A Mathematical Solution to Power Optimal Pipeline Design by Utilizing Soft Edge Flip-Flops

Mohammad Ghasemazar, Behnam Amelifard and Massoud Pedram
University of Southern California
Department of Electrical Engineering
Los Angeles, CA 90089 U.S.A.
{*ghasemaz,amelifar,pedram*}*@usc.edu*

ABSTRACT
This paper presents a novel technique to minimize the total power consumption of a synchronous linear pipeline circuit by exploiting extra slacks available in some stages of the pipeline. The key idea is to utilize soft-edge flip-flops to enable time borrowing between stages of a linear pipeline in order to provide the timing-critical stages with more time to complete their computations. Time borrowing, in conjunction with keeping the clock frequency unchanged, gives rise to a positive timing slack in each pipeline stage. The slack is subsequently utilized to minimize the circuit power consumption by reducing the supply voltage level. We formulate and solve the problem of optimally selecting the transparency window of the soft-edge flip-flops and choosing the minimum supply voltage level for the pipeline circuit as a quadratic program, thereby minimizing the power consumption of the linear pipeline circuit under a clock frequency constraint. Experimental results prove the efficacy of the problem formulation and solution technique.

Categories and Subject Descriptors
B.8.2 [**Performance and Reliability**]: Performance Analysis and Design Aides

General Terms
Algorithms, Design.

Keywords
Low-power microprocessor design, Synchronous pipelines, Soft edge flip-flop, Voltage scaling, Quadratic programming.

1. INTRODUCTION
Excessive power dissipation and resulting temperature rise have become key limiting factors to processor performance and a significant component of its cost. In modern microprocessors, expensive packaging and heat removal solutions are required to achieve acceptable substrate and interconnect temperatures. Due to their high utilization, pipeline circuits of a high-performance microprocessor are major contributors to the overall power consumption of the processor, and consequently, one of the main sources of heat generation on the chip [1]. Many techniques have been proposed to reduce the power consumption of a microprocessor's pipeline among which pipeline gating [1], clock gating [2, 3], and voltage scaling [4] have proven to be effective.

In this paper we present a technique to address the problem of reducing the power consumption in a synchronous linear pipeline i.e., one with the following properties: (i) processing stages are linearly connected, (ii) it performs a fixed function, and (iii) stages are separated by flip-flops which are clocked with the same CLK signal. Our technique is based on the idea of utilizing *soft-edge flip-flops* (SEFF) for slack passing and voltage scaling in the pipeline stages. Soft-edge flip-flops have a small transparency window which allows time borrowing across pipeline stages. Soft-edge flip-flops have been traditionally used for minimizing the effect of clock skew on static and dynamic circuits [5, 6]. Recently, the authors of [7] proposed an approach to utilize soft-edge flip-flops in sequential circuits in order to minimize the effect of process variation on the yield. They formulated the problem of statistically aware SEFF assignment which maximizes the gain in timing yield as an integer linear program (ILP) and proposed a heuristic algorithm to solve the problem.

We describe a unified methodology for optimally selecting the supply voltage level of a linear pipeline and optimizing the transparency window of the SEFF so as to achieve the minimum power consumption subject to a total computation time (latency) constraint. We formulate this problem as a quadratic program, which is a convex programming problem, and hence can be solved optimally in polynomial time. The remainder of this paper is organized as follows. In Section 2 we provide some background on pipeline design and soft-edge flip-flops. Section 3 describes our techniques for reducing the power consumption. Section 4 is dedicated to simulation results and Section 5 concludes the paper.

2. BACKGROUND
2.1 Preliminaries
A simple (synchronous) 2-stage linear pipeline circuit is shown in Figure 1. We call the set of flip-flops that separate consecutive stages of the linear pipeline as a *FF-set*, for example, FF_0 … FF_2 are the FF-sets. Let's assume for now that the FF-sets used in this design are all hard-edge FF's. To guarantee the correct operation of the pipeline, the following timing constraints should be satisfied in all stages of the pipeline:

$$d_i + t_{s,i} + t_{cq,i-1} \le T_{clk} \quad 1 \le i \le N \tag{1}$$

$$\delta_i + t_{cq,i-1} \ge t_{h,i} \quad 1 \le i \le N \tag{2}$$

This research was sponsored in part by a grant from the National Science Foundation under award number 0509564.

