International Journal of Electronics
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tetn20

A novel productivity-driven logic element for field-programmable devices

Thomas Marconi [a], Koen Bertels [b] & Georgi Gaydadjiev [b]
[a] School of Computing, National University of Singapore, Singapore, Singapore
[b] Computer Engineering Lab, Delft University of Technology, Delft, The Netherlands
Published online: 09 Jul 2013.

To cite this article: International Journal of Electronics (2013): A novel productivity-driven logic element for field-programmable devices, International Journal of Electronics, DOI: 10.1080/00207217.2013.794489

To link to this article: http://dx.doi.org/10.1080/00207217.2013.794489

International Journal of Electronics, 2013
http://dx.doi.org/10.1080/00207217.2013.794489

A novel productivity-driven logic element for field-programmable devices

Thomas Marconi[a]*, Koen Bertels[b] and Georgi Gaydadjiev[b]

[a]*School of Computing, National University of Singapore, Singapore, Singapore;* [b]*Computer Engineering Lab, Delft University of Technology, Delft, The Netherlands*

(*Received 13 August 2012; final version received 3 February 2013*)

Although various techniques have been proposed for power reduction in field-programmable devices (FPDs), they are still all based on conventional logic elements (LEs). In the conventional LE, the output of the combinational logic (e.g. the look-up table (LUT) in many field-programmable gate arrays (FPGAs)) is connected to the input of the storage element; while the D flip-flop (DFF) is always clocked even when not necessary. Such unnecessary transitions waste power. To address this problem, we propose a novel productivity-driven LE with reduced number of transitions. The differences between our LE and the conventional LE are in the FFs-type used and the internal LE organisation. In our LEs, DFFs have been replaced by T flip-flops with the T input permanently connected to logic value 1. Instead of connecting the output of the combinational logic to the FF input, we use it as the FF clock. The proposed LE has been validated via Simulation Program with Integrated Circuit Emphasis (SPICE) simulations for a 45-nm Complementary Metal–Oxide–Semiconductor (CMOS) technology as well as via a real Computer-Aided Design (CAD) tools on a real FPGA using the standard Microelectronic Center of North Carolina (MCNC) benchmark circuits. The experimental results show that FPDs using our proposal not only have 48% lower total power but also run 17% faster than conventional FPDs on average.

Keywords: logic element; reconfigurable devices; low power; field-programmable devices; reconfigurable computing; architecture

1. Introduction

Field-programmable devices (FPDs) are integrated circuits that can be (re)configured by their end users to implement various digital functions (Brown, Brown, & Rose, 1996). There are three main categories FPD: simple of programmable logic devices (SPLDs), complex PLDs (CPLDs) and field-programmable gate arrays (FPGAs) (Brown et al. 1996). Advantages of using FPDs in run-time reconfigurable systems are instant manufacturing turnaround, reduced start-up costs, low financial risk, short time-to-market and easy design changes (Brown et al. 1996). However to get these benefits, the users need to pay additional costs: higher power consumption (approximately 12× larger dynamic power), larger silicon areas (40× more area required) and lower operating speeds (3.2× slower), as compared to the Application-Specific Integrated Circuits (ASICs) (Kuon & Rose, 2006). Higher power consumption requires expensive packaging (Boemo, Rivera, López-Buedo, & Meneses, 1995; Gayasen, Lee, et al., 2004; Mondal & Memik, 2005b; shortens chip lifetimes (Boemo et al., 1995), asks for costly cooling systems (Boemo et al., 1995; Gayasen, Lee, et al., 2004; Mondal & Memik, 2005b), decreases system reliability (Mondal & Memik 2005b)

*Corresponding author. Email: marconi@comp.nus.edu.sg

and prohibits battery operations (Boemo et al., 1995; Gayasen, Lee, et al., 2004; Mondal & Memik, 2005b). Therefore, reducing the power consumption of FPDs is a critical issue.

Many techniques have been proposed for lowering the power consumed by FPDs. However, all existing power reduction techniques target what we call a "conventional logic element". This conventional logic element (LE) has been used by researchers of FPDs since it was patented by Birkaner and Chua in 1978 (Birkner and Chua, 1978). Although FPDs have been improved significantly since the original proposal, they still make use of a proposal dated 1978 that may need to be reconsidered. The conventional LE contains the combinational logic (e.g. the look-up table (LUT) in FPGAs) and the storage element (D flip-flop (DFF)). The output of the combinational logic is connected to the input of the storage element; the clock input of DFF is connected to the clock signal. Since the DFF clock input is connected directly to the clock signal, the DFF is always clocked even when this is not needed, i.e. unproductive activities. For example, when $D = Q$, the DFF does not need to be clocked. Such unnecessary transitions waste power in FPDs using the conventional LEs. This is related to the fact that even low-power flip-flops (FFs) consume power during logic transition from zero-to-zero and from one-to-one as shown in Lang, Musoll, and Cortadella (1997), Nogawa and Ohtomo (1998), Markovic, Nikolic, and Brodersen (2001), Zhao, Darwish, and Bayoumi (2002, 2004), Sayed and Al-Asaad (2006), and Teh et al. (2006). It is well known that the dynamic power consumption of a CMOS circuit is proportional to its clock frequency f, load capacitance C, activity factor α and the square of supply voltage V_{DD}^2. In the case that α is zero, many researchers (e.g. Lang et al., 1997; Markovic et al., 2001; Nogawa & Ohtomo, 1998; Sayed & Al-Asaad, 2006; Teh et al., 2006; Zhao, et al., 2002, 2004) have reported that a certain amount of dynamic power is still consumed by sequential circuits because the clock signal is continuously supplied. To model this phenomenon, Lang et al. (1997) define the average dynamic energy per cycle of a FF as

$$E = E_c \cdot N_c + E_0 \cdot N_0 + E_1 \cdot N_1 + E_{gb} \cdot N_{gb} + E_{ga} \cdot N_{ga},$$

where E_c is the energy consumed by one event during clock edges and during the real transition of the output Q when the FF changes, E_0 is the energy consumed by one event when flip-flop remains in state 0 (no transition in Q), E_1 is the energy consumed by one event when flip-flop remains in state 1 (no transition in Q), E_{gb} is the energy consumed by one event of glitch of Q before the leading edge of the clock, E_{ga} is the energy consumed by one event of glitch of Q after the leading edge of the clock and N_x is the corresponding average number of events per cycle. Lang et al. (1997) also demonstrated that a significant fraction of the energy is consumed when the FF output Q does not change. To solve this problem, we propose a novel *productivity-driven* LE for reduced FPDs power consumption. The proposed LE can be used in any kind of FPDs, such as SPLDs, CPLDs as well as FPGAs. The differences between our LE and the conventional proposal are in the FF-type and the LE internal organisation. Instead of using DFFs, we use T flip-flops (TFFs) with T input permanently at logic 1 (T = 1). This is related to the fact that designing sequential circuits using TFFs is more power efficient than DFFs as reported in Wu and Pedram (2001). The output of the combinational logic in our case is connected to the clock input of the FF. As a result, our LE is able to block unnecessary clock transitions without using additional clock gating logic, avoiding unproductive activities.

The Microelectronic Center of North Carolina (MCNC) benchmark circuits (Yang, 1991) are used to evaluate our proposal in 45-nm BSIM4 CMOS technology (Cao et al., 2010). We use LTSPICE tools shown in Technology (2008) for transistor-level circuit

simulations with nominal supply voltage VDD of 1.2 V. The evaluation is performed in terms of total power, logic power, clock power, interconnect power, dynamic power, static power, speed and LE area. Moreover, to complete the evaluation, we also evaluate our proposal using a real Computer-Aided Design (CAD) tool (Quartus II Compiler Tool, Altera Corporation, San Jose, CA) and a real FPGA from Altera by forcing the existing tool to implement circuits behaving like our proposed LE. In our previous conference paper Marconi, Theodoropoulos, Bertels, and Gaydadjiev (2010a), we only presented Hardware Description Language (HDL) coding style to reduce power consumption for FPGAs, no hardware architecture modification was performed. In this article, inspired by Marconi et al. (2010a), we move forward by presenting a new LE architecture for power-efficient FPDs.

The main contributions of this article are as follows:

(1) a novel *productivity-driven* LE for FPDs;
(2) an extensive evaluation using both transistor-level circuit simulations and a real FPGA and CAD tool experiments; and
(3) reduction of total power by 48% and improvement of performance by 17% compared to conventional FPDs on average.

The rest of the article is organised as follows. In Section 2, we review existing work in dealing with power consumption issue of FPDs. Our proposed LE to tackle power inefficiency of FPDs is presented in Section 3. In Section 4, we evaluate the proposal. Finally, we conclude in Section 5.

2. Related work

Before presenting our proposal of how to address power inefficiency in FPDs, in this section, we present existing published work in their attempts to address the same problems.

One of the ways to reduce dynamic power consumption in FPDs is to stop selected signals from flowing or propagating to the other part of the circuits, referred as stopping-signal-to-flow technique. Clock gating, one instance of this technique, is used to reduce dynamic power consumption by selectively blocking the circuit local clock when no state or output transition takes place. The clock gating controller is needed for detecting the conditions of the observed circuit. Based on these conditions, the clock gating controller can know the exact time when it can stop clock signal to be transported to the specific circuit for power saving. It is used in FPGAs as reported in Cadenas and Megson (2003), Sutter, Todorovich, and Boemo (2004), Zhang, Roivainen, and Mämmelä (2006), Khan (2006), Parlak and Hamzaoglu (2007), Achronix (2008), Huda, Mallick, and Anderson (2009), Wang, Gupta, and Anderson (2009), Klein (2009), Actel (2009), Rivoallon (2010), Saleem and Khan (2010), Sterpone, Carro, Matos, Wong, and Fakhar (2011), Hussein, Klein, and Hart (2012), Ravindra and Anuradha (2012) and CPLDs as presented in Semiconductor (2009). This technique is supported by commercial CAD tools from Xilinx (San Jose, CA) as reported in Rivoallon (2010). In Achronix (2008), an asynchronous FPGA with clock gating is proposed. Another example of this technique is the guarded evaluation. The idea is to stop some signals in a digital circuit that do not affect an output of the circuit for certain conditions as applied to FPGAs reported by Howland and Tessier (2008), Anderson and Ravishankar (2010). Similar to this technique, we stop clock signals to flow to no-need-clock LEs of FPDs. At least, there are three differences

from the conventional technique, such as the changing of LE architecture, the granularity of controlling and the unified path. There is no architecture changing of LE in conventional technique, while in our proposal, we change the architecture to make it much more suitable in order to stop unnecessary flowing of signals efficiently and easily. Different from the conventional technique that needs explicit controller to control the flow of signals (consuming power), our proposal implicitly builds this controller inside the circuits. Another difference from the conventional technique is the granularity of controlling. In our proposal, the granularity is at each single LE, while in the conventional one, it is at a group of LEs. In the conventional technique, the data path, the control path and the clock are separated, while in our proposal, we unify all of these paths into a single unified path.

Another technique to reduce dynamic power consumption is to change the FF. Conventional single-edge-triggered FFs respond only once per clock pulse cycle. To reduce power consumption, a FF that can respond to both the positive and the negative edge of the clock pulse (double-edge-triggered flip-flops) was proposed in Uuger (1981). This technique is used in Xilinx CPLDs (Xilinx 2002). Similar to this technique, we do change the FFs. Different from this technique, in our proposal, we change the FFs-type to T flip-flops with the *T* input permanently connected to logic value 1, while in conventional one, it still uses DFFs. Moreover, in our proposal, we do change the internal LE organisation. Instead of connecting the output of the combinational logic to the FF input, we use it as the FF clock.

Reducing interconnection is another technique to reduce power consumption in FPDs. The design that requires less interconnection consumes less static and dynamic power consumptions. Modern FPGAs contain embedded hardware blocks, such as multipliers, Digital Signal Processings (DSPs) and memories. It is reported in Kuon and Rose (2006), Klein (2009) that mapping designs to these blocks requires less interconnection. Building circuits with bigger size LUTs needs less interconnection between LUTs. This has triggered commercial FPGA vendors to use bigger-sized LUTs as reported in Klein (2009), Altera (2006a). Using a diagonally symmetric interconnect pattern in Virtex-5 FPGAs (Xilinx Inc., San Jose, CA) can reduce the number of interconnect routing hops as reported in Douglass (2006). Consequently, the interconnect power consumption is reduced. Constraining designs to be implemented on the specific regions within the FPGA to minimise power consumed by clock networks is reported in Wang, French, Davoodi, and Agarwal (2006). The idea is to place logic closer together for minimising the clock network usage. In our proposal, we do not make an effort to reduce the usage of interconnection resources; however, we do reduce interconnection power by minimising the glitches flowing to the interconnect network.

Another technique to reduce dynamic power consumption in FPDs is to reduce glitches. This is related to the fact that a circuit with fewer glitches consumes less dynamic power. Adding programmable delay circuits into configurable logic blocks of FPGAs is reported in Lamoureux, Lemieux, and Wilton (2008). The generation of glitches is avoided by aligning the arrival times of signals using the proposed programmable delay circuits. As a consequence, the glitches are reduced. Pipelining FPGA circuits to reduce the number of levels of the circuits between registers is reported in Wilton, Shin Ang, and Luk (2004), Bard and Rafla (2008), Rollins and Wirthlin (2005). As a result, circuits tend to produce fewer glitches. Inserting negative-edge-triggered flip-flops at the outputs of selected LUTs to block glitches for propagating further in FPGAs is demonstrated in Czajkowski and Brown (2007). Retiming can be used to reduce glitches in FPGAs (Fischer, Buchenrieder, and Nageldinger 2005). The idea is to redistribute registers

along a signal path without changing the functionality of the circuit. By doing so, the logic between registers is minimised, hence reducing glitches. A routing that can balance arrival times of the inputs of the same LUTs in FPGAs is proposed in Dinh, Chen, and Wong (2009). By doing so, the glitches are reduced. In our proposal, we reduce glitches produced by unnecessary clockings to the FFs.

