Research Article

Power Characterisation for Fine-Grain Reconfigurable Fabrics

Tobias Becker,[1] Peter Jamieson,[1] Wayne Luk,[1] Peter Y. K. Cheung,[2] and Tero Rissa[3]

[1] *Department of Computing, Imperial College London, London SW7 2AZ, UK*
[2] *Department of EEE, Imperial College London, London SW7 2AZ, UK*
[3] *Nokia Devices R&D, Tampere, Finland*

Correspondence should be addressed to Tobias Becker, tbecker@doc.ic.ac.uk

This paper proposes a benchmarking methodology for characterising the power consumption of the fine-grain fabric in reconfigurable architectures. This methodology is part of the GroundHog 2009 power benchmarking suite. It covers active and inactive power as well as advanced low-power modes. A method based on random number generators is adopted for comparing activity modes. We illustrate our approach using five field-programmable gate arrays (FPGAs) that span a range of process technologies: Xilinx Virtex-II Pro, Spartan-3E, Spartan-3AN, Virtex-5, and Silicon Blue iCE65. We find that, despite improvements through process technology and low-power modes, current devices need further improvements to be sufficiently power efficient for mobile applications. The Silicon Blue device demonstrates that performance can be traded off to achieve lower leakage.

1. Introduction

Rapidly evolving standards, convergence of increasingly complex features, and growing time to market pressure are pushing manufacturers of mobile consumer devices to consider alternatives to ASICs and microprocessors. There is a clear demand for power efficient circuits that are flexible, while capable of delivering performance through parallelism. Reconfigurable architectures, such as FPGAs, have good potential in meeting the demand for flexibility and performance, but they often miss the power requirements by up to several orders of magnitude. They can consume roughly ten times more active power than ASICs [1]. Moreover, they can draw inactive power that is one hundred to a thousand times higher than what is permitted for mobile devices in standby mode [2]. Devices can also cause thermal problems when they heat up during intense processing.

One of the problems with optimising for low power in FPGAs is that a designer has to map a particular application onto a range of target devices and evaluate their power consumption by using the power estimation tools provided with some FPGA CAD flows. But these estimation tools have often limited accuracy. Moreover, synthesising, implementing, and simulating a design on a range of devices can be very time consuming. The result of such an evaluation is only meaningful for this particular design. Instead, it would be desirable to have a set of test cases that can be used to benchmark the power characteristics of reconfigurable devices. A list of test results would allow users to choose suitable devices for a low-power design and the procedure can also be used to evaluate architectural improvements.

In order to address these issues we have developed the GroundHog 2009 power benchmarking suite for reconfigurable architectures [3]. GroundHog 2009 includes one application independent benchmarking technique, which is presented in this paper. In addition, GroundHog 2009 also provides six application specific test cases which are representative for mobile devices. These benchmarks are described in [4]. In this paper, we focus on the application independent method that allows a fast and easy evaluation and classification of the power consumption in fine-grain FPGAs. The classification is based on several activity modes which reflect realistic high-power or low-power scenarios a device could be operated in. Another important aspect is thermal characteristics such as the temperature dependency of static power and heating up of the device under different processing scenarios. The proposed methodology is intended to provide a fair comparison of existing devices, and should also be able to capture improvements in future devices with new low-power features.

The goal of this benchmark is to provide a simple technique for evaluating and comparing the power efficiency of devices. Such an evaluation will always be influenced by external factors such as implementation tools and measuring environment, although our benchmarking technique minimises these influences as much as possible. It is also important to point out that this paper describes an example of the benchmark implementation with regard to some of its parameters such as clock frequency or logic utilisation. But a benchmark user may find that a variation of these parameters might be more appropriate for their purposes. This will come at the expense of having limited comparability with other benchmark users but might be more appropriate for evaluating devices in a particular domain.