ISLPED'08, August 11–13, 2008, Bangalore, India.

where d_i and δ_i are the maximum and minimum delays of combinational logic in stage i, T_{clk} denotes the clock cycle time, $t_{s,i}$ and $t_{h,i}$ are the setup and hold times for the flip-flops in the i^{th} FF-set while $t_{cq,i-1}$ denotes the clock-to-q propagation delay of the flip-flops in $i-1^{\text{st}}$ FF-set. N denotes the number of pipelines stages.

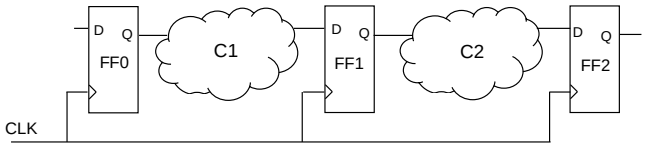

Figure 1. A simple linear pipeline

Equation (1) describes the constraint set on the maximum delay of pipeline stages to prevent setup time violations. It implies that the signal delay from one stage to the next stage should be less than a clock cycle by at least a setup time. The total delay is the sum of clock-to-q delay of the first stage and the longest path delay of the combinational circuit. Equation (2) describes the constraint set on the minimum delay of the pipeline stages to prevent data race hazard. In order not to overwrite the previous data, the new data of a stage must arrive at the next stage only after the hold time of the next stage FF has elapsed. The earliest time that new data can arrive at the next stage is the clock-to-q delay of the first stage plus the shortest path delay of the combinational logic in between the two stages. We have ignored the clock skew in both equations. To do so, we must add the clock skew, t_{skew}, to the left side of inequality (1) and subtract it from the left side of inequality (2).

2.2 Soft-Edge Flip-Flop

The key idea in designing a soft edge flip-flop [5] is to delay the clock of the master latch so as to create a window during which both master and slave latches are ON (cf. Figure 2). This window is called the *transparency window* of the SEFF and allows slack passing between adjacent pipeline stages separated by SEFF's. The delayed clock is achieved by utilizing an inverter chain and appropriately sizing inverters in the chain to achieve desired delay.

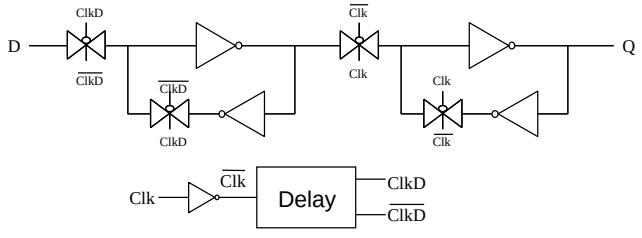

Figure 2. Master slave soft edge flip-flop

Referring back to Figure 1, for the sake of consistency with the input and output environments and to avoid imposing constraints on the sender or receiver of data for the linear pipeline circuit in question, we require that the first and last FF-sets in the pipeline are composed of hard-edge FF's whereas the intervening FF-sets may be SEFF's. Therefore, in this example, only FF1 can be made a soft-edge FF-set. In a SEFF, the transparency window size is an important parameter in the timing constraints since it changes the characteristics of the flip-flop. More precisely, the setup time, hold time, and clock-to-q delay of a soft-edge flip-flop are all functions of the transparency window width. By defining these timing parameters as functions of the window size, we can rewrite the timing constraints of a linear pipeline which utilizes SEFF's as,

$$d_i \le T_{clk} - t_{s,i}(w_i) - t_{cq,i-1}(w_{i-1}) \quad 1 \le i \le N \tag{3}$$

$$\delta_i \ge t_{h,i}(w_i) - t_{cq,i-1}(w_{i-1}) \quad 1 \le i \le N \tag{4}$$

Inequalities (3) and (4) are the SEFF versions of inequalities (1) and (2). Notice that the setup/hold times and the clock-to-q delay are now dependant on the transparency window size of the SEFF's.