Reducing switching activity is another technique to reduce dynamic power consumption in FPDs. Since power consumption depends linearly on the switching activity, reducing this results in power consumption improvement. Reordering input signals to LUTs can reduce dynamic power consumption in FPGAs. By doing so, we can minimise the switching activity inside LUTs as reported in Alexander (1997). Choosing state encoding of FSM to minimise the bit changes during state transitions for reducing switching activity in FPGAs is reported in Sutter, Todorovich, Lpez-Buedo, and Boemo (2002b), Mengibar, Entrena, Lorenz and Sánchez-Reillo (2003). Logic synthesis in FPGAs is a process of transforming a given design (coded in schematic or HDL) into a gate-level circuit. Minimising switching activity during logic synthesis for FPGAs is presented in Tinmaung, Howland, and Tessier (2007). Clock scaling is an approach to reduce switching activity by adjusting operating clock frequency dynamically. Applying this approach in FPGAs is reported in Paulsson, Hübner and Becker (2008). In our work, we reduce the switching activity by avoiding unproductive activity caused by unnecessary clockings to LEs.

Another technique to reduce power consumed by FPDs is to turn off not-needed circuits. Power gating is a technique for reducing power consumption by temporarily turning off circuits that are not in use. It is applied in FPGAs (Gayasen, Tsai, et al., 2004; Hassan, et al., 2005; Ishihara, Hariyama, & Kameyama, 2010; Nair, Koppa, & John, 2009; Rahman, Das, Tuan, & Trimberger, 2006). This technique is used in industrial products, such as Atmel PLDs (Atmel, 2000), Altera CPLDs (Altera, 2006b), Actel FPGAs (Actel, 2008), QuickLogic FPGAs (QuickLogic, 2008), Xilinx FPGAs (Klein, 2009), Altera FPGAs (Khan, 2007). In Ishihara (2010), an asynchronous FPGA with autonomous fine-grain power gating is proposed. How to partition a design to benefit better from power gating technique is reported in Hassan et al. (2005). Dividing a finite state machine (FSM) into two smaller sub-FSM using a probabilistic criterion is reported in Sutter et al. (2002a).The idea is to activate only one sub-FSM at a time; meanwhile, the other is disabled for power reduction. Since not all inputs of LUTs are used in real FPGA designs, leakage power can be reduced by shutting-off Static Random-Access Memory (SRAM) cells and transistors associated with unused LUT inputs as reported in Mondal and Memik (2005b). Resources used by hardware tasks implemented on partially reconfigurable FPGAs cannot be turned off after configuration, consuming leakage power. Therefore, hardware tasks need to be operated as soon as possible after configuration in run-time systems using partially reconfigurable devices. This technique for leakage power reduction in FPGAs is reported in Li, Yuh, Yang, and Chang (2007). Similar to this technique, we also turn off not-needed resources for power saving.

Optimising the usage of bits to represent the processed data is another technique to reduce power consumption in FPDs. The bit-widths of the internal signals of circuits can be optimized to reduce dynamic power consumption. A circuit with shorter bit-widths consumes less power. This approach applied in FPGAs is reported in Gaffar, Clarke, and Constantinides (2006), Constantinides (2006). Different from this technique, while this technique optimise the usage of signals at data path, in our proposal, the usage of clock signals is optimised.

A simple technique to reduce power consumptions in FPDs is to manipulate the voltage (e.g. reducing the supply voltage or the voltage swing, increasing threshold

voltage) either statically or dynamically, called here as voltage-manipulation technique. Since there is a quadratic relationship between supply voltage and dynamic power, reducing the voltage will significantly reduce the dynamic power. Moreover, a cubic relationship between supply voltage and leakage power reduces significantly the leakage power. Since the dynamic power consumption is linearly proportional to the voltage swing, power is reduced by minimising the voltage swing. A lower threshold voltage transistor runs faster, but it consumes more power. By increasing this voltage intelligently at certain part of the circuits, the power can be reduced without suffering performance degradation. Klein (2009), Jenkins and Ekas (2006) reported the usage of this technique in the lowering supply voltage in marketable FPGAs. Powering FPGAs with a variable supply voltage is reported in Chow, Tsui, Leong, Luk, and Wilton (2005). This method is referred as dynamic voltage scaling (DVS). Powering FPGAs with two different supply voltages (dual-Vdd) is reported in Gayasen, Lee, et al. (2004), Mukherjee and Memik (2005), Li, Lin, et al. (2004), Lin and He (2006), Li et al. (2004a), Hu, Lin, He, and Tuan (2008). It is to use lower supply voltages on noncritical paths to reduce power, and higher supply voltages on critical paths to maintain performance. Algorithms for Vdd assignment are presented in Lin and He (2006), Li et al. (2004a). Hu et al. (2008) combines concurrently this technique with retiming to better reduce power consumption in FPGAs. Using LUTs with two different operating modes (high performance and low power) reduces leakage power as reported in Azizi and Najm (2005). The idea is to use some transistors for lowering supply voltage across input inverters of LUTs during low-power operation mode. Since not all LUTs need to be operated in high-performance mode, the leakage power is reduced. Choosing the best operating mode for each memory on FPGAs based on prior knowledge of its dead intervals is reported in Meng, Sherwood, and Kastner (2006) to reduce leakage power consumption. The memory can be operated in three operating modes: active, drowsy and sleep. The sleep mode is a condition when the power supply is disconnected to the memory, whereas the drowsy mode is a condition when the memory is connected to a lower supply voltage. The idea is to operate the memory based on its dead intervals. The memory with long/medium/short dead interval is operated on sleep/drowsy/active mode. To reduce interconnect power, low-voltage swing interconnects are applied for FPGAs in George, Zhang, and Rabaey (1999), Matsumoto and Masaki (2005). Because this technique degrades the performance, in George et al. (1999), the dual-edge-triggered FFs are used to handle this degradation. Applying low swing interconnects only on non critical paths is proposed in Matsumoto and Masaki (2005) to reduce the performance degradation of this technique. Multi-threshold voltage technique is to use higher threshold voltage transistors on noncritical paths to reduce static power, and low threshold voltage transistors on critical paths to maintain performance. This technique has been applied in industrial FPGAs as reported in Klein (2009), Jenkins and Ekas (2006). Routing switches that can operate in three different modes, such as high speed, low power or sleep is reported in Anderson and Najm (2004). Using dual-Vdd-dual-Vt routing switches for reducing interconnect power is demonstrated in Mondal and Memik (2005a). Applying dual-vdd and power gating techniques for routing switches is proposed in Li (2004). Instead of supply voltage or swing voltage or increasing threshold voltage, in our proposal, we turn off not-needed resources.

Hardware sharing is another technique to reduce power consumption in FPDs. Modern FPGAs have the ability to reconfigure part of their resources without interrupting the remaining resources at run-time. Hardware sharing can be realised by utilising this partial reconfiguration feature for power consumption reduction. Power saving using this approach in FPGAs is reported in Patterson (2000), Park and Burleson (1998), Tessier,

Swaminathan, Ramaswamy, Goeckel, and Burleson (2005), Noguera and Kennedy (2007), Klein (2009), Kao (2005), Becker, Hübner, and Ullmann (2006). While this technique focuses on scheduling the hardware tasks to benefit from hardware sharing, in our work, we schedule the flow of clock signals to LEs.

Another technique to reduce power consumed by FPDs is to lower circuit capacitance. A lower capacitive circuit consumes less dynamic power. One of the ways to reduce capacitance is to use a low-k dielectric material. This technique is used by commercial FPGAs as shown in Klein (2009), Jenkins and Ekas (2006). In our proposal, clock signals only drive the productive LEs, while the conventional ones drive all LEs at the same time.

Optimising power consumption using CAD tools is another technique to reduce power absorbed by FPDs. During high-level synthesis(HLS), a circuit can be implemented by combining functional units, such as multipliers, adders, multiplexers, etc. Each functional unit can be realised using one of the varied implementations. Each implementation requires a certain area and runs at a specific speed with required power consumption. To reduce power consumption, we need to choose the best design for a given circuit that can meet the timing requirement with minimal power. HLS algorithms for minimising power consumption in FPGAs are reported in Wolff, Knieser, Weyer, and Papachriston (2000), Chen, Cong, and Fan (2003). Technology mapping in FPGAs is a process of transforming a given circuit into a circuit that only consists of LUTs. The way we map circuits into FPGAs can affect the power consumption. The algorithms to perform this process for power reduction are presented in Anderson and Najm (2002), Chen, Cong, Li, and He (2004), Wang, Liu, Lai, and Wang (2001), Farrahi and Sarrafzadeh (1994), Li, Chen, He, and Cong (2003). The main idea is to pack nodes with high switching activity inside LUTs. By doing so, we can minimise power needed to transport signals of nodes among LUTs. To better estimate the switching activity, glitches are considered during technology mapping in Cheng, Chen, and Wong (2007). Transformation by changing the functionalities of LUTs with re-routing (Chen, Hwang, & Liu 1997) and without re-routing (Kumthekar, Benini, Macii, & Somenzi 2000) can be used to reduce power consumption in FPGAs. Chen et al. (1997) performs the transformation after technology mapping by reducing switching densities of the outputs of the LUTs, whereas Kumthekar et al. (2000) transforms the design after mapping, placement and routing by considering switching activity and capacitance at the outputs of the LUTs. Clustering logic blocks in FPGAs can affect reduction in power consumption. Clustering reduces the usage of interconnect resources. As a result, it reduces interconnect power. The optimal number of LEs per cluster for power reduction is 12 as reported in Li et al. (2003). The way we cluster a circuit into an FPGA can affect the power consumption. The clustering algorithms to reduce power consumption are presented in Singh and Marek-Sadowska (2002), Chen and Cong (2004). The main idea in Singh and Marek-Sadowska (2002) is to minimise intercluster connections for reducing interconnection power. Clustering for FPGAs with dual-Vdd is shown in Chen and Cong (2004). Assigning noncritical paths to clusters with low power supply voltage is the key idea of Chen and Cong (2004). Placement algorithms to reduce power consumption in FPGAs are presented in Gupta, Anderson, Farragher, and Wang (2007), Vorwerk, et al. (2008), Lamoureux and Wilton (2007). The main idea is to add estimated dynamic power into cost function of the placement algorithms. Dynamic power is thus reduced during placement. A placement algorithm that takes into account the cost of using clock network resources to reduce power consumed by clock network is reported in Lamoureux and Wilton (2007). Routing algorithms to reduce power consumption in FPGAs are reported in Gupta et al. (2007), Dinh et al. (2009). Assigning nodes with high switching activity to low-capacitance

routing resources is the main idea behind the routing algorithm for reducing interconnect power in Gupta et al. (2007). Combining power-aware technology-mapping, clustering, placement and routing algorithms to reduce FPGA power consumption is reported in Lamoureux and Wilton (2003). They found that the most contribution to the power reduction is from the clustering algorithm. In this work, the basic building block of FPDs, which is LE, is totally different from the conventional LE used in existing CAD tools. For that reason, the way to design digital circuits out of FPDs will be different. This needs changing in CAD tools. In this current work, we have not yet created automatic CAD tools, we do all the design by hands.

Another technique to minimise power consumption in FPDs is to change the configuration memory. The idea to replace SRAM memory with nonvolatile memory targeting FPGAs is presented in Actel (2008), Zhao, Belhaire, Chappert, and Mazoyer (2009), Onkaraiah et al. (2011), Koga et al. (2010). Since SRAM memory is volatile, SRAM-based FPGAs need to be reconfigured before usage. This reconfiguration consumes power. In contrast, FPGAs that use nonvolatile memory (e.g. the commercial Actel FPGA Actel (2008)) can be operated directly without reconfiguration. The leakage power consumed by an asymmetric SRAM cell depends on its stored data. Since 87% of the configuration memory cells in FPGAs store logic 0 in the real FPGA design Gayasen, Srinivasan, Vijaykrishanan, and Kandemir (2007), using asymmetric SRAM cells with low leakage at logic 0 for FPGAs to reduce leakage power consumed by reconfiguration memory is reported in Gayasen et al. (2007). The idea is to select polarities for logic signals (i.e. inverted or not) that can increase the number of zeros stored on the configuration memory. Since the number of zeros is increased, the number of memory cell that operates at low leakage is increased. Thus, the leakage power consumed by the reconfiguration memory is reduced. To reduce power during run-time reconfiguration, configuration memory with two different types of memories (Ramo, Resano, Mozos, & Catthoor 2006) or run-time configurable memory with two different modes (Wang, Miranda, Papanikohou, Catthoor, & Dehaene, 2005) is proposed. One type (mode) is optimised for high-speed operation, whereas the other type(mode) is optimised for low-power operation. Hardware tasks implemented on FPGAs that do not require high-speed reconfiguration can be reconfigured to the low power one for power saving during reconfiguration. In our work, there is no effort to change the configuration memory. We only change the basic LE that builds the FPDs.

Lowering leakage power is another way to reduce power consumed by FPDs. Since leakage power in multiplexers is dependent on their input states, selecting polarities for logic signals (i.e. inverted or not) so that the multiplexers are operated in low-leakage states in the majority of time can be used to reduce leakage power in FPGAs (Anderson, Najm, & Tuan 2004). To reduce more leakage power, the work in Anderson et al. (2004) is extended by Hassan et al. (2008). In Hassan et al. (2008), not only polarity is considered to achieve low leakage states, but also the order of input signals to LUTs is modified to have a better leakage power reduction. It is different from Alexander (1997) that targets dynamic power, the work in Hassan et al. (2008) targets static power by reordering input signals to LUTs. Since the leakage power is state dependent (Roy, Mukhopadhyay, & Mahmoodi-Meimand, 2003), changing this state results leakage power reduction. Using nanoelectromechanical relays for programmable routing in FPGAs is reported in Chen et al. (2010) to reduce power consumption due to its zero leakage and low on-resistance characteristics. Although it is more power efficient than the conventional FPGA, it is not suitable for run-time reconfigurable systems due to its large mechanical switching delay. Older generation FPGAs use dual-oxide process technology:

thick oxide transistors (slow transistors) for Inputs/Outputs (IOs) and thin oxide transistors (fast transistors) for core. To reduce leakage power in FPGAs, triple-oxide process technology is used in modern FPGAs (e.g. Virtex-4) Abusaidi and Philofsky (2008), Klein (2009). In these FPGAs, another type of transistors with medium thickness oxide is dedicated for the configuration memory and interconnect pass gates. Leakage power reduction is not the focus of our current work. Instead, we focus on reducing dynamic power consumption by introducing a new architecture for basic LE for FPDs.