The remainder of this paper is organised as follows. Section 2 describes previous work. Section 3 proposes a method for power characterisation of fine-grain reconfigurable fabrics. Section 4 describes how this method can be implemented on commercial FPGAs, and Section 5 shows some results of our power characterisation. Finally, Section 6 concludes the paper.

2. Background

In this section we consider some of the general aspects of power consumption as well as their dependencies and trends. We further review previous research on power consumption in FPGAs. Additionally, we review existing FPGA architectures and techniques that are designed for low power.

2.1. Power Consumption in CMOS Devices. The total power consumption in CMOS devices is a combination of static and dynamic power. Static power is caused by leakage currents inside the device while dynamic power is caused by switching activity.

Static power has two main components: gate leakage and subthreshold leakage, with the latter being the most significant component [5]. Gate leakage is a leakage current from the transistor gate through the gate oxide into the substrate. Subthreshold leakage or source-to-drain leakage is a leakage current from the transistor source to drain when the transistor is turned off and the gate voltage is below the threshold voltage. Lowering the threshold voltage, which increases performance, causes subthreshold leakage to grow exponentially. The subthreshold leakage also increases exponentially with the junction temperature. Static power used to be a minor component of the total power consumption. But lower threshold voltages in modern CMOS devices have led to a growth of this component. Static power can be addressed by using thicker gate oxides for noncritical path transistors, back-biasing the substrate or providing standby modes that turn off circuits during periods of inactivity. Static power also has a strong dependency on the variation of process parameters. Hence, the static power profile of a device can vary from die-to-die, wafer-to-wafer, and lot-to-lot [6].

Dynamic power on the other hand is caused by a combination of charging and discharging load capacitances as well as short-circuit currents when transistors switch. Charging and discharging capacitances are usually the dominant effect [5]. Dynamic power is given by (1). It has linear dependency on the clock frequency f and the capacitance C, and a quadratic dependency on the supply voltage V. In general, the capacitance is defined by all transistors in the device. In an FPGA however, it depends on the number of logic and routing elements used in a particular configuration. The factor α is the activity or toggle rate and depends on the design and its input stimuli. Dynamic power is independent of temperature and also does not depend as strongly on process variation as static power

$$P_{\text{dynamic}} = \alpha \cdot C \cdot V^2 \cdot f. \tag{1}$$

Dynamic power is improved by reducing the transistor capacitances through feature scaling and reducing the voltage through voltage scaling. Additional improvements can be made by reducing the switching activity in the device. For example, the clock frequency can be scaled according to processing requirements or the clock can be stop completely during inactive periods.

2.2. FPGA Power Research. One of the earliest comparative studies for power consumption on FPGAs is done by George et al. [7]. The authors create a low-power FPGA through architecture and low-level circuit design, and compare their FPGA to Xilinx and Altera devices. The comparison is based on three test circuits that are evaluated with Synopsis Powermill. The three circuits consist of a single flip-flop driving 9 routing segments, a 1 K array of 16 bit counters, and a toggle circuit.

Shang et al. [8] measure the dynamic power consumption of a Xilinx Virtex-II FPGA [9] using one Xilinx internal benchmark that represents a large industrial circuit. Using this internal benchmark and input stimuli, they calculate the switching activity of the design. They estimate power based on the calculated switching activity and the effective capacitance of each resource on the FPGA. This is possible since they have access to low-level models of the FPGA, but such models are usually not accessible.

Gayasen et al. propose an FPGA architecture with two supply voltages where the lower voltage is used for all noncritical path components [10]. The efficiency of their architecture is evaluated with MCNC benchmarks which provide a range of simple circuits and state machines.

Tessier et al. show that the power efficiency of an application can be improved by reconfiguring between a more powerful, less efficient core and a less powerful, more efficient core on demand [11]. Becker et al. analyse the power consumption during reconfiguration [12].

Recently, Kuon and Rose [1] assess the gap between FPGAs and ASICs. This work includes an attempt to measure the dynamic and static power consumption gap between the two technologies. They estimate the static and dynamic power consumption of an FPGA using the power estimation tools provided by the FPGA vendor, and use either included testbenches or estimates of net activity.