Intuitively, it is expected that all three critical times of a SEFF, i.e., the setup time, hold time and clock-to-q delay, are postponed by the size of the transparency window w, because the data has more time to arrive. As a result, the setup time is *decreased* by w while the hold time and clock-to-q delay are *increased* by w. The reason for the linear dependence of the setup and hold times on w is that the input data may be read a time w after the clock edge. In section 3, we will show that the optimal window size of a SEFF is equal to the borrowed time in the preceding pipeline stage. In other words, in the optimal linear pipeline design, data arrives at the end of the transparency window of the SEFF, and as a result, the output of the SEFF is valid after a data to Q delay with respect to the end of transparency window, i.e., after $w + t_{dq}$ with respect to the clock edge. On the other hand, if there is no time borrowing, the output Q becomes valid only a clock to Q time, t_{cq}, after the clock edge. Based on the above discussion, the setup time, hold time, and clock-to-q delay of a SEFF may be modeled as linear functions of window size, as follows,

$$\begin{cases} t_{s,i}(w_i) = a_1 w_i + a_0 \\ t_{h,i}(w_i) = b_1 w_i + b_0 \\ t_{cq,i}(w_i) = c_1 w_i + c_0 \end{cases} \tag{5}$$

where a_0 to c_1 are technology and design specific coefficients.

Power consumption of a SEFF also changes with w. This is due to the fact that increasing the window size is performed by increasing the size or the number of inverters in the delayed clock path. Both methods for altering w result in an *increase* in the power consumption of the SEFF. Power consumption is a monotonically increasing function of window size, as shown in Figure 3 for the master-slave flip-flops. The discontinuities (jumps) in the curve are due to a change in the number of inverters in the delay path. From this figure, one can conclude that the power dissipation of the SEFF may be approximated as a quadratic function of the transparency window width, i.e.,

$$P_{FF,i} = d_2 w_i^2 + d_1 w_i + d_0 \tag{6}$$

where d_0 to d_2 are technology and design specific coefficients.

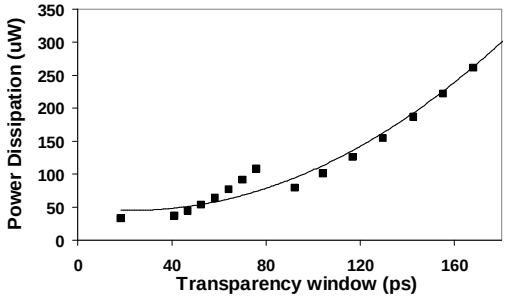

Figure 3. Power consumption of a SEFF as a function of transparency window

3. POWER OPTIMAL PIPELINE DESIGN

The key idea for using SEFF's in a pipeline circuit is that some positive slack may be available in one or more stages of the pipeline. Utilizing SEFF allows passing this slack to more timing critical stages of the pipeline to provide them with more freedom in power optimization through voltage scaling. As an example, consider the three stage pipeline circuit of Figure 4 operating at a supply voltage level of V_{DD}. The per-stage maximum logic delays are shown in the figure. Let's assume the setup time, hold time, and the clock-to-q delay of all (hard-edge) FF's are 30ps each. Assuming fixed and uniform time allocation across the three pipeline stages, from equation (1), it is easily seen that the minimum clock period is 560ps. If T_{clk}=560ps, no slack will be available to the first stage of the pipeline, and consequently, the supply voltage of the pipeline circuit cannot be scaled down in order to reduce the power consumption. However, if FF1 is replaced with a SEFF with a transparency window of 50ps, available slack at the second stage is passed to the first stage, providing the first stage with 50ps of borrowed time. Now since positive slacks are available in all stages of the pipeline, the circuit can be powered with a smaller supply voltage in order to reduce the power consumption (ideally, V_{DD} may be reduced by approximately 10%, resulting in roughly 19% power saving).

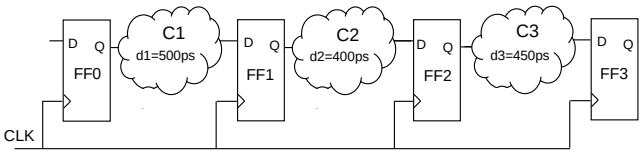

Figure 4. Example of slack passing

3.1 Soft-Edge Flip-Flop Modeling

To *optimally* select the transparency window of the SEFF's and choose the minimum supply voltage level, we need to accurately account not only for the effect of the transparency window on the setup/hold times and clock-to-q delay, but also for the power consumption of the SEFF's. In Section 2.2 it was shown that for a SEFF, the setup/hold times and clock-to-q delay can be modeled as linear functions of transparency window size (c.f. equation set (5)). If the supply voltage of the flip-flop can also be adjusted to a new voltage level, v, then coefficients of these linear models will become voltage-dependent parameters, i.e.,

$$\begin{cases} t_{s,i}(w_i, v) = a_1(v)w_i + a_0(v) \\ t_{h,i}(w_i, v) = b_1(v)w_i + b_0(v) \\ t_{cq,i}(w_i, v) = c_1(v)w_i + c_0(v) \end{cases} \quad (7)$$

Figure 5 through Figure 7 show SPICE simulations of the setup time, hold time, and clock-to-q delay as functions of the transparency window size and supply voltage level for the SEFF of Figure 2. From these figures one can see that the equation set (7) is quite accurate. Similarly, an extension of (6) can be used to model the effect of adjusting the supply voltage level, v, on the SEFF power consumption as:

$$P_{FF,i} = d_2(v)w_i^2 + d_1(v)w_i + d_0(v) \quad (8)$$

where $d_0(v)$ through $d_2(v)$ are voltage-dependent parameters.