In summary, although many techniques have been proposed for power reduction in FPDs as discussed in this section, they are all based on conventional LEs. In the conventional LE, the output of the combinational logic (e.g. the LUT in many FPGAs) is connected to the input of the storage element; while the DFF is always clocked even when it is not necessary. Such unnecessary transitions waste power. To address this problem, we propose a novel low-power LE as presented in Section 3.

3. The proposed logic element

The purpose of LEs in FPDs is to provide the basic programmable combinational logic and storage elements used in digital systems. An LE contains a combinational logic circuit generator (CLCG) and a storage element as shown in Figure 1. The CLCG is used for the combinational function, while the storage element is used for storing temporary results.

In conventional LEs, the output of CLCG is connected to the input of the storage element as illustrated in Figure 1(a). The storage element in the conventional LE is a DFF.

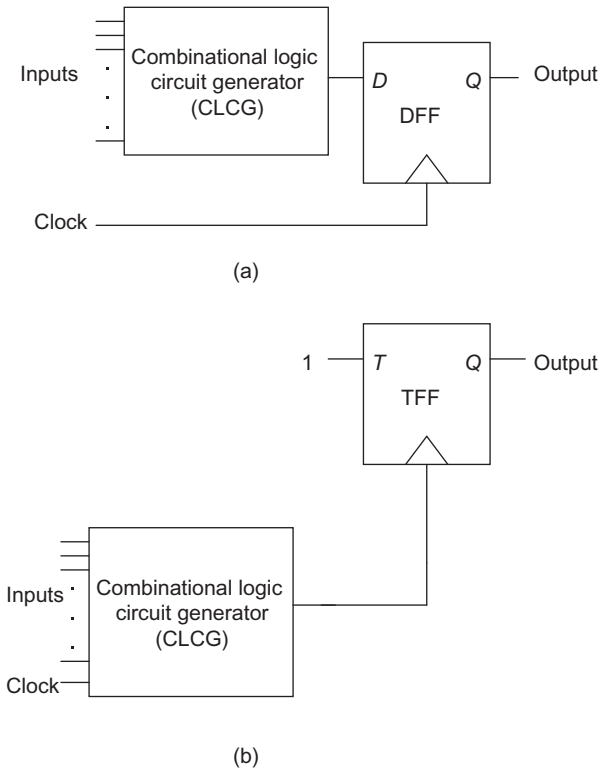

Figure 1. Logic elements. (a) Conventional logic element. (b) Our logic element.

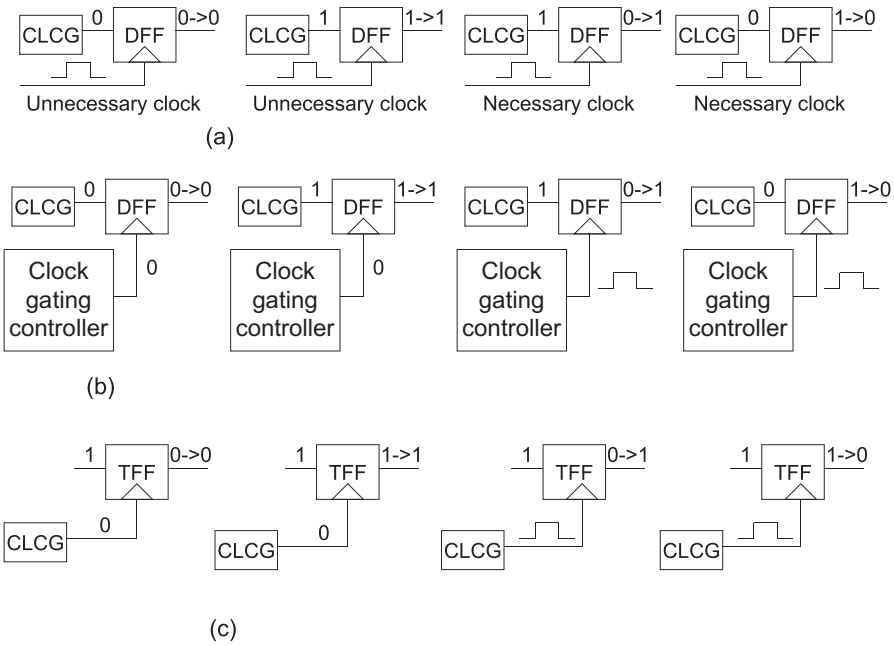

Figure 2. Basic operations of logic elements. (a) Conventional logic element without clock gating. (b) Conventional logic element with clock gating (no unnecessary clock). (c) Our logic element (no unnecessary clock).

Since the clock input of DFF is connected to the clock signal, the DFF is always clocked. When the D input of DFF has a different value compared to its output Q ($D \neq Q$), the DFF needs to be clocked in order to update its state as presented in Figure 2(a). Otherwise, when $D = Q$, the DFF does not need to be clocked. Such unnecessary transitions (i.e. unproductive activities) will waste power in the conventional LEs.

To stop unnecessary clock transitions in conventional LEs, clock gating was introduced in previous works of Cadenas and Megson (2003), Sutter et al. (2004), Zhang et al. (2006), Khan (2006), Parlak and Hamzaoglu (2007), Achronix (2008), Huda et al. (2009), Wang et al. (2009), Klein (2009), Actel (2009), Rivoallon (2010), Saleem and Khan (2010), Sterpone et al. (2011), Hussein et al. (2012), Ravindra and Anuradha (2012) and Semiconductor (2009). In clock gating, the clock input of DFF is not anymore connected directly to the clock signal, but it is controlled by the clock gating controller as shown in Figure 2(b). The clock gating controller blocks the clock signal for reaching DFFs clock inputs when the DFFs should not be clocked ($D = Q$). As a consequence, the unnecessary clock transitions can be avoided for power saving. The drawback of clock gating is the need of additional controllers that consume additional area and power. To reduce this overhead, the controller usually does not control an individual FF, but it controls a group of FFs together. Hence, the clock gating cannot block all of the unnecessary clock transitions.

To solve the above issues of conventional LEs, we propose a novel low-power LE depicted in Figure 1(b). The differences between our LE and the conventional LE are in the type of FFs and the LE organisation. Instead of using DFFs, we use T flip-flops (TFFs) with the T input kept at logic 1. The output of the CLCG is connected to the FF clock input. No clock signal is directly connected to the TFF, the clock signal is connected to the TFF through the CLCG when required. In FPGAs, CLCGs are implemented using

LUTs. In the case that one of the inputs of the LUT is used for feeding the clock signal, the LUT capacity is effectively decreased. In this case, our proposal needs either larger LUTs or more number of LUTs for implementing digital circuits. Since not all inputs of LUTs are used in real FPGA designs as reported in Mondal and Memik (2005b), it is possible that we can use these unused inputs for free to feed the clock signal. However, it is not a surprise that we need supported CAD tools to get the maximum benefits of unused inputs. This modification of CAD tools is left as one of our future directions. In this article, we only rely on the existing CAD tools, no modification of CAD tools, was performed.

The benefits of our LE are avoiding unnecessary clock transitions while omitting the additional clock gating controller as shown in Figure 2(c). The CLCG avoids clock transitions to be propagated to an individual FF when the state of the FF will not change. Thus, the unnecessary clock transitions are totally avoided at the level of individual FFs and hence dynamic power is reduced. Additional power and area are also saved in comparison to the clock gating approach, since the additional controller is not present.

Although not shown for simplicity in Figure 2, the *present state* and *inputs* are used to generate the *next state function* in the conventional LE; while in our circuit, the *present state*, *inputs* and *clock signal* are used to generate the *function to control TFFs clocks*. As a result, the way we design logic circuit will be different compared to the conventional approach. In conventional circuits, the data path, the control path and the clock are separated. In our circuits, all these paths are combined together into a *single unified path*.

Allowing faster clock rates than the conventional LEs is one additional advantage of our proposal. The FF can be clocked properly if its input is stable at least before its set-up time. In conventional LE, the input value of the DFF is not constant; it depends on the output of the connected CLCG. In our LE, since the T input of the TFF is constant ($T = 1$), the TFF is always ready to be clocked. As a consequence, logic circuits implemented using our LEs can be clocked faster than logic circuits using conventional LEs.

The shortest possible clock timing diagrams for circuits using our LEs compared to the conventional LEs are presented in Figure 3. Note that this experiment is used to investigate the differences in maximal clock rates. The CLCG (our) in Figure 3(b) has the clock as an additional input. This, however, does not impact the first-to-second stage shortest possible clock timing due to the $t_{\mathrm{pcq}}(\mathrm{TFF})$ delay that has to be satisfied. Here, Also note that the clock signal of the first-level TFF (A) is produced by the previous-level CLCG (our) not shown on the figure for simplicity. In the figure, $t_{\mathrm{pcq}}(\mathrm{DFF})$ is the clock-to-Q propagation delay of DFF; $t_{\mathrm{pd}}(\mathrm{CLCG(conv)})$ is the propagation delay of conventional CLCG; $t_{pd}(\mathrm{CLCG(our)})$ is the propagation delay of our CLCG; $t_{\text{set-up}}(\mathrm{DFF})$ is the set-up time of DFF; $t_{\mathrm{pcq}}(\mathrm{TFF})$ is the clock-to-Q propagation delay of TFF. From this figure, we can obtain the clock period of the circuit using conventional LEs as

$$T_c(\mathrm{conv}) \geq t_{pcq}(\mathrm{DFF}) + t_{\mathrm{pd}}(\mathrm{CLCG(conv)}) + t_{\text{set-up}}(\mathrm{DFF}) \tag{1}$$

and the clock period of the circuit using our LEs as

$$T_c(Our) \geq t_{pcq}(\mathrm{TFF}) + \mathrm{t_{pd}}(\mathrm{CLCG(Our)}) \tag{2}$$

From Equations (1) and (2), we can obtain the speedup as

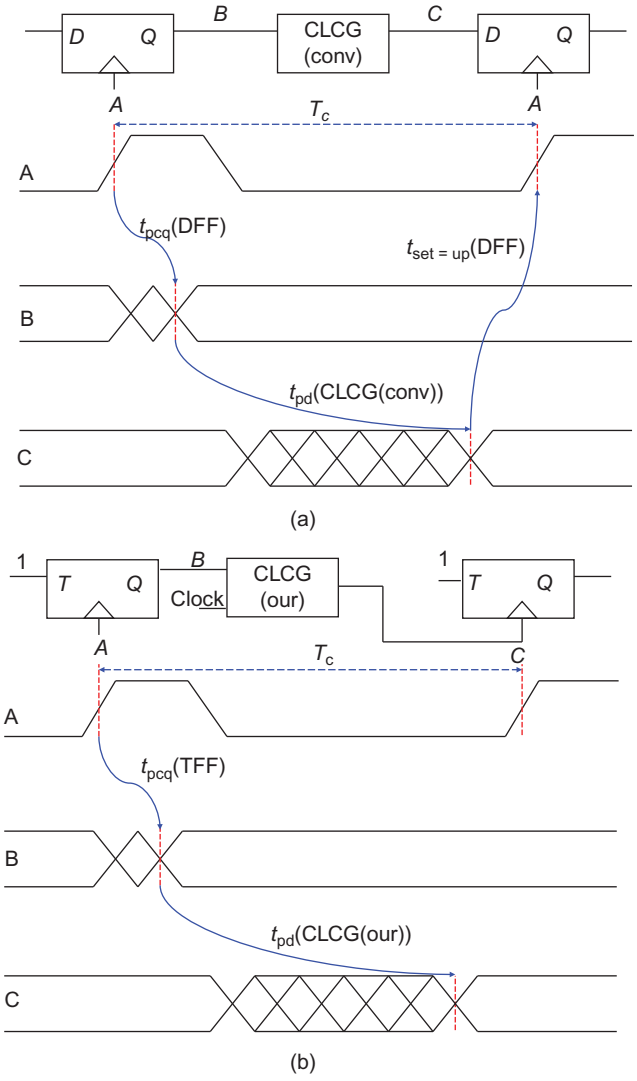

Figure 3. Shortest clock timing of conventional (a) and our (b) logic elements.

$$SPEEDUP = \frac{T_c(\text{conv})}{T_c(\text{Our})} = \frac{t_{\text{pcq}}(\text{DFF}) + t_{pd}(\text{CLCG}(\text{conv})) + t_{\text{set-up}}(\text{DFF})}{t_{pcq}(\text{TFF}) + t_{pd}(\text{CLCG}(\text{our}))} \tag{3}$$

If $t_{\text{pcq}}(\text{DFF}) = t_{\text{pcq}}(\text{TFF})$ and $t_{\text{pd}}(\text{CLCG}(\text{conv})) = t_{pd}(\text{CLCG}(\text{our}))$, the speedup becomes

$$\text{SPEEDUP} = 1 + \frac{t_{\text{set-up}}(\text{DFF})}{t_{\text{pcq}}(\text{TFF}) + t_{\text{pd}}(\text{CLCG})} \tag{4}$$

If the input of circuit changes during clock at logic 1, the possibility exists that this input will generate glitches that can alternate the next-stage TFF value. To address this

problem, we used pulsed clock signal. The width of the pulsed clock signal is set to be the minimum pulsed clock width of correctly operating TFF. In our experiments, the pulse width was 0.1 ns. Since the pulsed clock signal is narrow, the possibility that inputs change during clock at logic 1 is reduced. In case this very low possibility happens, the width of pulses caused by inputs during clock signal at logic 1 is always less than the width of the original pulsed clock signal and will not change the state of the TFFs. Hence, the circuit will keep working properly. It is possible to generate this narrow pulsed clock signal as demonstrated in Kozu et al. (1996), Pontikakis and Nekili (2002), McCorkle, Huynh, and Ochoa (2006), Yuan, Zheng, Ang, and Li (2007), Paulino, Goes, and Steiger-Garcao (2008), Ardehali (2010). This option is also currently supported by state-of-the-art industrial FPGAs (e.g. Xilinx Virtex 6) as reported in Xilinx (2012). Another way to handle this clocking issue is to register/synchronise the input with clock signal before it goes to the actual circuit. Since inputs are synchronised, the changing of input during clock at logic 1 will be ignored by the circuit. However, this requires additional logic area, latency and power overhead. For that reason, we choose to use a narrow-sized pulsed clock approach in our proposal.

