is in a stable state, all capacitors act as open circuits and all power switches, which are in the stable power-on state, act as resistors.

REFERENCES

- [1] A. Davoodi and A. Srivastava, "Wake-up protocols for controlling current surges in MTCMOS-based technology," in *Proc. ASP-DAC*, Shanghai, China, 2005, pp. 868–871.
- [2] C. Long and L. He, "Distributed sleep transistor network for power reduction," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 12, no. 9, pp. 937–946, Sep. 2004.
- [3] C. Hwang, C. Kang, and M. Pedram, "Gate sizing and replication to minimize the effects of virtual ground parasitic resistances in MTCMOS designs," in *Proc. ISQED*, San Jose, CA, USA, 2006, pp. 741–746.
- [4] H. Jiang and M. Marek-Sadowska, "Power gating scheduling for power/ground noise reduction," in *Proc. DAC*, Anaheim, CA, USA, 2008, pp. 980–985.
- [5] J. N. Kozhaya and L. A. Bakir, "An electrically robust method for placing power gating switches in voltage islands," in *Proc. CICC*, Orlando, FL, USA, 2004, pp. 321–324.
- [6] J. Kao, A. Chandrakasan, and D. Antoniadis, "Transistor sizing issues and tool for multi-threshold CMOS technology," in *Proc. DAC*, Anaheim, CA, USA, 1997, pp. 409–414.
- [7] J. Kao, S. Narendra, and A. Chandrakasan, "MTCMOS hierarchical sizing based on mutual exclusive discharge patterns," in *Proc. DAC*, San Francisco, CA, USA, 1998, pp. 495–500.
- [8] K. Usami *et al.*, "Design and implementation of fine-grain power gating with ground bounce suppression," in *Proc. VLSI Design*, New Delhi, India, 2009, pp. 381–386.
- [9] L. K. Yong and C. K. Ung, "Power density aware power gate placement optimization," in *Proc. ASQED*, Penang, Malaysia, 2010, pp. 38–42.
- [10] P. F. Butzen and R. P. Ribas, "Leakage current in sub-micrometer CMOS gates," Univ. Federal do Rio Grande do Sul, Porto Alegre, Brazil, 2006. [Online]. Available: http://inf.ufrgs.br/logics/docman/book_emicro_butzen.pdf
- [11] D. Flynn, R. Aitken, A. Gibbons, and K. Shi, *Low Power Methodology Manual for System on Chip Design*. New York, NY, USA: Springer, 2007.
- [12] S. Paik, S. Kim, and Y. Shin, "Wake-up synthesis and its buffered tree construction for power gating circuit designs," in *Proc. ISLPED*, Austin, TX, USA, 2010, pp. 413–418.
- [13] S.-H. Chen, Y.-L. Lin, and M. C.-T. Chao, "Power-up sequence control for MTCMOS designs," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 3, pp. 413–423, Mar. 2013.
- [14] S. Kim, S. V. Kosonocky, and D. R. Knebel, "Understanding and minimizing ground bounce during mode transition of power gating structures," in *Proc. ISLPED*, Seoul, Korea, 2003, pp. 22–25.
- [15] T.-M. Tseng, M. C.-T. Chao, C.-P. Lu, and C.-H. Lo, "Power-switch routing for coarse-grain MTCMOS technologies," in *Proc. ICCAD*, San Jose, CA, USA, 2009, pp. 39–46.
- [16] V. Sreekumar and S. Ravichandran, "Impact of leakage and short circuit current in rush current analysis of power gated domains," in *Proc. SoutheastCon*, Concord, NC, USA, 2010, pp. 41–44.
- [17] V. Khandelwal and A. Srivastava, "Leakage control through fine-grained placement and sizing of sleep transistors," in *Proc. ICCAD*, San Jose, CA, USA, 2004, pp. 533–536.
- [18] Y.-T. Chen, D.-C. Juan, M.-C. Lee, and S.-C. Chang, "An efficient wake-up schedule during power mode transition considering spurious glitches phenomenon," in *Proc. ICCAD*, San Jose, CA, USA, 2007, pp. 779–782.
- [19] Y. Lee, D.-K. Jeong, and T. Kim, "Simultaneous control of power/ground current, wake-up time and transistor overhead in power gated circuits," in *Proc. ICCAD*, San Jose, CA, USA, 2008, pp. 169–172.
- [20] *Datasheet Synopsys*. [Online]. Available: <http://www.synopsys.com/home.aspx>, accessed Feb. 2016.



Ya-Ting Shyu received the M.S. degree in electrical engineering from National Cheng Kung University, Tainan, Taiwan, in 2008, where she is currently pursuing the Ph.D. degree.

Her current research interests include integrated circuit design and design automation for analog, mixed-signal circuits, and low-power techniques for digital design.



Chun-Po Huang was born in Tainan, Taiwan, in 1986. He received the B.S. degree in electrical engineering from National Cheng Kung University, Tainan, in 2008, where he is currently pursuing the Ph.D. degree.

His current research interests include design automation for high-speed and low-power analog-to-digital converters.



Jai-Ming Lin (M'08) received the B.S., M.S., and Ph.D. degrees from National Chiao Tung University, Hsinchu, Taiwan, in 1996, 1998, and 2002, respectively, all in computer science.

He is an Associate Professor with the Department of Electrical Engineering, National Cheng Kung University (NCKU), Tainan, Taiwan. From 2002 to 2007, he was an Assistant Project Leader with the CAD Team, Realtek Corporation, Hsinchu. He was an Assistant Professor with the Department of Electrical Engineering, NCKU, from 2007 to 2012.

His current research interests include physical design for 3-D-ICs, design automation for analog ICs, and low-power designs.



Soon-Jyh Chang (M'03) was born in Tainan, Taiwan, in 1969. He received the B.S. degree in electrical engineering from National Central University, Zhongli, Taiwan, in 1991, and the M.S. and Ph.D. degrees in electronic engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1996 and 2002, respectively.

Since 2003, he has been with the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan, where he has been a Professor and the Director of the Electrical Laboratories, since 2011. He has authored and co-authored around 100 technical papers and seven patents. His current research interests include design, testing, and design automation for analog and mixed-signal circuits.

Dr. Chang was a recipient and co-recipient of several technical awards, including the Greatest Achievement Award from National Science Council, Taiwan, in 2007, the National Chip Implementation Center Outstanding Chip Awards in 2008, 2011, and 2012, the Best Paper Awards of VLSI Design/CAD Symposium, Taiwan, in 2009 and 2010, the Best Paper Award of Institute of Electronics, Information and Communication Engineers in 2010, the Gold Prize of the Macronix Golden Silicon Award in 2010, the Best GOLD Member Award from the IEEE Tainan Section in 2010, the International Solid-State Circuits Conference/DAC Student Design Contest in 2011, and the ISIC Chip Design Competition in 2011. He has been the Chair of the IEEE Solid-State Circuits Society Tainan Chapter, since 2009. He was the Technical Program Co-Chair of the IEEE International Symposium on Next-Generation Electronics in 2010, and a Committee Member of the IEEE Asian Test Symposium in 2009, Asia and South Pacific Design Automation Conference in 2010, International Symposium on VLSI Design, Automation and Test in 2009, 2010, and 2012, and Asian Solid-State Circuits Conference in 2009 and 2011.



Che-Chun Lin received the B.S. degree in computer science and engineering from Chiao Tung University, Hsinchu, Taiwan, in 2012, and the M.S. degree in electrical engineering from National Cheng Kung University, Tainan, Taiwan, in 2014.

He is currently an Engineer with Himax Technologies, Tainan, Taiwan. His current research interests include power planning.