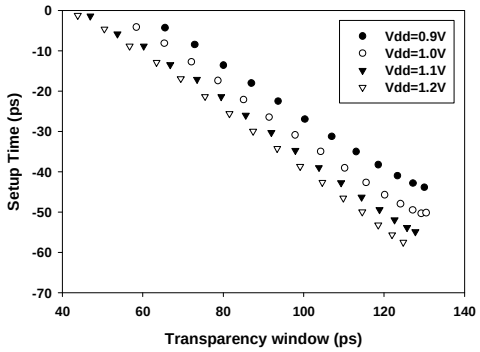

Figure 5. Setup time as a function of the supply voltage level and the transparency window width

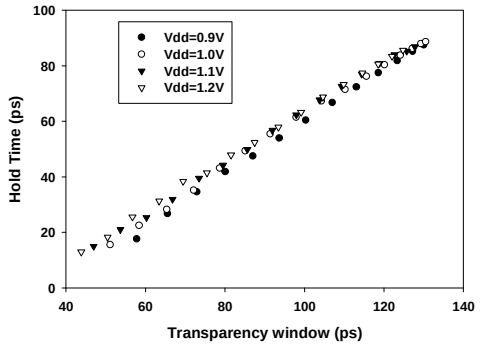

Figure 6. Hold time as a function of the supply voltage level and the transparency window width

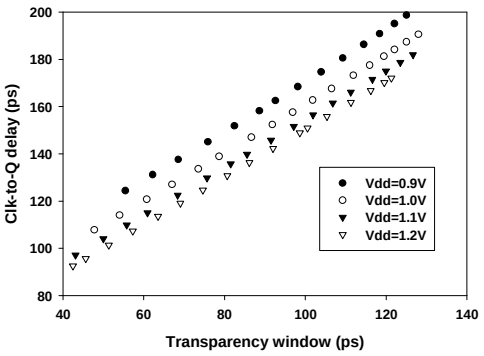

Figure 7. Clock-to-q delay as a function of the supply voltage level and the transparency window width

3.2 Combinational Logic Block Modeling

As a result of voltage scaling, for a fixed clock frequency, the total power consumption of combinational logic changes as follows[1]:

$$P_{Comb,i}(v) = \left(\frac{v}{V_0}\right)^2 P_{dyn,i} + \left(\frac{v}{V_0}\right)^3 P_{leak,i} \quad (9)$$

[1] This super-linear dependency of leakage power on the supply voltage is due to the combined effect of drain induced barrier lowering and the off-state leakage equation: $V_{DD} \times I_{OFF}$. The cubic form of this dependency has been empirically observed from SPICE simulations.

where $P_{dym,i}$ and $P_{leak,i}$ are the dynamic and leakage power consumption of the combinational logic at the nominal supply voltage V_0, and $P_{Comb,i}$ is the total power consumption of the combinational logic at the new supply voltage level v. On the other hand, it is known that when the supply voltage of a combinational logic is changed, its new delay can be obtained from the alpha-power law [8]; therefore,

$$d_i(v) = \left(\frac{V_0 - V_t}{v - V_t}\right)^{\alpha} d_i(V_0) \quad (10)$$

$$\delta_i(v) = \left(\frac{V_0 - V_t}{v - V_t}\right)^{\alpha} \delta_i(V_0) \quad (11)$$

where α is a technology parameter which is around 2 for long channel devices and 1.3 for short channel devices, and V_t denotes the magnitude of the threshold voltage of transistors.