For exemplifying our proposal, we show here an instance of how conventional circuits are converted into the circuits that implemented according to our proposal. Let us assume that we have a conventional circuit as illustrated in Figure 4(a) and we want to convert this circuit to our circuit as shown in Figure 4(b). In general, we use a simple formula when converting conventional circuits,

$$\text{clock}_i(\text{our}) = \begin{cases} \text{clock}_i(\text{conv}) & \text{if } Q_i(\text{conv}) \neq D_i(\text{conv}) \\ 0 & \text{if } Q_i(\text{conv}) = D_i(\text{conv}) \end{cases}$$

where

$\text{clock}_i(\text{our})$ is the clock input of FF i in our circuit;
$\text{clock}_i(\text{conv})$ is the clock input of FF i in conventional circuit;

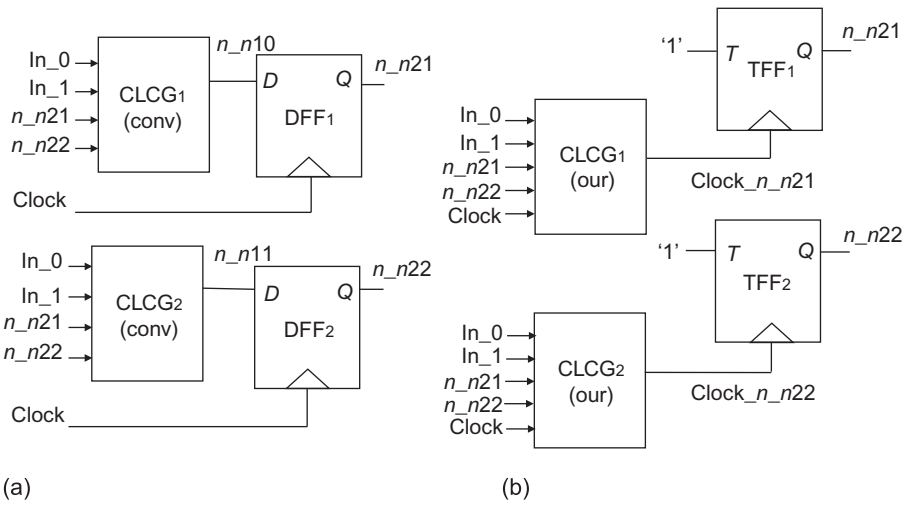

Figure 4. Simple circuit examples. (a) An example of conventional circuit. (b) An example of our circuit.

Q_i(conv) is the Q output of FF i in conventional circuit; and
D_i(conv) is the D input of FF i in conventional circuit.

Let us assume that CLCG_1(conv) has the truth table as shown in Table 1. To convert CLCG_1(*conv*) to CLCG_1(*our*), we can capture that $\mathrm{clock}_1(\mathrm{conv}) = \mathrm{clock}$, $D_1(\mathrm{conv}) = n\_n10$, $Q_1(\mathrm{conv}) = n\_n21$, and $\mathrm{clock}_1(\mathrm{our}) = \mathrm{clock}\_n\_n21$. By applying the above formula for computing clock_i(our), we can obtain the truth table of CLCG_1(our) for the logic function of clock_n_n21 as shown in Table 2.

Table 1. The truth table of CLCG_1(conv).

In_0	In_1	n_n21	n_n22	n_n10
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Table 2. The truth table of CLCG_1(Our).

In_0	In_1	n_n21	n_n22	clock_n_n21
0	0	0	0	0
0	0	0	1	Clock
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	Clock
0	1	1	1	Clock
1	0	0	0	Clock
1	0	0	1	Clock
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	Clock
1	1	1	1	Clock

4. Evaluation

4.1. *Transistor-level circuit evaluation*

4.1.1. *Experimental setup*

To evaluate the proposed LE, transistor-level circuit simulations were performed using LTSPICE tools (Technology, 2008) and 45 nm BSIM4 CMOS device models (Cao et al., 2010) with nominal VDD of 1.2 V. Because we use transistor-level simulation, all internal glitches are implicitly considered. The MCNC benchmark circuits (Yang 1991) were used for our study. Since our proposal is new, no CAD tools (high-level synthesis, technology mapping and place and route tools) are available for targeting FPDs using the proposed LE. For that reason, we performed all design transformations by hand. This is also why we did not evaluate our proposal with all MCNC benchmark circuits; we only evaluated the proposal with the circuits that were not too complex for manual LUT design as shown in Table 3. Since our proposal saves power for circuits with storage elements, we selected the representative MCNC benchmark circuits.

Due to the fact that SRAM cell values remain constant after configuration (no additional dynamic power) and there is no difference in the number of SRAM cells for FPDs using the conventional and our LEs (same static power), we do not model SRAM in our experiments. We connect the internal signals directly to VDD or ground depending on the intended SRAM content. The simulated nMOS and pMOS transistor dimensions were as follows: length ($L_n = 45\,\text{nm}$) / width ($W_n = 90\,\text{nm}$) and $L_p = 45\,\text{nm}$ / $W_p = 270\,\text{nm}$, respectively. The selected ratio between the nMOS and pMOS transistor widths $\left(\frac{W_p}{W_n} = 3\right)$ is to model the worst case scenario in respect to our proposal when leakage power is considered. To accurately model LUTs, multiplexers and routing circuits, we selected transmission-gate-based implementation as used by Xilinx commercial FPGAs patented in Pi and Crotty (2003). In this experiment, we assume that unused resources can be turned off to model power gating both for the conventional FPDs and for our proposal. Both the conventional and our FPDs are assumed to have the same architecture parameters (e.g. segment length, connection topologies, logic block size, clock network, cluster size and LUT size). The only difference between the conventional FPDs and our FPDs is the internal organisation of the LE.

First, we created experimental circuits representing both for the conventional and the proposed LEs. The experimental LEs are shown in Figure 5. In this experiment, an additional AND gate for feeding clock signal was used to make manual implementation of the MCNC circuits easier. An experimental conventional LE consists of a 4-input LUT, a DFF, and an output multiplexer as illustrated in Figure 5(a); while our proposal is

Table 3. The MCNC benchmark circuits.

Names	Inputs	Outputs	States	State transitions (STs)	STs to same state
bbtas	2	2	6	24	10
dk27	1	2	7	14	0
lion	2	1	4	11	5
mc	3	5	4	10	5
shiftreg	1	1	8	16	2
tav	4	4	4	49	0
train4	2	1	4	14	7

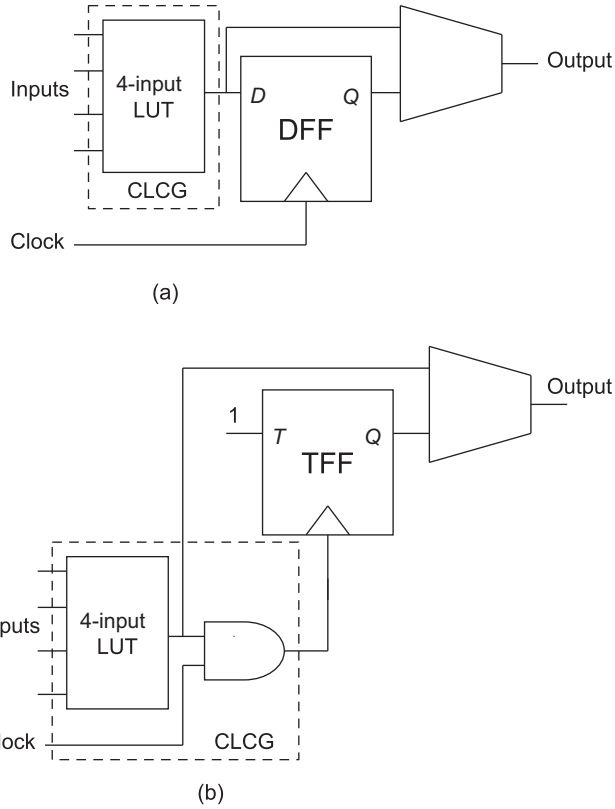

Figure 5. Logic elements used in our experiments. (a) Conventional logic element. (b) Our logic element.

represented by a 4-input LUT, a TFF, an output multiplexer, and an AND gate as shown in Figure 5(b). These two LE circuits were used for creating experimental FPD circuits. The FF circuits used for the simulation of the conventional LE and our LE are shown in Figures 6 and 7. For fair comparison, the only difference between the FF representing our proposal and the conventional FF is the feedback line from the inverted output to the *D* input. This feedback line forces the FF to behave as a TFF with *T* input permanently connected to a logic 1 value.

Next, the LE circuits were combined with interconnection components to create complete FPD circuits. The interconnection circuits (fixed wires and programmable switches) were used for connecting needed LE circuits which will be used for creating benchmark circuits.

Finally, we implemented each MCNC benchmark circuit onto the FPDs using conventional LEs and our LEs. MCNC circuits described using Berkeley logic interchange format (BLIF) (BLIF, 2005) mapped for 4-input-LUT-based FPDs were used for implementing circuits onto the FPD based on conventional LEs. We manually implemented each MCNC benchmark circuit onto our FPD circuit. In this step, we computed all of the functions needed for the LUTs in the new LEs which are totally different from the functions of the conventional approach. Next, we placed the computed functions in LUTs of the FPD using the proposed LEs and created the required interconnections for

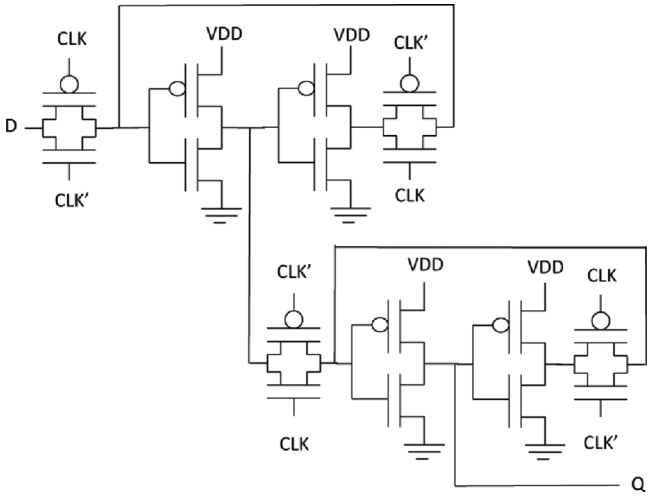

Figure 6. A flip-flop circuit used in conventional LE experiments.

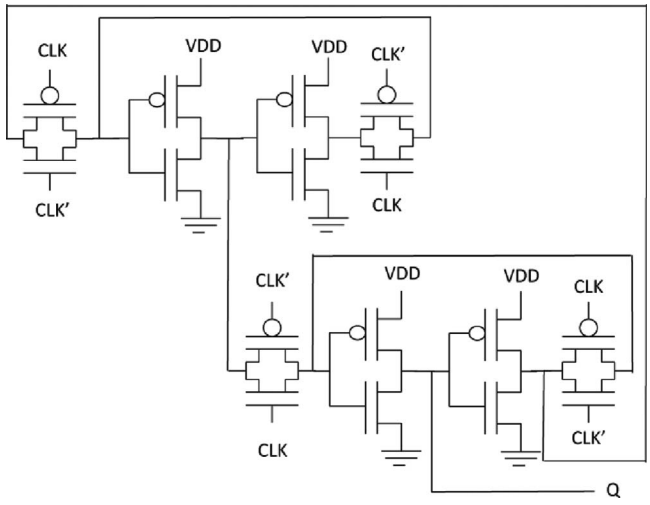

Figure 7. A flip-flop circuit used in the proposed LE experiments.

each MCNC circuit. The reconfigurations were done by modifying the contents of the 4-input LUTs, the output multiplexes, and the interconnect control signals.

We are aware that correct operation of circuits is much more important than power savings. For that reason, before we measured the needed performance parameters, all circuits have been verified to make sure that our circuits perform the same function as the conventional ones by using the same test vectors and the same simulation length. Figure 8 shows one example of the captured waveforms during this verification. For each benchmark circuit, we compared the simulation results of the two implementations (conventional and our proposal). After adopting the pulsed clock in our case, all circuits using our LEs worked properly, functioning correctly as the conventional one. The test vectors representing all possible input values combinations were used.