TABLE 1: Comparison of devices and low-power features.

Device manufacturer	Xilinx	Lattice	Actel	Silicon Blue
Device Name	Spartan-3A, Spartan-3AN	Mach XO	IGLOO	iCE65
Device size [LUTs]	1.6 k–53 k	256–2.3 k	192–36 k (∗)	1.8 k–17 k
Low-power feature	Suspend mode	Sleep mode	Flash freeze	iCEgate
Wake-up time [μs]	100–500	1000	1	Instant
Retain state	Yes	No	Yes	Yes

(∗) device does not provide LUT/FF pairs like all other devices, reconfigurable tiles can be used as LUT or FF.

FPGA fabrics have also been characterised in terms of their thermal characteristics and die variation. Lopez-Buedo et al. [13] use ring oscillators programmed onto an FPGA. These oscillators are placed around an existing mapped design to measure the temperature of the fabric when the design is operating. The local temperature is determined by measuring the frequency of the ring oscillator.

FPGA clock networks can be responsible of a significant amount of power consumption. Lamoureux and Wilton examine clock-aware placement techniques, and investigate the tradeoff between the power consumption of the clock network and the impact of the constraints imposed by the design automation tools [14].

2.3. Low-Power FPGAs. Tuan et al. present an experimental low-power FPGA with several power optimisations such as voltage scaling, leakage reduction of configuration memory cells, and power gating of tiles with preservation of state and configuration [2]. An implementation on a 90 nm process technology demonstrates a reduction of 46% in active power and 99% in standby power. These values are not specific to a particular FPGA configuration.

Most current commercial FPGAs have limited low-power capabilities. Traditionally, FPGA power optimisations target the operational power in order to reduce the complexity of the power supply and eliminate extra cooling systems. Xilinx introduced triple-oxide technology, which uses less leaky transistors with thicker gate oxide for configuration memory cells and interconnect pass gates [15]. Altera developed a programmable transistor back-biasing feature that allows selecting high-performance logic with high power consumption only for timing-critical components in the design [16]. But these improvements alone are insufficient to enable the use of FPGAs in mobile, battery-based applications. Such a scenario would require dedicated low-power modes that reduce the total power consumption by several orders of magnitude during periods of inactivity. With most current devices, the only methods of saving power during inactivity are to employ clock gating or to turn off the entire device. The first method has limited potential whereas in the latter case, state and configuration are lost.

Recently, some devices with additional low-power capabilities were introduced. Xilinx Spartan-3A and Spartan-3AN FPGAs support a suspend mode in which auxiliary clock circuitry can be stopped and powered down [17]. Lattice Mach XO devices provide a sleep mode [18] that can reduce the standby power by a factor of 100. However, the application state is lost when using this sleep mode. Another example is Actel IGLOO devices which feature a sleep state that does retain the application state [19]. Silicon Blue provides iCE65 FPGAs [20] that are optimised for low static power. Table 1 shows a comparison of these devices and their low-power features.

3. Fabric Characterisation Method

When developing a general characterisation method one faces a number of challenges. The method should:

(i) be applicable to a wide range of devices,

(ii) be a fair comparison and results free from implementation tool influences or hand optimisations,

(iii) allow to capture different power modes and possible future techniques that are currently not available.

The basic idea of our method is to implement a highly active circuit on the FPGA and measure the power in several activity modes. These activity modes represent active processing, inactivity and dedicated low-power states. This method allows us to characterise power consumption quickly in best-case and worst-case scenarios and it outlines the suitability of a device for low-power design. Furthermore, we consider the thermal aspects of the device such as heating up under active processing.

The benefit of this characterisation is that it allows us to assess the adequacy of a device for a low-power design. Moreover, it can be used to compare and optimise devices for lower power. The key aspects of our method can be summarised as follows.