3.3 Delay Elements

From equation (4) and Figure 6, one can see that increasing the transparency window of the i^{th} soft-edge FF-set puts more rigid constraint on the hold time condition for the i^{th} stage of the pipeline. Therefore, if needed, delay elements may be utilized in the minimum-delay path(s) to alleviate the hold time constraint violation. Similar to the delayed clock path, this is achieved by utilizing some inverters and appropriately sizing them in a similar fashion to [9], in order to meet the desired delay lower bound while incurring minimum power loss. The power overhead of a delay element is denoted as $P_{DE}(z,v) = k(v) \cdot z$, where z is the desired delay and $k(v)$ is a voltage dependent parameter.

3.4 Problem Formulation

The problem of power-optimal soft linear pipeline (PSLP) design is defined as finding optimal values of the supply voltage level for the whole design and the transparency windows of the individual soft-edge FF-sets in the design so as to minimize the total power consumption of an N-stage pipeline circuit subject to setup and hold time constraints:

$$\begin{cases} Min.\ \ P_{total} = \sum_{i=1}^{N} P_{Comb,i}(v) + \sum_{i=1}^{N-1} P_{FF,i}(w_i, v) + \sum_{i=1}^{N} P_{DE,i}(z_i, v) \\ s.t.\ (I)\ \ d_i(v) \le T_{clk} - t_{s,i}(w_i, v) - t_{cq,i-1}(w_{i-1}, v);\ 1 \le i \le N \\ \quad (II)\ \ \delta_i(v) + z_i \ge t_{h,i}(w_i, v) - t_{cq,i-1}(w_{i-1}, v);\ 1 \le i \le N \\ \quad (III)\ w_{\min} \le w_i \le w_{\max};\ 1 \le i \le N-1 \\ \quad (IV)\ \ v \in \{V_0, V_1 ..., V_{m-1}\} \end{cases} \quad (12)$$

where $P_{Comb,i}$, $P_{FF,i}$, and $P_{DE,i}$ are respectively the power dissipation of the combinational logic, FF's, and delay elements in the i^{th} stage of the pipeline. The first and second sets of constraints in (12) are respectively the setup and hold time constraints in the pipeline stages, the third set of constraints imposes an upper bound and a lower bound on the transparency window of the flip-flop ($w_{\min} \ge 0$ and $w_{\max} < 1/2T_{clk}$), and finally, the last constraint in (12) enforces the supply voltage of the pipeline to be from the set of available voltages $\{V_0, V_1, ... V_m\}$, where V_0 is the nominal supply voltage and $V_0 > V_1 ... > V_{m-1}$. Note that problem formulation (1) has $2N$ optimization variables corresponding to $N-1$ transparency window sizes, w_i, for the $N-1$ soft-edge FF-sets in the linear pipeline, N delay element values, z_i, for the N stages of the pipeline, and one supply voltage variable setting, v.

To solve (12) efficiently, we enumerate all possible values for v, and for each fixed v we solve a quadratic program (i.e., we minimize a quadratic cost function subject to linear inequality constraints), which can be solved optimally in polynomial time. In the fixed supply voltage PSLP problem formulation, $P_{Comb,i}$ terms drop out of the cost function, constraint (IV) disappears, and all other timing and power parameters become only dependent on w_i and z_i variables. We refer to this version of the problem as PSLP-FV, PSLP with fixed voltage.

Lemma 1: In the optimal solution of PSLP-FV design problem, the transparency window of the i^{th} soft-edge FF-set is exactly equal to the time borrowed by the combinational logic in the i^{th} stage of the linear pipeline.

Proof: According to the discussion in Section 2.2 and Figure 3, the power consumption of a SEFF is a monotonically increasing function of the transparency window size while its setup time is a decreasing function of the same. Now, from condition (I) in the PSLP-FV problem formulation of equation (12), a minimum decrease in the setup time of the i^{th} soft-edge FF-set $t_{s,i}(w_i, v)$ which meets the long-path constraint in the i^{th} stage of the linear pipeline, will produce the minimum increase in the power dissipation of the i^{th} soft-edge FF-set $P_{FF,i}(w_i, v)$. Therefore, the optimal solution is achieved by utilizing the smallest possible transparency window sizes which prevent setup time violation. ■

Lemma 2: In the optimal solution of PSLP-FV design problem, the delay element inserted in the i^{th} stage of the linear pipeline is exactly equal to the minimum extra time needed to meet the hold time constraint at the i^{th} soft-edge FF-set.

Proof: According to the discussion in Section 3.3, the power consumption of a delay element is a monotonically increasing function of the target delay value while the hold time of a SEFF is an increasing function of the same. Now, from condition (II) in the PSLP-FV problem formulation, a minimum delay value z_i added to the i^{th} stage of the linear pipeline which meets the short-path constraint for that stage, will produce the minimum increase in the power dissipation of the combinational logic in the i^{th} $P_{DE,i}(z_i, v)$.