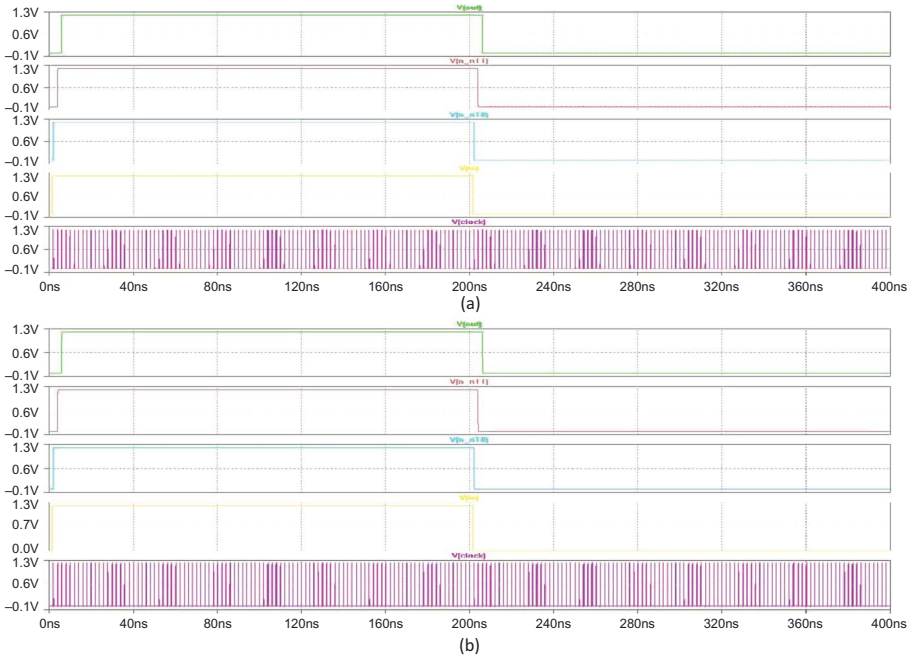

Figure 8. An example of correct operation experiments for shiftreg benchmark circuit using LTSPICE tools. (a) Waveform of shifterg using conventional LEs. (b) Waveform of shifterg using our LEs.

The benchmark circuits were simulated to obtain the needed performance parameters such as power, speed, and area for each benchmark circuit. Area is in terms of number of transistors required to implement the benchmark circuit using FPD circuits. The breakdowns of total power which consists of logic power (total power inside LEs), clock power and interconnect power were also obtained. To make our power study complete, we also analysed the static and dynamic power. The evaluation was conducted using 500 MHz clock speed representative for the CMOS technology node assumed in our experiments.

4.1.2. *Experimental results*

The experimental results in terms of power consumption for FPDs using both conventional and proposed LEs are depicted in Tables 4 and 5. The power reduction results

Table 4. Experimental results of logic, clock and interconnect power (μW).

	Logic power		Clock power		Interconnect power	
Benchmarks	Conventional	Our	Conventional	Our	Conventional	Our
bbtas	11357	9925	2320	811	2394	1002
dk27	39105	37642	2320	891	4188	2649
lion	5943	4460	1515	540	1507	627
mc	28204	25913	1515	559	2282	1374
shiftreg	3361	2975	2317	804	2171	777
tav	40505	39480	1522	641	3886	2970
train4	5576	4287	1514	538	1475	586

Table 5. Experimental results of dynamic, static and total power (μW).

	Dynamic power		Static power		Total power	
Benchmarks	Conventional	Our	Conventional	Our	Conventional	Our
bbtas	14461	9650	1610	2088	**16071**	**11738**
dk27	44166	39257	1447	1925	**45613**	**41182**
lion	8047	4390	918	1237	**8965**	**5627**
mc	30444	25970	1557	1876	**32001**	**27846**
shiftreg	6720	2948	1129	1608	**7849**	**4556**
tav	44034	40893	1879	2198	**45913**	**43091**
train4	7647	4174	918	1237	**8565**	**5411**

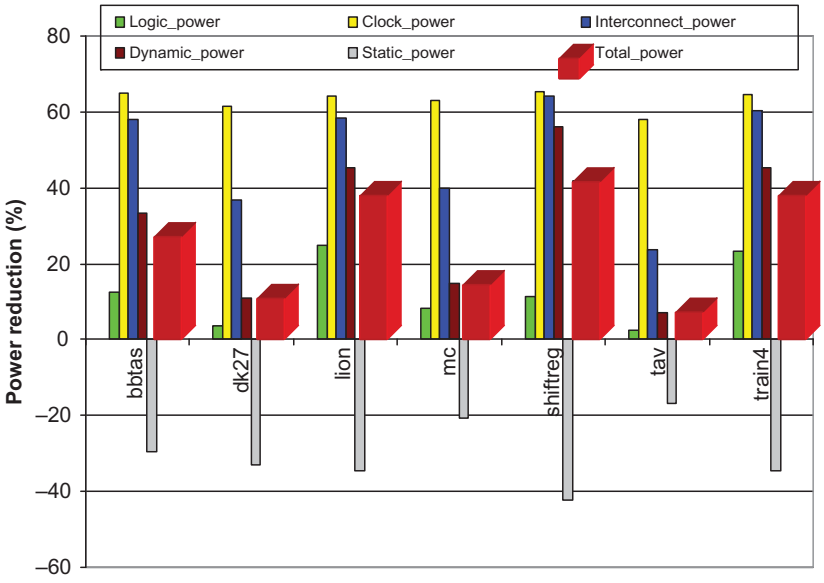

Figure 9. Power reduction (%).

presented in Figure 9 were computed based on the results from Tables 4 and 5. Besides power evaluation, we also investigated the area overhead and the performance improvements of the FPD using the proposed LEs as shown in Figure 10.

Since the FPD using the proposed LEs avoids unnecessary clocks as discussed earlier, it consumes 63% on average less clock power compared to the FPD using conventional LEs as shown in Figure 9. By avoiding unnecessary clocking clock inputs, the activity inside the proposed LE is also reduced. More specifically, in this case, the FPD using the proposed LEs reduces the energy denoted by Lang et al. (1997) as $E_0 \cdot N_0$ and $E_1 \cdot N_1$. As a result, the FPD using proposed LEs has 12% on average less logic power compared to the FPD using conventional LEs. This is related to the fact that even low-power FFs consume power during logic transition from zero-to-zero and from one-to-one as demonstrated in Lang et al. (1997), Nogawa and Ohtomo (1998), Markovic et al. (2001), Zhao et al. (2002, 2004), Sayed and Al-Asaad (2006), and Teh et al. (2006). In this work, we found that clocking a FF when its input at the condition of no making

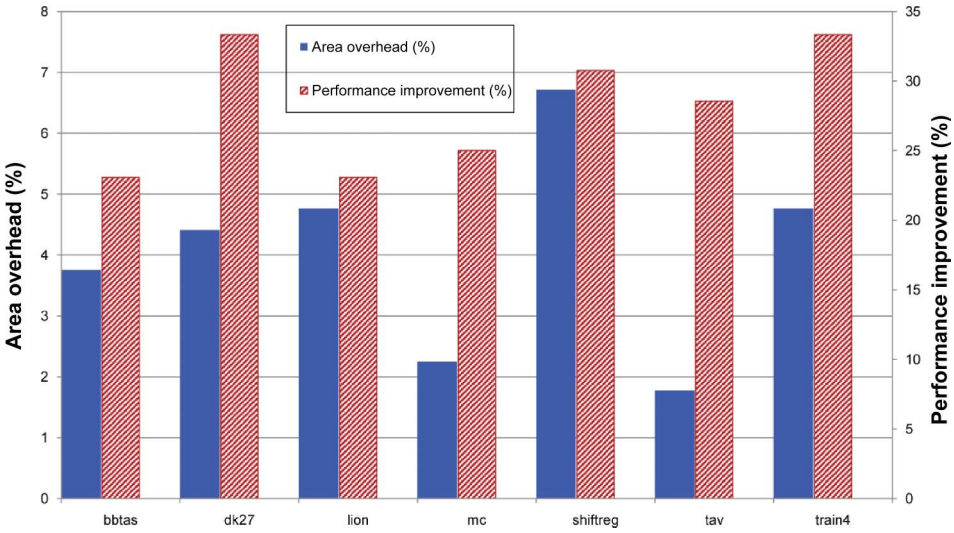

Figure 10. Area overhead and performance improvement (%).

state changing (e.g. DFF when its input D at logic 0 while its Q is also at logic 0) will not change the output Q of the FF (no logical transition); however, if we inspect further its output Q, we can see the voltage at output Q indeed changes (physical voltage changing), consuming power. This behaviour is like the one observed by other researchers (e.g. Lang et al., 1997; Markovic et al., 2001; Nogawa & Ohtomo, 1998; Sayed & Al-Asaad, 2006; Teh et al., 2006; Zhao et al., 2002, 2004). Since the FPD using our proposal is clocked only if needed, this kind of activities is avoided. More precisely, in this case, the FPD using the proposed LEs reduces the energy denoted by Lang et al. (1997) as $E_{gb} \cdot N_{gb}$ and $E_{ga} \cdot N_{ga}$. These glitches propagate through the interconnect resources of the FPDs, consuming power. By avoiding these glitches, the FPD using our LEs consumes 49% on average lower interconnect power compared to the FPD using conventional LEs.

The FPD using our proposal reduces 30% on average dynamic power compared to the FPD using conventional LEs by avoiding unnecessary activities such as clock, logic, and interconnect as presented in Figure 9. Since the proposed experimental LE has an additional AND gate, the FPD has 30% on average higher static power as shown in Figure 9 and 4% on average bigger area compared to the FPD using conventional LEs as presented in Figure 10. *In the future, not in this article*, since not all inputs of LUTs are used in real designs as reported in Mondal and Memik (2005b), we can use these unused inputs to feed the clock signal. In this case, we can avoid the additional logic level (the AND gate) for feeding the clock signal.

Although the FPD using our LEs consumes more static power than the FPD using conventional elements, the overall power consumption of the FPD using our proposal is lower than the conventional one as shown in Figure 9. Since the impact of increase in static power is lower than the impact of reducing the clock, logic and interconnect powers, the FPD using our proposed LEs still can reduce 25% on average total power compared to the FPD using conventional LEs as shown in Figure 9.

Circuits that do not change their internal state very often will avoid many clock transitions and will be able to achieve more dynamic power reduction compared to

circuits that frequently change their states. As shown in Table 3, the state of the storage elements in the *dk27* and *tav* benchmark circuits never remains the same. That is why the total power reduction achieved for these benchmark circuits is smaller compared to other benchmark circuits.

In the conventional LE, the DFF can be clocked by clock signal if only if the *D* input is ready before the needed set-up time for the FF to work properly. In contrast, the TFF in our LE is always ready to receive clock signal because the T input of its TFF is always ready at logic 1. Thus, the FPD using proposed LEs runs 28% on average faster than the FPD using conventional LEs as shown in Figure 10.

4.2. *Evaluation using a real CAD tool on a real FPGA*

4.2.1. *Experimental set-up*

To evaluate our proposal further, in this experiment, we force a CAD tool to implement circuits in the real FPGA behaving like our proposed LE using a new HDL coding style. This coding style is presented in our previous work: Marconi et al. (2010a, 2010b), Marconi (2011).

The experimental set-up is shown in Figure 11. Each MCNC benchmark circuit (Yang, 1991) is converted into two Very High Speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL) files (conventional and our VHDL files) to represent the two VHDL coding styles (conventional and our coding styles). Each VHDL file is compiled for Stratix EP1S10F484C5 (Altera Corporation, San Jose, CA) using Compiler Tool from Quartus II. The area needed for implementing each circuit in terms of number of LEs is reported by the Altera Compiler Tool. The Waveform Editor from

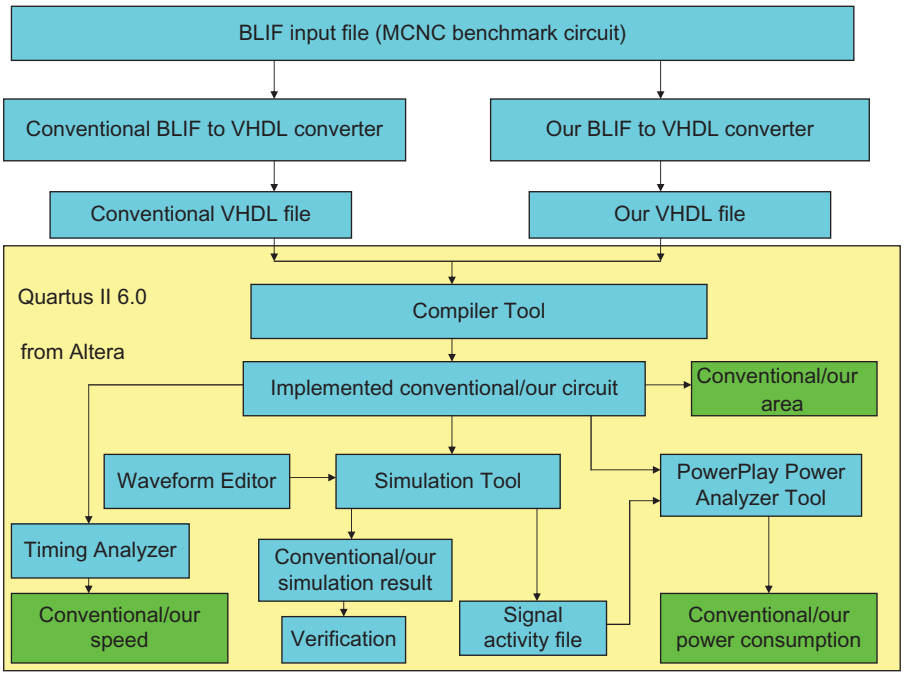

Figure 11. Experimental setup.

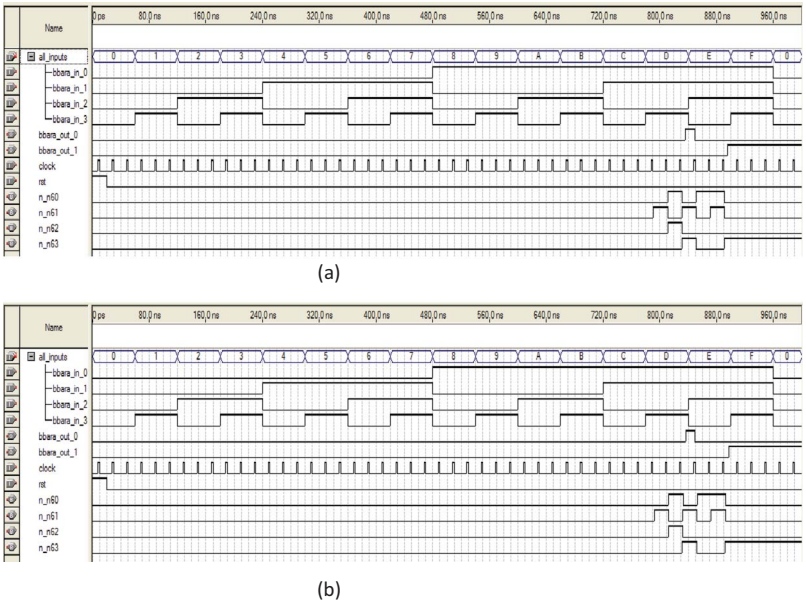

Figure 12. An example of correct operation experiments for bbara benchmark circuit using the simulation tool of Quartus II from Altera. (a) Waveform of bbara using conventional LEs. (b) Waveform of bbara using our LEs.