(i) Use random number generators (RNGs) as test circuit with high activity.

(ii) Use 90% of the logic resources in the device.

(iii) Run the test circuit at a fixed clock rate of 100 MHz when active.

(iv) Specify the behaviour of activity modes and switch between these with various duty cycles.

(v) Measure power and temperature in these modes.

To create a worst case active processing scenario, we use pseudo random number generators as test circuits [21]. These random number generators are based on binary linear recurrence where each bit of the next state is generated based on a linear combination of the current state. Compared to linear feedback shift registers (LFSRs), the most common type of random number generators, this improves quality of the random numbers. But for our purposes, the main advantage is the lack of optimisation potential in the circuit. This will minimise the influence of the implementation tools on the result of our characterisation. Using binary linear recurrence yields a circuit where each RNG state flip-flop is fed by a LUT which has its inputs connected to some other state flip-flops. An n-bit RNG will therefore map to exactly n LUTs and n flip-flops. This circuit does not provide any potential for logic optimisation and thus, eliminates the influence of the synthesis tools. The circuit is also characterised by LUTs that are heavily interconnected to seemingly random points. Hence, it does not provide any opportunity for optimised placement and routing other than concentrating the RNG circuit to the smallest possible area. It will also result in an implementation that will exercise all different kinds of short and long wires of the routing fabric.

The random number generator circuit is also characterised by a high and uniformly distributed toggle rate, and therefore suitable to act as worst case scenario of maximum activity in the fine grain fabric. The statistical chance of toggling on the rising clock edge is 50% for all flip-flops in the circuit. Hence, the total toggle rate of the circuit is 50%.

Currently, we use a 512-bit random number generator core that maps to 512 LUTs and 512 flip-flops. Since current FPGAs provide tens to hundreds of thousands of LUTs and flip-flops, we scale the size of the test circuit with the size of the device. To achieve high logic utilisation while still allowing routability, we implement multiple instances of the random number generators so that 90% of all logic resources are used. The resulting power consumption is normalised to the number of LUTs in order to allow a comparison of differently sized chips. If, however, 90% logic utilisation should lead to routing congestion and prevent the implementation tools from completing the design on other target devices, then the utilisation can be lowered. This should only affect active power and there should be minimal to no influence on inactive power or power in low-power modes.

The cores are driven by a 100 MHz clock when the circuit is active. This frequency simply acts as a reference point for a typical FPGA clock frequency. The power characteristics for different clock frequencies can be estimated by scaling the power consumption linearly to the clock rate. However, a benchmark user can also decide to run the circuit at a different clock frequency if the suggested 100 MHz clock seems unrealistic for their purposes. Varying utilisation or clock frequency limits the comparability with other published benchmark results but may yield in a more realistic evaluation in a particular domain.

To enable a comparison of devices with different low-power capabilities, we define the behaviour of activity modes. These activity modes specify how the device behaves in a

Table 2: Examples of activity modes. The first two modes are fixed, further modes can be defined based on the device capabilities.

	Activity mode			
	Standard		Device-specific	
	Active	Inactive	Sleep	Hibernate
Generate output	Yes	No	No	No
Retain state	—	Yes	Yes	No
Wakeup time	—	Instant	$500\,\mu s$	50 ms

certain mode rather than by which means this mode is implemented. The two basic modes that are applicable to all devices are *active* and *inactive*. In *active* mode, the test circuit continuously generates random numbers at 100 MHz clock frequency. The power consumption in this mode is a combination of static and high dynamic power. In *inactive* mode, the circuit does not generate random numbers. However, its state is preserved and it can be instantly brought back into active mode. These are the most basic activity modes and a transition between these two modes can usually be implemented with a simple clock gating approach as illustrated in Figure 1. However, we are only concerned about the power profile of the inactive device with preservation of state and instant wake-up capability, and not the details of its technical implementation. This mode corresponds to static power only, if clock gating is chosen as a method to implement this mode. However, depending on the implementation details, there might be supporting circuitry such as clock managers that are still operating and drawing dynamic power. It is important to point out that in this context, we are not necessarily interested in measuring pure static power but rather the minimal power to implement the *inactive* mode.