Therefore, the optimal solution is achieved by utilizing the smallest possible delay elements which prevent hold time violations. ■

Theorem 1: The optimal solution to PSLP design problem is obtained by solving the PSLP-FV design problem m times for each distinct voltage level and selecting the voltage level v^* and the corresponding w_i^* and z_i^* values that minimize the total power dissipation for v^*.

Proof: This easily follows from the observation that solution of the PSLP-FV problem produces w_i's and z_i's for each possible v and we enumerate over all v's to get the global optimum solution in an exhaustive manner. ■

Finally we point out that a greedy solution to PSLP-FV whereby each pipeline stage is allocated a total combinational delay equal to the average combinational delay of all stages and the difference between actual delay of the stage and the allocated delay is

corrected for by setting the transparency window size of the corresponding soft-edge FF's, cannot meet the long-path constraints in all stages of the pipeline since the macro model equations for the setup/hold time and clock-to-q delays of the soft edge FF's have different slopes with respect to w_i 's.

4. EXPERIMENTAL RESULTS

To solve the mathematical problem developed in this paper, MOSEK optimization toolbox [10] has been used. To extract the parameters used in the optimization problem, we performed transistor-level simulations on soft edge flip-flops by using HSPICE [11]. The technology used in this simulation is a 65nm predictive technology model [12], the nominal supply voltage of this technology is 1.2V, and the die temperature is 100°C.

We synthesized a number of linear pipeline circuits which capture the characteristics of a typical pipeline in a modern processor as a set of benchmark circuits. SIS [13] optimization package was used to synthesize the set of benchmarks. The minimum and maximum delays of each pipeline stage were computed at the maximum allowed supply voltage (1.2V) and at the low and high temperature corners. The minimum clock cycle time for the pipeline (maximum frequency) and power dissipation of the linear pipeline were subsequently computed. This data defined the baseline for our comparison. Next, PSLP was run on each circuit under the condition that we maintain the clock frequency, while exploiting time borrowing across different stages to enable voltage scaling, and thus, power saving. The specifications of these benchmarks are shown in Table 1. The first column in this table gives the name of the benchmark, the second column reports the max and min delays of each stage of the pipeline at the nominal voltage, whereas the last column provides the clock frequency.

Table 1. Specification of the benchmark

Test-bench	Stage delays at nominal voltage (ps)	Clock freq.
TB1	(320,140), (332,150), (308,150), (320,170)	2.0GHz
TB2	(320,140), (332,150), (308,150), (280,145), (320,170)	2.0GHz
TB3	(325, 150) (310,155) (219,160)	2.0GHz
TB4	(275,40), (235,40), (245,60), (275,50), (275,70)	2.5GHz
TB5	(310,100), (245,40), (245,50), (245,60)	2.5GHz

Experimental results on these benchmarks are provided in Table 2. The first entry in the table is the name of the benchmark and the second entry shows the percentage power reduction achieved by PSLP (compared to conventional way of using hard-edge FF's in the pipeline). From this table, one can see that PSLP, which combines time borrowing and voltage scaling to reduce the power consumption, produces circuits with much lower power consumption at the same clock frequency. The supply voltage level and soft-edge FF-set transparency window sizes are reported in the last two columns of the table. Notice that for the first entry of the table, the window sizes are such that the first and second stages borrow larger times from their next stages, while the third stage cannot borrow much time; the reason is that since the last stage of the pipeline has a large max delay and ends up into a hard edge FF-set, it can lend very little time to its previous stage.

In another set of experiments, we studied how using SEFF's can improve the performance of a pipeline. In these experiments, the supply voltage of each pipeline was set at the nominal value and PSLP has been invoked for different values of T_{clk}. A binary search has been used to find the minimum T_{clk} for which PSLP has a solution. Table 3 shows that utilizing SEFF in the FF-set of pipelines improves the performance by an average of 12.8%. The area overhead of our technique is very small because it only replaces standard flip-flops with SEFF's when helpful. The circuit structure of the SEFF's is different from that of conventional FF's only in that SEFF's use an additional delay element (e.g., chain of inverters). The area overhead of this delay element is small compared to the area of the original FF. In addition, compared to the size of the combinational circuit plus the original FF-sets, the area overhead of the added delay elements inside SEFF's is miniscule. Consequently, in the final physical layout of the circuit, PSLP does not introduce any significant additional area. The runtime of our algorithm for all benchmarks is less than one second on a 2.4GHz Pentium-4 PC with 2GB of memory.