Quartus II is used to generate test vectors for each benchmark circuit. Those vectors are applied to the implemented circuit using Simulation Tool from Quartus II. Since the functional correctness is much more important than power savings, each circuit is verified by comparing the simulation results between the conventional and our circuits. This step is needed to ensure that these circuits generate functionally correct identical circuits. One instance of this verification is shown in Figure 12. Besides generating simulation results, the Simulation Tool also generates the signal activity file (SAF). To evaluate power consumption, the SAF file and the implemented circuit from the previous step are fed into the Quartus II PowerPlay Power Analyzer Tool to obtain total, dynamic and static power results. To compare performance of the implemented circuits, the Timing Analyzer from Quartus II is used. Our study focused on the maximum clock frequency.

4.2.2. *Experimental results*

The experimental results of power consumption using a 50 MHz clock are presented in Table 6. Table 6 shows that our proposal can lead to reduction in dynamic power and total power, but will not reduce static power. Since our proposal can avoid unnecessary transitions by clocking flip-flops only when needed, it can lead to reduction in dynamic power consumption (75% on average) compared to conventional circuit. The degree of power reduction depends on the nature of the circuit, circuits with many unnecessary transitions can take more advantages of our proposal in terms of power consumption. This 75% dynamic power reduction results in only 15% on average total power consumption reduction at 50 MHz since the static power is dominating. The static power reported by the tools for all of the investigated circuits was 187.5 mW.

Table 6. Experimental results of power consumption at 50 MHz.

	Dynamic power (mW)			Total power (mW)		
Circuits	Conventional	Our	Reduction (%)	Conventional	Our	Reduction (%)
lion	39.87	4.75	88.09	227.37	192.25	15.45
bbara	36.2	0.78	97.85	223.7	188.28	15.83
bbsse	39.58	6.43	83.75	227.08	193.93	14.6
s298	45.81	10.54	76.99	233.31	198.04	15.12
dk16	51.82	16.93	67.33	239.32	204.43	14.58
dk14	55.58	16.32	70.64	243.08	203.82	16.15
tbk	40.47	4.3	89.37	227.97	191.8	15.87
beecount	44.92	9.56	78.72	232.42	197.06	15.21
cse	41.96	6.48	84.56	229.46	193.98	15.46
s1494	71.73	27.17	62.12	259.23	214.67	17.19
ex1	48.71	16.81	65.49	236.21	204.31	13.5
keyb	41.09	5.21	87.32	228.59	192.71	15.7
planet	42.33	5.88	86.11	229.83	193.38	15.86
pma	89.47	53.51	40.19	276.97	241.01	12.98
s1	52.95	21.23	59.91	240.45	208.73	13.19
styr	66.53	31.55	52.58	254.03	219.05	13.77
s1488	63.96	30.37	52.52	251.46	217.87	13.36
sand	36.14	0.49	98.64	223.64	187.99	15.94

Table 7. Experimental results of area and maximum clock frequency.

	Area (#LEs)			Maximum clock frequency(MHz)		
Circuits	Conventional	Our	Overhead (%)	Conventional	Our	Improvement (%)
lion	7	7	0	437.06	467.07	6.87
bbara	25	29	16	305.44	340.02	11.32
bbsse	45	49	8.89	264.27	274.73	3.96
s298	740	903	22.03	93.82	95.27	1.55
dk16	85	86	1.18	219.97	226.3	2.88
dk14	28	37	32.14	276.78	331.79	19.87
tbk	69	78	13.04	139.14	153.82	10.55
beecount	11	16	45.45	367.92	390.63	6.17
cse	73	80	9.59	216.08	232.34	7.52
s1494	249	261	4.82	190.99	208.9	9.38
ex1	110	118	7.27	242.19	256.41	5.87
keyb	90	96	6.67	190.19	214.5	12.78
planet	215	231	7.44	188.82	210.7	11.59
pma	76	83	9.21	210.39	224.77	6.83
s1	140	146	4.29	117.04	120.44	2.9
styr	202	210	3.96	298.78	310.95	4.07
s1488	243	255	4.94	194.89	197.71	1.45
sand	205	213	3.9	180.08	199.48	10.77

The experimental results of area and performance are presented in Table 7. Table 7 shows that our proposal can also increase the performance of the circuits by 7.6% on average. Since the set-up time is becoming far less significant compared to total longest path for circuits with more logic level, the performance improvement is minimal.

Table 8. Power reduction at 100, 150 and 200 MHz.

	Dynamic power reduction (%)			Total power reduction (%)		
Circuits	100	150	200	100	150	200
lion	88.09	88.09	88.09	26.28	34.31	40.49
bbara	97.85	97.85	97.85	27.26	35.89	42.64
bbsse	83.75	83.75	83.75	24.86	32.47	38.34
dk16	67.33	67.33	67.33	23.97	30.52	35.35
dk14	70.64	70.64	70.64	26.29	33.25	38.32
tbk	89.37	–	–	26.95	–	–
beecount	78.72	78.72	78.72	25.5	32.92	38.52
cse	84.56	84.56	84.56	26.14	33.97	39.94
s1494	62.12	62.12	–	26.93	33.2	–
ex1	65.49	65.49	65.49	22.39	28.68	33.37
keyb	87.32	87.32	–	26.61	34.64	–
planet	86.11	86.11	–	26.79	34.77	–
pma	40.19	40.19	40.19	19.63	23.66	26.37
s1	59.91	–	–	21.62	–	–
styr	52.58	52.58	52.58	21.82	27.11	30.85
s1488	52.52	52.52	–	21.3	26.56	–
sand	98.64	98.64	–	27.45	36.14	–

The clock signal needs to be fed to LUTs before it reaches the flip-flops, our proposal consumes 11% on average more area compared to the conventional one as shown in this table. If the clock signal can be fed to LUTs using unused inputs, our style does not need additional LUTs for this purpose (e.g. lion). However, if this is not the case, our proposal will consume more LEs as shown in Table 7. In our experiment, we had considered this area overhead when we evaluated power consumption and performance.

To investigate all implemented circuits further, we run them using different clock frequencies: 100, 150 and 200 MHz. The results of this experiment are presented in Table 8. Note that some of the benchmarks did not synthesised at this frequency for both design styles (shown with a dash sign in the table). Since static power, area and performance are not affected by changing the clock frequency, these tables only show dynamic power and total power consumption results. From these tables, we can observe that dynamic power consumption is linearly proportional to clock frequency. These tables also show that our proposal can reduce total power consumption by 25%, 32% and 36% on average compared to conventional one at 100, 150 and 200 MHz, respectively. Since dynamic power is higher when the clock frequency is increased, the reduction of total power is also increased for higher clock frequencies.

Note that simple experimental circuits were used in earlier evaluation. To further evaluate the proposal, we experimented with bigger representative circuits by duplicating multiple simple circuits into the FPGA and investigated the effect on overall power reduction as depicted in Figure 13. More working circuits means additional dynamic power; the dynamic power becomes more dominant compared to static power. As a result, our proposal reduces more total power when the number of circuits simultaneously implemented on the FPGA increases. This figure indicates that our proposal can reduce total power by 47% on average at 50 MHz. Total power is significantly reduced at higher frequency, at least 53% at 300 MHz for bigger circuits. Total power reduction saturates as

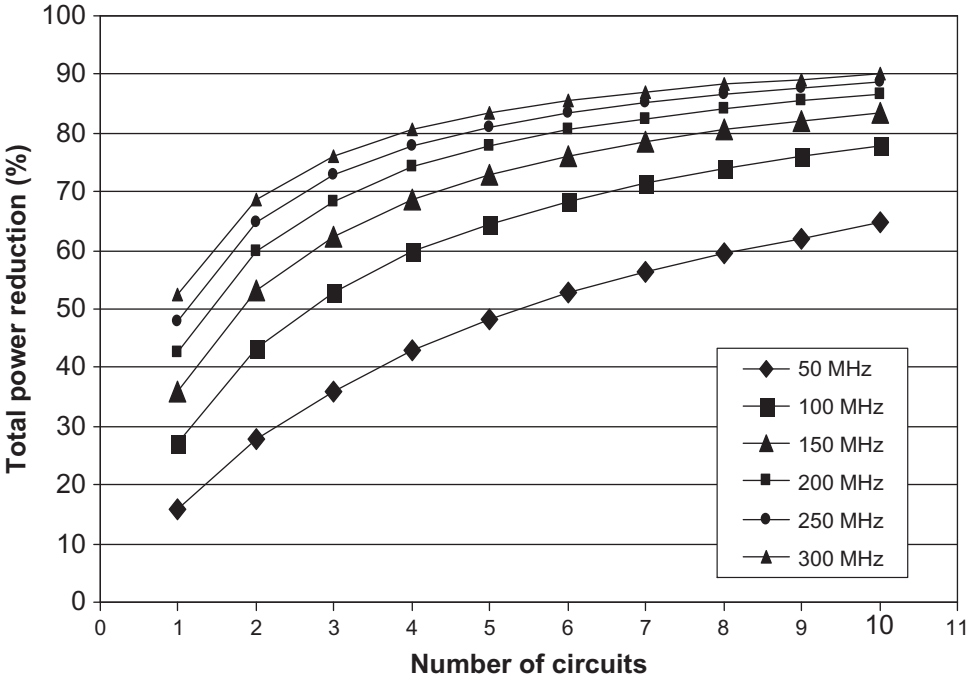

Figure 13. Overall power reduction (%) versus number of circuits.

Table 9. Comparison to clock gating solutions.

Proposals	Power reduction	Performance
Cadenas and Megson (2003)	No power benefit	Not available
Sutter et al. (2004)	0–39% dynamic power	0–4.7 ns delay penalty
Zhang et al. (2006)	5–33% total power	Not available
Parlak and Hamzaoglu (2007)	up to 13% total power	Not available
Huda et al. (2009)	6.2–7.7% total power	0–2% slower
Wang et al. (2009)	1.8–27.9% total power	1.1% faster
Rivoallon et al. (2010)	up to 30% dynamic power	Not available
Saleem and Khan (2010)	5–7.33% total power	Not available
Sterpone et al. (2011)	40–62% total power	Not available
Hussein et al. (2012)	10–80% dynamic power	Not available
Ravindra and Anuradha (2012)	20% total power	Not available
Our	48% total power on average	17% faster on average

shown in Figure 13. This effect is caused by the constant static power contribution that will start dominating the total power number when the number of implemented circuits increases.

Since the closest related work is clock gating technique, we conclude this section with Table 9 that shows the comparison between our solution and clock gating solutions. Clock gating results are obtained from the original papers: Cadenas and Megson (2003), Sutter et al. (2004), Zhang et al. (2006), Parlak and Hamzaoglu (2007), Huda et al. (2009), Wang et al. (2009), Rivoallon (2010), Saleem and Khan (2010), Sterpone et al. (2011), Hussein

et al. (2012), and Ravindra and Anuradha (2012). Unlike clock gating, our proposal does not need an additional controller to stop clock propagation. As a consequence, the FPD using our proposed LEs not only consumes 48% less total power by avoiding unnecessary activities: clock, logic, and interconnect, but also it runs 17% faster than traditional FPDs on average of all experimental results (i.e. SPICE simulations and real FPGA implementations). Refering to the gap between FPGAs and ASICs presented in Kuon and Rose (2006), our proposal can reduce the power consumption gap from 12 times to 6 times. We could not directly compare the area overhead since this information is not reported in the clock gating papers considered. In our case the area overhead is 22.5% on average of the overall experimental results.

5. Conclusions

In this article, we have proposed a novel low-power LE to replace the conventional structures in PLDs and FPGAs. Since unnecessary clock transitions are avoided, the clock power is reduced. By avoiding unnecessary clock transitions, the activity inside the proposed LEs is also reduced. As a result, the FPD using the proposed LEs consumes less logic power compared to the FPD using conventional LEs. Because of activity reduction, the LEs interconnect power is also reduced compared to the FPD using conventional LEs. Moreover, since we do not need an additional controller to hold clock activity, power and area are reduced in comparison to clock gating.

In our LE, since the T input of the FF is always in logic 1, the FF is always ready to be clocked. As a consequence, the FPD using our proposed LEs not only consumes less total power by avoiding unnecessary activities: clock, logic, and interconnect, but also runs faster compared to conventional LEs because of its "always ready" flip-flops.

The proposed LE has been validated and evaluated via SPICE simulations for a 45-nm CMOS technology as well as via a real CAD tool (Altera Quartus II Compiler Tools) on a real FPGA (Altera Stratix EP1S10F484C5) using the standard MCNC benchmark circuits. The overall experimental results show that FPDs using our proposal not only have 48% lower total power but also run 17% faster than conventional FPDs on average.

Designing circuits targeting FPDs based on our proposed low-power LEs was performed by hand in this work. To make this design process automatically, CAD tools development for FPDs targeting our proposed LEs is needed to be investigated further. Benefits of replacing FFs with latches are increased performace, area reduction and minimized power consumption as have been investigated in ASIC designs. Another interesting research direction is to study of replacing FFs with latches in FPDs targeting our proposed LEs. Applying the proposed idea in this current work to ASICs could also be investigated further.