In order to evaluate devices with advanced low-power modes such as the ones in Table 1, we characterise the behaviour of the low-power mode and compare the power consumption with the two basic modes. Table 2 illustrates an example with our two basic activity modes *active* and *inactive*, and two hypothetical advanced modes *sleep* and *hibernate*. The behaviour of the basic modes is fixed, while the behaviour of the advanced modes depends on the device capabilities.

Since device temperature has a feedback effect on static power, we define a specific environment in order to reduce external influences on the measurement. The test environment is specified as a chip mounted on a PC board surrounded by ambient air with a temperature of 25°C. The board is placed in a large open cardboard enclosure which is supposed to reduce airflow to natural convection and reflect infrared radiation. No heatsinks or active cooling systems are to be used.

In the following, we distinguish between *cold* and *hot* devices in our characterisation. Devices which do not exceed a surface temperature of 35°C when being in active mode are characterised as *cold*. In this case, the influence of temperature is small and can be neglected. The power is measured in each activity mode and the results are reported normalised to LUTs.
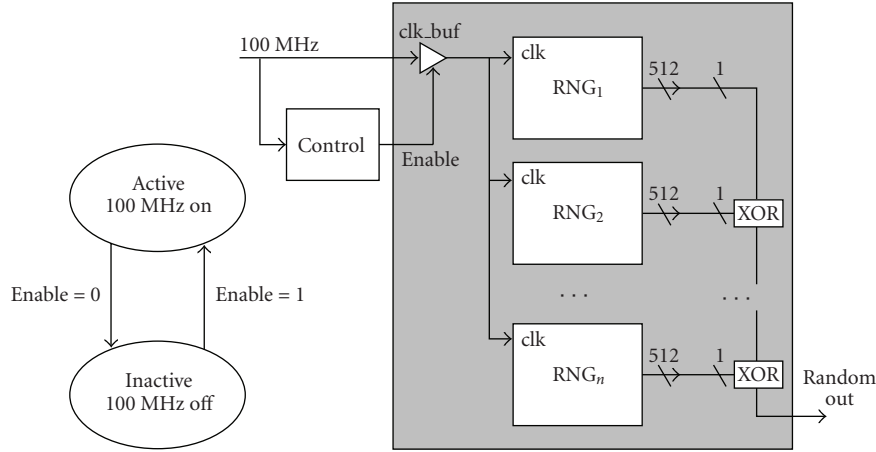
FIGURE 1: Fabric characterisation circuit implemented with random number generators.

For *hot* devices, we propose a more detailed analysis that takes the heating up of the device and its influence on power into account. Even though mobile applications are characterised by strict low-power constraints, thermal considerations are also important. In addition to power constraints, devices also have a thermal budget. Hence, we want to analyse how a device heats up for a given amount of activity.

Thermal characterisation of a device usually involves determining the thermal resistance of the device between die (also called junction) and case or junction and ambient air. For this purpose, a special test die is usually mounted in a case and heated up to a defined junction temperature T_j by applying power P. The junction-to-case thermal resistance θ_{jc} can then be calculated based on the following equation:

$$\theta_{jc} = \frac{T_j - T_c}{P}. \qquad (2)$$

Equation (2) can also be used to calculate the case temperature T_c if θ_{jc} is specified by the device manufacturer. This however requires the knowledge of the junction temperature T_j. A further disadvantage is that (2) is not very accurate since it does not consider heatflow into the PC board or the ambient air. As an alternative, we propose to measure the case temperature as well as power consumption in our well-defined environment over the device activity. The activity in the device is adjusted by periodically switching the device between active and inactive state with various duty cycles. For each duty cycle we measure instantaneous active power, inactive power and temperature. To take these measurements, we first wait until the temperature has converged to its final value as illustrated in Figure 2. We record this temperature and then take a reading of active and inactive power. As mentioned earlier, inactive power does not necessarily have to be equivalent to static power and can have dynamic components as well. Nonetheless, we expect a strong temperature dependency on inactive power because of its close relation to static power. Likewise, we record the instantaneous active power that we expect to be less temperature dependent, since it is largely based on dynamic
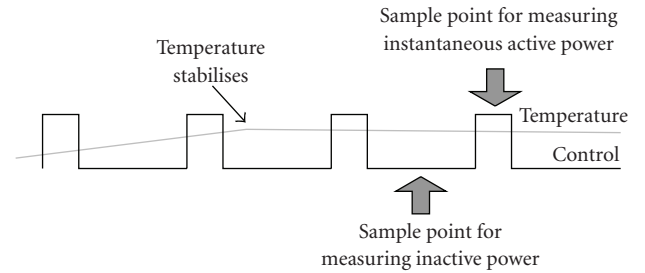

FIGURE 2: Depiction of power measurement sample points relative to temperature.

power. These measurements are taken for duty cycles from 0% to 100% with 5% increments. Results for temperature and normalised active and inactive power are reported over duty cycle. For devices with additional low-power modes, the power in these modes is measured at ambient temperature without any duty cycle.

4. Implementation of the Fabric Characterisation Method

We implement our fabric characterisation in Xilinx Virtex and Spartan series FPGAs and one Silicon Blue iCE65 FPGA. The synthesis and mapping of the random number generator circuit for these devices is straightforward. Xilinx Virtex series and Spartan-3E FPGAs do not support any low-power modes other than simply stopping the clock. The most efficient way of setting the device into inactive mode as specified in Section 3 is to disable the clock tree using an internal clock buffer. This buffer is located at the root of the clock tree and therefore reduces the dynamic component of inactive power to a minimum. The enable signal of the clock buffer is connected to an external pulse generator with variable duty cycle. Spartan-3A and Spartan-3AN FPGAs also feature an advanced low-power mode called *suspend* [22]. This mode is controlled by an external signal pin and does not require modifications to the logic design itself.

Silicon Blue iCE65 FPGAs provide an iCEgate latch which will stop input signals from propagating into the device. In our case we use the iCEgate as a clock buffer, which is again controlled by an external pulse generator.

Figure 1 illustrates the implementation of our test circuit. The RNG cores only have a clock input and a 512-bit wide output. Each core is initialised with a different seed and one output pin of each core is connected via an XOR chain to avoid logic optimisation of the circuit. The reported LUT usage of the implementation tools should match the expected value of $512 \cdot n$, where n is the number of instantiated RNGs.

The power consumption of an FPGA can be obtained by measuring the current on the supply rails. However, it should be considered that FPGAs are sensitive to core voltage variations and usually have to stay within 50 mV of the nominal value. The voltage drop caused by the internal shunt resistor of a current meter can exceed this value if high currents are measured. It is therefore not recommended to wire a current meter directly into the supply rail. As an alternative, we insert precision current sense resistors directly into the rail and measure the voltage drop over the resistor. Our current sense resistors are in the milliohm range and have 1% tolerance. The resistor value is selected such that the voltage drop at maximum current is less than 50 mV. We measure the voltage drop over the resistor with a Hameg HM8012 digital multimeter and use this value to calculate current and power. Most FPGAs provide different voltage rails for core logic and IOs. For our fabric characterisation we only measure core power because IO power is highly dependent on pin loads of a given design.

We use an Optris MiniSightPlus infrared thermometer to determine the surface temperature of the device. This contactless method minimises the influence of the measurement on the system. Infrared thermometers work very well on matt, black plastic cases. However, they are inaccurate on shiny metal surfaces since these emit less infrared radiation. We therefore apply a thin blackened aluminium plate to devices with a metallic surface.