Table 2. Power reduction in PSLP compared to regular FF pipeline.

TB	Power reduction (%)	Optimum Vdd (V)	Optimum window size (ps)
TB1	32.1	1.0	40, 49, 22
TB2	33.8	1.0	40, 49, 46, 21
TB3	48.1	0.95	70, 24
TB4	16.3	1.10	35, 35, 30
TB5	25.4	1.05	37,36

Table 3. PSLP's performance improvement results

TB	Performance improvement (%)
TB1	14%
TB2	15%
TB3	20%
TB4	5%
TB5	10%

4.1 A Case Study

In order to demonstrate the efficacy of the proposed technique and provide insight as to how it operates, in this section, we provide details of applying our technique for performance/power optimization of a 34-bit pipelined adder. We used the PSLP design technique to determine the best way of pipelining this adder into four stages in order to achieve the maximum performance and also minimum power dissipation at that performance level. Assuming ripple carry adder (RCA) structure for the circuit, splitting the 34-bit adder can be done by including different number of cascaded 1-bit full adders in each stage of the pipeline. For example, a possible configuration is to build three stages of eight 1-bit full adders and one stage of ten 1-bit full adders, resulting in the 8–8–8–10 pipeline configuration. If hard-edge FF's are used in the pipeline, the minimum clock period of the 8–8–8–10 pipelined adder is 475ps under a supply voltage of 1.2V (the delay of a single full adder is 38.5ps and the setup time and clock-to-q delay are 35ps and 50ps, respectively). This delay can be reduced to 450ps by utilizing soft edge flip-flops.

The PSLP design technique can choose the minimum power and the fastest design among all possible configurations. Table 4 compares four pipeline structures for the 34-bit adder operating in the same supply voltage. In this table, all designs have three stages of eight 1-bit full adders, and a stage of ten 1-bit full adders.

Placing the 10-bit stage in the pipeline is critical in performance and power consumption of the circuit. In the 10–8–8–8 configuration a higher clock frequency can be achieved by means of time borrowing between stages, resulting in lower power consumption. The 8–8–8–10 needs a higher clock period, because time borrowing is not possible for the last stage, and therefore it needs more time. Another pipeline configuration is to have two 9-bit ripple carry adders and two 8-bit ripple carry adders. In this case, the performance is only a little worse than the 10-8-8-8 configuration. The PSLP design technique finds the optimal window assignment to each inter-stage flip-flop to optimally satisfy the timing constraints for the given clock period.

Table 4. Comparing performance of pipeline configurations

Configuration	Vdd (V)	Min clock period (ps)	Power consumption (mW)
10–8–8–8	1.2	450	6.42
8–10–8–8	1.2	472	6.50
8–8–10–8	1.2	472	6.51
8–8–8–10	1.2	486	6.55
9–9–8–8	1.2	455	6.42
9–8–9–8	1.2	433	6.51

Assuming a clock frequency of 2GHz, we will have a 500ps clock cycle which creates positive slack in the stages. This slack allows us to scale down the supply voltage. Reducing the voltage level decreases the power consumption by a noticeable amount due to the quadratic dependency of power on voltage. Moreover, by using the flexibility that the SEFF's add to the pipeline, voltage can be further reduced to save even more power. The PSLP technique searches for the minimum power consumption by changing the operating voltage and finding optimum window size assignment for that voltage. Table 5 lists the optimum operating voltage and minimum power consumption of four different configurations. For instance, in the case of 10-8-8-8 adder, PSLP suggests a window of 47ps for the first stage and 42ps for the next two soft edge stages to meet the 2GHz constraints under a supply voltage of 1.05volts.

Table 5. Minimum power consumption of pipeline configurations

Configuration	Optimum Vdd (V)	Power consumption (mW)	Clock frequency
10–8–8–8	1.05	4.9	2GHz
8–10–8–8	1.15	5.1	2GHz
9–9–8–8	1.05	4.9	2GHz
9–8–8–9	1.10	4.9	2GHz

5. CONCLUSION

We presented a new technique to minimize the total power consumption of a linear pipeline circuit by utilizing soft-edge flip-flops and choosing the optimal supply voltage level for the pipeline. We formulated the problem as a mathematical program and solved it efficiently. Our experimental results demonstrated that this technique is quite effective in reducing the power consumption of a pipeline circuit under a performance constraint.