Acknowledgements

This work is sponsored by the hArtes project (IST-035143) supported by the Sixth Framework Programme of the European Community under the thematic area Embedded Systems. Besides, the first author has also been supported by MOE Singapore research grant No. MOE2009-T2-1-033. The authors would like to thank Prof. Jarmo Takala, Dr. Dimitris Theodoropoulos and the reviewers for their valuable comments, suggestions and helps.

References

Abusaidi, P., Klein, M., & Philofsky, B. (2008). Virtex-5 FPGA system power design considerations. Xilinx WP285 (v1.0).

Achronix. (2008). *Introduction to Achronix FPGAs*. WP001 Rev.1.6.

Actel. (2008). Total system power: Evaluating the power profile of FPGAs. Mountain View, CA: Actel Corporation.

Actel. (2009). Dynamic power reduction in flash FPGAs. Application Note AC323. Mountain View, CA: Actel Corporation.

Alexander, M. J. (1997). Power optimization for FPGA look-up tables. *Proceedings of the 1997 international symposium on Physical design*, Napa Valley, CA, ISPD '97, pp. 156–162. New York, NY: ACM.

Altera. (2006a). *FPGA Architecture*. WP-01003. San Jose, CA: Altera Corporation.

Altera. (2006b). Power management in portable systems using MAX II CPLDs. *Application Note*, *422*. San Jose, CA: Altera Corporation.

Anderson, J., & Najm, F. (2002). Power-aware technology mapping for LUTbased FPGAs. *Proceedings of IEEE international conference on field-programmable technology*, 16–18 December (pp. 211–218). Washington, DC: IEEE.

Anderson, J., & Najm, F. (2004). A novel low-power FPGA routing switch. *Proceedings of the IEEE custom integrated circuits conference*, 3–6 October, pp. 719–722. Washington, DC: IEEE.

Anderson, J. H., Najm, F. N., & Tuan, T. (2004). Active leakage power optimization for FPGAs. *Proceedings of the 2004 ACM/SIGDA 12th international symposium on Field programmable gate arrays*, Monterey, CA, FPGA '04, pp. 33–41. New York, NY: ACM.

Anderson, J. H., & Ravishankar, C. (2010). FPGA power reduction by guarded evaluation. *Proceedings of the 18th annual ACM/SIGDA international symposium on Field programmable gate arrays*, Monterey, CA, FPGA '10, pp. 157–166. New York, NY: ACM.

Ardehali, M. (2010). *Narrow pulse generator*. US Patent No. 7782111 B2.

Atmel. (2000). Saving power with Atmel PLDs. Atmel Application Note. San Jose, CA: Atmel Corporation.

Azizi, N., & Najm, F. (2005). Look-up table leakage reduction for FPGAs. *Proceedings of the IEEE custom integrated circuits conference*, 21 September, pp. 187–190. Washington, DC: IEEE.

Bard, S., & Rafla, N. I. (2008). Reducing power consumption in FPGAs by pipelining. *Proceedings of circuits and systems*, pp. 173–176. Washington, DC: IEEE.

Becker, J., Hübner, M., & Ullmann, M. (2006). Run-time FPGA reconfiguration for power-/cost-optimized real-time systems. *Proceedings of the international conference on very large scale integration of system-on-chip*, 16–18 October. Washington, DC: IEEE.

Berkeley Logic Interchange Format (BLIF). (2005). Berkeley, CA: University of California Berkeley.

Birkner, J. M., & Chua, H. T. (1978). Programmable Array Logic Circuit. US Patent No. 4124899.

Boemo, E. I., Rivera, G. G. d., López-Buedo, S., & Meneses, J. M. (1995). Some notes on power management on FPGA-based systems. in *Proceedings of the 5th international workshop on field-programmable logic and applications*, FPL '95, London, pp. 149–157. Berlin: Springer-Verlag.

Brown, S., Brown, S., & Rose, J. (1996). Architecture of FPGAs and CPLDs: A tutorial. *IEEE Design and Test of Computers, 13*, 42–57.

Cadenas, O., & Megson, G. (2003). Power performance with gated clocks of a pipelined Cordic core. *Proceedings of the 5th international conference on ASIC, 2003*., October. (Vol. 2, pp. 1226–1230).

Cao, Y., Balijepalli, A., Sinha, S., Wang, C. C., Wang, W., & Zhao, W. (2010). The predictive technology model in the late silicon era and beyond. *Foundations and Trends in Electronic Design Automation, 3*, 305–401.

Chen, C., Parsa, R., Patil, N., Chong, S., Akarvardar, K., Provine, J., …, , & Mitra, S. (2010). Efficient FPGAs using nanoelectromechanical relays. *Proceedings of the 18th annual ACM/SIGDA international symposium on Field programmable gate arrays*, Monterey, CA, FPGA '10 (pp. 273–282). New York, NY: ACM.

Chen, C. S., Hwang, T., & Liu, C. L. (1997). Low power FPGA design a reengineering approach. *Proceedings of the 34th annual design automation conference*, Anaheim, CA, DAC '97 (pp. 656–661). New York, NY: ACM.

Chen, D., & Cong, J. (2004). Delay optimal low-power circuit clustering for FPGAs with dual supply voltages. in *Proceedings of the 2004 international symposium on Low power electronics and design*, Newport Beach, CA, ISLPED '04 (pp. 70–73). New York, NY: ACM.

Chen, D., Cong, J., & Fan, Y. (2003). Low-power high-level synthesis for FPGA architectures. *Proceedings of the 2003 international symposium on Low power electronics and design*, Seoul, Korea, ISLPED '03 (pp. 134–139). New York, NY: ACM.

Chen, D., Cong, J., Li, F., & He, L. (2004). Low-power technology mapping for FPGA architectures with dual supply voltages. *Proceedings of the 2004 ACM/SIGDA 12th international symposium on Field programmable gate arrays*, Monterey, CA, FPGA '04, New York, NY, USA: ACM.

Cheng, L., Chen, D., & Wong, M. D. F. (2007). GlitchMap: an FPGA technology mapper for low power considering glitches. *Proceedings of the 44th annual design automation conference*, San Diego, CA, DAC '07 (pp. 318–323). New York, NY: ACM.

Chow, C. T., Tsui, L. S. M., Leong, P. H. W., Luk, W., & Wilton, S. J. E. (2005). Dynamic voltage scaling for commercial FPGAs. *Proceedings of IEEE International conference on field-programmable technology*, pp. 173–180. Washington, DC: IEEE.

Constantinides, G. A. (2006). Word-length optimization for differentiable nonlinear systems. *ACM Transactions on Design Automation of Electronic Systems, 11*, 26–43.

Czajkowski, T. S., & Brown, S. D. (2007). Using negative edge triggered ffs to reduce glitching power in FPGA circuits. *Proceedings of the 44th annual design automation conference*, San Diego, CA, DAC '07 (pp. 324–329). New York, NY: ACM.

Dinh, Q., Chen, D., & Wong, M. D. (2009). A routing approach to reduce glitches in low power FPGAs. *Proceedings of the 2009 international symposium on physical design*, San Diego, CA, ISPD '09 (pp. 99–106). New York, NY: ACM.

Douglass, S. (2006). *Introducing the Virtex-5 FPGA Family*. Xilinx.

Farrahi, A. H., & Sarrafzadeh, M. (1994). FPGA technology mapping for power minimization. *Proceedings of international workshop on field-programmable logic and applications: Field-programmable logic, architectures, synthesis and applications*, *849*, 66–77. Berlin: Springer.

Fischer, R., Buchenrieder, K., & Nageldinger, U. (2005). Reducing the power consumption of FPGAs through retiming. *Proceedings of the 12th IEEE international conference and workshops on engineering of computer-based systems* (pp. 89–94). Washington, DC: IEEE Computer Society.

Gaffar, A. A., Clarke, J. A., & Constantinides, G. A. (2006). PowerBit – power aware arithmetic bit-width optimization. *Proceedings of IEEE international conference on field programmable technology*, 13–15 December (pp. 289–292). Washington, DC: IEEE.

Gayasen, A., Lee, K., Vijaykrishnan, N., Kandemir, M. T., Irwin, M. J., & Tuan, T. (2004). A dual-VDD low power FPGA architecture. *Proceedings of the international conference on field-programmable logic and its applications (FPL)*, *3203*, 145–157. Berlin: Springer.

Gayasen, A., Srinivasan, S., Vijaykrishnan, N., & Kandemir, M. T. (2007). Design of power-aware FPGA fabrics. *International Journal of Embedded Systems, 3*(1/2). 52–64.

Gayasen, A., Tsai, Y., Vijaykrishnan, N., Kandemir, M., Irwin, M. J., & Tuan, T. (2004). Reducing leakage energy in FPGAs using region-constrained placement. *Proceedings of the 2004 ACM/SIGDA 12th international symposium on field programmable gate arrays*, Monterey, CA, FPGA '04 (pp. 51–58). New York, NY: ACM.

George, V., Zhang, H., & Rabaey, J. (1999). The design of a low energy FPGA. *Proceedings of the 1999 international symposium on Low power electronics and design*, San Diego, CA, ISLPED '99 (pp. 188–193). New York, NY: ACM.

Gupta, S., Anderson, J., Farragher, L., & Wang, Q. (2007). CAD techniques for power optimization in virtex-5 FPGAs. *Proceedings of IEEE custom integrated circuits conference*, September (pp. 85–88). Washington, DC: IEEE.

Hassan, H., Anis, M., Daher, A. E., & Elmasry, M. (2005). Activity packing in FPGAs for leakage power reduction. *Proceedings of the conference on design, automation and test in Europe – Volume 1*, DATE '05 (pp. 212–217). Washington, DC: IEEE Computer Society.

Hassan, H., Anis, M., & Elmasry, M. I. (2008). Input vector reordering for leakage power reduction in FPGAs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 27*(9). 1555–1564.

Howland, D., & Tessier, R. (2008). RTL dynamic power optimization for FPGAs. *Proceedings of IEEE midwest symposium on circuits and systems*, August (pp. 714–717). Washington, DC: IEEE.

Hu, Y., Lin, Y., He, L., & Tuan, T. (2008). Physical synthesis for FPGA interconnect power reduction by dual-Vdd budgeting and retiming. *ACM Transactions on Design Automation of Electronic Systems, 13*, 30: 1–30:29.

Huda, S., Mallick, M., & Anderson, J. H. (2009). Clock gating architectures for FPGA power reduction. *Proceedings of IEEE international conference on field-programmable logic and applications (FPL)* (pp. 112–118). Berlin: Springer.

Hussein, J., Klein, M., & Hart, M. (2012) *Lowering power at 28 nm with Xilinx 7 series FPGAs*., Xilinx WP389.

Ishihara, S., Hariyama, M., & Kameyama, M. (2010). A low-power FPGA based on autonomous fine-grain power gating. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems, PP*(99), 1–13.

Jenkins, C., & Ekas, P. (2006). Low-power Software-defined Radio Design using FPGAs, Altera.

Kao, C. (2005). Benefits of Partial Reconfiguration, Xilinx.

Khan, M. (2006). Power Optimization in FPGA Designs, Altera.

Khan, M. (2007). Power Optimization Innovations in 65-nm FPGAs, Altera.

Klein, M. (2009). Power Consumption at 40 and 45 nm, Xilinx WP298.

Koga, M., Iida, M., Amagasaki, M., Ichida, Y., Saji, M., Iida, J., & Sueyoshi, T. (2010). First prototype of a genuine power-gatable reconfigurable logic chip with FeRAM cells. *FPL' 10* (pp. 298–303).

Kozu, S., Daito, M., Sugiyama, Y., Suzuki, H., Morita, H., Nomura, M., … Yano, Y. (1996). A 100 MHz, 0.4 W RISC processor with 200 MHz multiply adder, using pulse-register technique. *Solid-state circuits conference, 1996. Digest of technical papers. 42nd ISSCC., 1996 IEEE International*, February (pp. 140–141, 432).

Kumthekar, B., Benini, L., Macii, E., & Somenzi, F. (2000). Power optimization of FPGA-based designs without rewiring. *IEE Proceedings Computers and Digital Techniques, 147*(3), 167–174.

Kuon, I., & Rose, J. (2006). Measuring the gap between FPGAs and ASICs. *Proceedings of the 2006 ACM/SIGDA 14th international symposium on Field programmable gate arrays*, Monterey, CA, FPGA '06 (pp. 21–30). New York, NY: ACM.

Lamoureux, J., Lemieux, G. G. F., & Wilton, S. J. E. (2008). GlitchLess: Dynamic power minimization in FPGAs through edge alignment and glitch filtering. *IEEE Transactions on Very Large Scale Integration Systems, 16*, 1521–1534.

Lamoureux, J., & Wilton, S. J. E. (2003). On the interaction between power-aware FPGA CAD algorithms, *Proceedings of the 2003 IEEE/ACM international conference on computer-aided design*, ICCAD '03 (pp. 701–708). Washington, DC: IEEE Computer Society.

Lamoureux, J., & Wilton, S. J. E. (2007). Clock-aware placement for FPGAs. *Proceedings of international conference on field programmable logic and applications* (pp. 124–131). Berlin: Springer.

Lang, T., Musoll, E., & Cortadella, J. (1997). Individual flip-flops with gated clocks for low power datapaths. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, 44*(6), 507–516.

Lattice Semiconductor Corporation (2009). Practical low power CPLD design. Hillsboro, OR: Lattice Semiconductor.

Li, C. F., Yuh, P. H., Yang, C. L., & Chang, Y. W. (2007). Post-placement leakage optimization for partially dynamically reconfigurable FPGAs. *Proceedings of the 2007 international symposium on Low power electronics and design*, Portland, OR, ISLPED '07 (pp. 92–97). New York, NY: ACM.

Li, F., Chen, D., He, L., & Cong, J. (2003). Architecture evaluation for powerefficient FPGAs. *Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays*, Monterey, CA, FPGA '03 (pp. 175–184). New York, NY: ACM.

Li, F., Lin, Y., & He, L. (2004a). FPGA power reduction using configurable dual- Vdd. *Proceedings of the 41st annual design automation conference*, San Diego, CA, DAC '04 (pp. 735–740). New York, NY: ACM.