A number of extensions to the work presented in this paper are possible. One is to allow different transparency windows for FF's in the same FF-set. The only difference is that in this case the setup and hold time constraints should be satisfied for every I/O conduit of the circuit (see [14] for an exact definition). The maximum number of I/O conduits in any stage of linear pipeline is the product of the cardinality of its input FF-set and its output FF-set. It is seen that the size of PSLP design problem for a this case still remains manageable. Another extension is to consider the interdependency between setup and hold times. It is known that the "independent" characterization of setup, hold time, and clock-to-q delay of FF's results in pessimistic timing analysis [15]. In our problem definition, considering the interdependency between the setup and hold time provides more freedom in the optimization problem and it is expected to improve the quality of the results. Yet another extension is to solve the PSLP design problem for the nonlinear pipelines, i.e. pipelines that perform variable functions and have multi-stage feed-forward paths or multi-stage feedback paths [16]. The problem setup in this case will be similar to that of Section 3 but the constraints are more complex. Finally one may combine our technique with clock skew control and retiming methods [17] to achieve higher power savings.

REFERENCES

[1] S. Manne, A. Klauser, and D. Grunwald, "Pipeline gating: speculation control for energy reduction," *Proc. of International Symposium on Computer Architecture*, 1998, pp. 132-141.

[2] H. M. Jacobson, "Improved clock-gating through transparent pipelining," *Proc. of International Symposium on Low Power Electronics and Design*, 2004, pp. 26-31.

[3] H. Jacobson, P. Bose, H. Zhigang, *et al.*, "Stretching the limits of clock-gating efficiency in server-class processors," *Proc. of High-Performance Computer Architecture*, 2005, pp. 238-242.

[4] D. Ernst, N. Kim, S. Das, *et al.*, "Razor: a low-power pipeline based on circuit-level timing speculation," *Proc. of International Symposium on Microarchitecture*, 2003, pp. 7-18.

[5] H. Partovi, R. Burd, U. Salim, *et al.*, "Flow-through latch and edge-triggered flip-flop hybrid elements," *Proc. of International Solid-State Circuits Conference*, 1996, pp.138-139.

[6] D. Harris and M. A. Horowitz, "Skew-tolerant domino circuits," *IEEE Journal of Solid-State Circuits*, vol. 32, no. 11, Nov. 1997, pp. 1702-1711.

[7] V. Joshi, D. Blaauw, and D. Sylvester, "Soft-edge flip-flops for improved timing yield: design and optimization," *Proc. of International Conference on Computer-Aided Design*, 2007, pp. 667-673.

[8] T. Sakurai and A. R. Newton, "Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas," *IEEE Journal of Solid-State Circuits*, vol. 25, no. 2, Apr. 1990, pp. 584-594.

[9] B. Amelifard, F. Fallah, and M. Pedram, "Low-power fanout optimization using MTCMOS and multi-Vt techniques," *Proc. of International Symposium on Low Power Electronics and Design*, 2006, pp. 334 -337.

[10] MOSEK Optimization Software, http://www.mosek.com [online]

[11] HSPICE: The gold standard for accurate circuit simulation, http://www.synopsys.com/products/mixedsignal/hspice/hspice.html [online]

[12] Predictive Technology Model, http://www.eas.asu.edu/~ptm/

[13] E. M. Sentovich, K. J. Singh, L. Lavagno, *et al.*, "SIS: A System for Sequential Circuit Synthesis," University of California, Berkeley, Report M92/41, May 1992.

[14] C.-S. Hwang and M. Pedram, "PMP: Performance-driven multilevel partitioning by aggregating the preferred signal directions of I/O conduits," *Proc. of Asia and South Pacific Design Automation Conference*, 2005, pp. 428-431.

[15] E. Salman, A. Dasdan, F. Taraporevala, *et al.*, "Exploiting setup–hold-time interdependence in static timing analysis," *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, vol. 26, no. 6, Jun. 2007, pp. 1114-1125.

[16] K. Hwang, *Advanced Computer Architecture*. New York, NY: McGraw Hill, 1993.

[17] J.Monterio, S. Devadas, and A. Ghosh. "Retiming sequential circuits for low power" In Digest of *Technical Papers of the 1993 IEEE International Conference on CAD*, pages 398-402, November 1993.