Li, F., Lin, Y., & He, L. (2004b). Vdd programmability to reduce FPGA interconnect power. *Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design*, ICCAD '04 (pp. 760–765). Washington, DC: IEEE Computer Society.

Li, F., Lin, Y., He, L., & Cong, J. (2004c). Low-power FPGA using pre-defined dual-Vdd/dual-Vt fabrics. *Proceedings of the 2004 ACM/SIGDA 12th international symposium on field programmable gate arrays*, Monterey, CA, FPGA '04 (pp. 42–50). New York, NY: ACM.

Li, H., Mak, W. K., & Katkoori, S. (2003). Efficient LUT-based FPGA technology mapping for power minimization. *Proceedings of the 2003 Asia and South Pacific design automation conference*, Kitakyushu, Japan, ASP-DAC '03 (pp. 353–358). New York, NY: ACM.

Lin, Y., & He, L. (2006). Dual-Vdd interconnect with chip-level time slack allocation for FPGA power reduction. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 25*(10), 2023–2034.

Linear Technology (2008). SwitcherCAD III/LTspice getting started guide. Milpitas, CA: Linear Technology.

Marconi, T., Theodoropoulos, D., Bertels, K., & Gaydadjiev, G. (2010a). A novel HDL coding style to reduce power consumption for reconfigurable devices, *Proceedings of the international conference on field-programmable technology (FPT)* (pp. 295–299). Washington, DC: IEEE.

Marconi, T., Theodoropoulos, D., Bertels, K., & Gaydadjiev, G. (2010b). A novel HDL coding style for power reduction in FPGAs. Technical Report, Computer Engineering Lab CE-TR-2010-02, Delft University of Technology, Delft.

Marconi, T. (2011). Efficient runtime management of reconfigurable hardware resources, PhD Thesis, Computer Engineering Lab, Delft University of Technology, Delft.

Markovic, D., Nikolic, B., & Brodersen, R. W. (2001). Analysis and design of lowenergy flip-flops. *Proceedings of international symposium on low power electronics and design* (pp. 52–55). New York, NY: ACM.

Matsumoto, Y., & Masaki, A. (2005). Low-power FPGA using partially low swing routing architecture. *Electronics and Communications in Japan (Part III: Fundamental Electronic Science), 88*(11), 11–19.

McCorkle, J. W., Huynh, P. T., & Ochoa, A., Method and apparatus for generating narrow pulse width monocycles. US Patent No. 7030663 B2 (2006).

Meng, Y., Sherwood, T., & Kastner, R. (2006). Leakage power reduction of embedded memories on FPGAs through location assignment. *Proceedings of the 43rd annual design automation conference*, San Francisco, CA, DAC '06 (pp. 612–617). New York, NY: ACM.

Mengibar, L., Entrena, L., Lorenz, M. G., & S´anchez-Reillo, R. (2003). State encoding for low-power FSMs in FPGA. *Proceeding of power and timing modeling, optimization and simulation* (Vol. 2799, pp. 31–40). Berlin: Springer.

Mondal, S., & Memik, S. O. (2005a). A low power FPGA routing architecture, *Proceedings of international symposium on circuits and systems* (pp. 1222–1225). Washington, DC: IEEE.

Mondal, S., & Memik, S. O. (2005b). Fine-grain leakage optimization in SRAM based FPGAs, *Proceedings of the ACM great lakes symposium on VLSI* (pp. 238–243). New York, NY: IEEE.

Mukherjee, R., & Memik, S. O. (2005). Evaluation of dual VDD fabrics for low power FPGAs, *Proceedings of Asia and South Pacific design automation conference* (pp. 1240–1243). Washington, DC: IEEE.

Nair, P., Koppa, S., & John, E. (2009, August). A comparative analysis of coarse-grain and fine-grain power gating for FPGA lookup tables, *Proceedings of IEEE international midwest symposium on circuits and systems* (pp. 507–510).

Nogawa, M., & Ohtomo, Y. (1998). A data-transition look-ahead DFF circuit for statistical reduction in power consumption, *IEEE Journal of Solid-State Circuits, 33*(5), 702–706.

Noguera, J., & Kennedy, I. O. (2007). Power reduction in network equipment through adaptive partial reconfiguration, *Proceedings of field-programmable logic and applications (FPL)* (pp. 240–245). Washington, DC: IEEE.

Onkaraiah, S., Gaillardon, P. E., Reyboz, M., Clermidy, F., Portal, J. M., Bocquet, M., & Muller, C. (2011). Using OxRRAM memories for improving communications of reconfigurable FPGA architectures. *Proceedings of the 2011 IEEE/ACM international symposium on nanoscale architectures*, NANOARCH '11. Washington, DC: IEEE Computer Society (pp. 65–69).

Park, S. R., & Burleson, W. (1998). Reconfiguration for power saving in real-time motion estimation. *Proceedings of international conference on acoustics, speech, and signal processing* (pp. 3037–3040). Washington, DC: IEEE.

Parlak, M., & Hamzaoglu, I. (2007). A low power implementation of H.264 adaptive deblocking filter algorithm. *Proceedings of the second NASA/ESA conference on adaptive hardware and systems*, AHS '07 (pp. 127–133). Washington, DC: IEEE Computer Society.

Patterson, C. (2000). A dynamic FPGA implementation of the Serpent Block Cipher, *Proceedings of the second international workshop on cryptographic hardware and embedded systems*, CHES '00 (pp. 141–155). London: Springer-Verlag.

Paulino, N., Goes, J., & Steiger-Garcao, A. (2008, May). A CMOS variable width short-pulse generator circuit for UWB RADAR applications, *IEEE international symposium on circuits and systems, 2008. ISCAS 2008* (pp. 2713–2716). Washington, DC: IEEE.

Paulsson, K., Hübner, M., & Becker, J. (2008, September). Exploitation of dynamic and partial hardware reconfiguration for on-line power/performance optimization, *Proceedings of reconfigurable communication-centric SoCs* (pp. 699–700). Washington, DC: IEEE.

Pi, T., & Crotty, P. J. (2003). FPGA lookup table with transmission gate structure for reliable low-voltage operation. US Patent No. 6667635.

Pontikakis, B., & Nekili, M. (2002). A novel double edge-triggered pulse-clocked TSPC D flip-flop for high-performance and low-power VLSI design applications, *IEEE international symposium on circuits and systems, 2002. ISCAS 2002* (Vol. 5, pp. V–101–V–104). Washington, DC: IEEE.

QuickLogic. (2008). Minimizing energy consumption with very low power mode in PolarPro solution platform CSSPs. *Application Note*, *88*. Sunnyvale, CA: QuickLogic.

Rahman, A., Das, S.,Tuan, T., &Trimberger, S. (2006, September). Determination of power gating granularity for FPGA fabric, *Proceedings of custom integrated circuits conference* (pp. 9–12). Washington, DC: IEEE.

Ramo, E. P., Resano, J., Mozos, D., & Catthoor, F. (2006). A configuration memory hierarchy for fast reconfiguration with reduced energy consumption overhead. *Proceedings of the 20th international conference on parallel and distributed processing*, Rhodes Island, Greece, IPDPS'06 (pp. 189–189). Washington, DC: IEEE Computer Society.

Ravindra, J., & Anuradha, T. (2012). Article: Design of low power RISI processor by applying clock gating technique. *International Journal of Applications, 43*(20), 38–42, Published by Foundation of Computer Science, New York, USA.

Rivoallon, F. (2010). Reducing switching power with intelligent clock gating, Xilinx WP370 (v1.1).

Rollins, N., & Wirthlin, M. J. (2005). Reducing energy in FPGA multipliers through Glitch reduction. *Proceedings of international conference on military applications of programmable logic devices* (pp. 1–10). Washington, DC: NASA.

Roy, K., Mukhopadhyay, S., & Mahmoodi-Meimand, H. (2003). Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits, *Proceedings of the IEEE, 91*(2), 305–327.

Saleem, A., & Khan, S. (2010, August). Low power state machine design on FPGAs. 2010 *3rd international conference on advanced computer theory and engineering (ICACTE)* (Vol. 4, pp. V4–442–V4–445). Washington, DC: IEEE.

Sayed, A., & Al-Asaad, H. (2006, August). A new low power high performance flip-flop. *49th IEEE international midwest symposium on circuits and systems, 2006. MWSCAS '06* (Vol. 1, pp. 723–727). Washington, DC: IEEE.

Singh, A., & Marek-Sadowska, M. (2002). Efficient circuit clustering for area and power reduction in FPGAs. *Proceedings of the 2002 ACM/SIGDA tenth international symposium on field-programmable gate arrays*, Monterey, CA, FPGA '02 (pp. 59–66). New York, NY: ACM.

Sterpone, L., Carro, L., Matos, D., Wong, S., & Fakhar, F. (2011). A new reconfigurable clock-gating technique for low power SRAM-based FPGAs, *DATE* '11 (pp. 752–757). Washington, DC: IEEE.

Sutter, G., Todorovich, E., & Boemo, E. (2004). Design of Power Aware FPGA based Systems, *Proceedings of Jornadas de Computación Reconfigurable y Aplicaciones (JCRA)* (pp. 1–10).

Sutter, G., Todorovich, E., Lpez-Buedo, S., & Boemo, E. I. (2002a). FSM decomposition for low power in FPGA, *Proceedings of IEEE international conference on field-programmable logic and applications (FPL)* (Vol. 2438, pp. 350–359). Washington, DC: IEEE.

Sutter, G., Todorovich, E., Lpez-Buedo, S., & Boemo, E. I. (2002b). Low-power FSMs in FPGA: Encoding alternatives. *Proceeding of power and timing modeling* (Vol. 2451, pp. 363–370). Berlin: Springer.

Teh, C. K., Hamada, M., Fujita, T., Hara, H., Ikumi, N., & Oowaki, Y. (2006). Conditional data mapping flip-flops for low-power and high-performance Systems. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 14*(12), 1379–1383.

Tessier, R., Swaminathan, S., Ramaswamy, R., Goeckel, D., & Burleson, W. (2005). A reconfigurable, power-efficient adaptive Viterbi decoder. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 13*, 484–488.

Tinmaung, K. O., Howland, D., & Tessier, R. (2007). Power-aware FPGA logic synthesis using binary decision diagrams. *Proceedings of the 2007 ACM/SIGDA 15th international symposium*

on field programmable gate arrays, Monterey, California, USA, FPGA '07 (pp. 148–155). New York, NY: ACM.

Uuger, S. H. (1981). Double-edge-triggered flip-flops. *IEEE Transaction on Computation*, *30*, 447–451.

Vorwerk, K., Raman, M., Dunoyer, J., Hsu, Y., Kundu, A., & Kennings, A. A. (2008). A technique for minimizing power during FPGA placement. *Proceedings of international conference on field programmable logic and applications* (pp. 233–238). Washington, DC: IEEE.

Wang, H., Miranda, M., Papanikolaou, A., Catthoor, F., & Dehaene, W. (2005). Variable tapered pareto buffer design and implementation allowing run-time configuration for low-power embedded SRAMs. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 13*, 1127–1135.

Wang, L., French, M., Davoodi, A., & Agarwal, D. (2006). FPGA dynamic power minimization through placement and routing constraints, *EURASIP Journal on Embedded Systems, 2006*, 7–7. Berlin: Springer.

Wang, Q., Gupta, S., & Anderson, J. H. (2009). Clock power reduction for virtex-5 FPGAs. *Proceedings of ACM/SIGDA international conference on field programmable gate arrays (FPGA)* (pp. 13–22). New York, NY: ACM.

Wang, Z. H., Liu, E. C., Lai, J., & Wang, T. C. (2001). Power minimization in LUT-based FPGA technology mapping. *Proceedings of the Asia and South Pacific design automation conference* (pp. 635–640). Washington, DC: IEEE.

Wilton, S. J. E., shin Ang, S., & Luk, W. (2004). The impact of pipelining on energy per operation in field-programmable gate arrays. *Proceedings of field programmable logic and applications* (pp. 719–728). Berlin: Springer-Verlag.

Wolff, F., Knieser, M., Weyer, D., & Papachristou, C. (2000). High-level low power FPGA design methodology. *Proceedings of the IEEE National aerospace and electronics conference* (pp. 554–559). Washington, DC: IEEE.

Wu, X., & Pedram, M. (2001). Low-power sequential circuit design using T flip-flops. *International Journal of Electronics, 88*, 635–643.

Xilinx. (2002). Low power design with coolrunner-II CPLDs. XAPP377.

Xilinx. (2012). Virtex-6 FPGA clocking resources. UG362.

Yang, S. (1991). Logic Synthesis and Optimization Benchmarks User Guide Version 3.0, *Proceedings of international workshop on logic and synthesis*. Washington, DC: IEEE.

Yuan, T., Zheng, Y., Ang, C. W., & Li, L. W. (2007, June). A fully integrated CMOS transmitter for ultra-wideband applications. *Radio frequency integrated circuits (RFIC) symposium, 2007 IEEE* (pp. 39–42).

Zhang, Y., Roivainen, J., & Mämmelä, A. (2006). Clock-gating in FPGAs: A novel and comparative evaluation. *Proceedings of EUROMICRO conference on digital system design* (pp. 584–590). Washington, DC: IEEE.

Zhao, P., Darwish, T., & Bayoumi, M. (2002, August). Low power and high speed explicit-pulsed flip-flops. *The 2002 45th midwest symposium on circuits and systems, 2002. MWSCAS-2002* (Vol. 2, pp. II–477–II–480).

Zhao, P., Darwish, T. K., & Bayoumi, M. A. (2004). High-performance and lowpower conditional discharge flip-flop. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 12*(5), 477–484.

Zhao, W., Belhaire, E., Chappert, C., & Mazoyer, P. (2009). Spin transfer torque (STT)-MRAM–based runtime reconfiguration FPGA circuit. *ACM Transactions on Embedded Computing Systems, 9*(2), 14: 1–14:16.