5. Results and Measurements

We perform a fabric characterisation as described in Section 3 on five commercially available FPGAs. Our measurements cover four Xilinx devices and one Silicon Blue device as listed in Table 3. The tested devices vary notably in their process technology and core voltage and thus have significantly different power profiles. Table 3 also shows the logic capacity of each device, the number of RNG cores used, and the resulting device usage which should be as close as possible to 90%. Both Spartan devices have less than half of the logic capacity than the Virtex devices. The Silicon Blue device is the smallest device with significantly less logic capacity than all others. The Spartan devices have plastic packages which have higher thermal resistances. This leads to increased junction temperatures for a given surface temperature. The Xilinx Virtex-II Pro and the Silicon Blue device have ceramic cases which provide better thermal conductivity than plastic. However the exact thermal

resistance is not specified for the Silicon Blue device. Xilinx Virtex-5 FPGAs have metal cases which provide the best thermal conductivity. All boards have current measurement facilities except the ML505 board which was specifically modified to allow equally precise measurements. The designs targeting Xilinx device are implemented with Xilinx ISE 9.2. The design for the Silicon Blue FPGA is implemented with Silicon Blue iCEcube 2008 development software.

Table 4 shows the total and normalised inactive and active power for all devices. The table also lists the minimum temperature the device reaches when constantly being inactive and the maximum temperature the device reaches with full duty cycle of active processing. Inactive power at minimum temperature represents the case when the device is held in inactive state for a longer period of time and hence, is cooled down whereas inactive power at maximum temperature represents the case where the device is just switched off from active processing. Virtex-II Pro is the only device that cannot be run at full duty cycle without overheating. At 60% duty cycle, the device reaches a surface temperature of 74°C. Based on the thermal resistance and the power consumption, we estimate that this surface temperature corresponds to the maximum allowed junction temperature of 85°C. All other devices can be run at full duty cycles without overheating. The minimum and maximum case temperatures are shown in columns two and three. Table 4 also lists the total and normalised active power for maximum temperature, since this is the temperature the device operates under for full duty cycle.

The Silicon Blue iCE65 FPGA does not exceed a temperature of 29°C. The inactive and active power is not influenced by any temperature feedback of activity and the device can therefore be treated as a cold device. The low temperature can be explained with good thermal conductivity of the case and overall low total power. The normalised active power of the iCE65 device is similar to Virtex-5 which is manufactured in the same process technology. The inactive power in the iCE65 device is measured using iCEgate latch feature which can freeze the state of an IO bank. Outputs are kept at their current state and input signals do not propagate into the device. Internal switching activity is therefore stopped completely. This is not a dedicated low-power state by itself, but contributes to good power efficiency when being inactive. The inactive power of the iCE65 FPGA is significantly better than all other devices which can be explained by the prevention of internal switching activity by the iCEgate and the fact that the device is specifically optimised for low leakage. Optimising for low leakage however comes at the expense of performance. It is challenging to achieve timing closure for our test design on the iCE65 FPGA while all other Xilinx devices provided good performance headroom. For all Xilinx devices, we can observe a temperature feedback of the active processing on inactive power. For Virtex-II Pro, we find a ninefold increase of inactive power between minimum and maximum temperature. In Xilinx devices, we can also observe an improvement of normalised active power with modern process technology while at the same time, inactive power deteriorates. In the following we further analyse the four Xilinx devices using our duty cycle measurement method.

TABLE 3: Details of FPGAs used for our fabric characterisation experiment.

FPGA family	Device	Board	Process technology	Core voltage	Thermal resistance Θ_{JC}	Number of LUTs/FFs	Number of RNGs	Logic utilisation
Xilinx Virtex-II Pro [9]	XC2VP30	XUP	130 nm	1.5 V	0.6°C/W	27,392	48	89.7%
Xilinx Spartan-3E [17]	XC3S500E	Spartan-3E Starter Kit	90 nm	1.2 V	9.8°C/W	9,312	16	88%
Xilinx Spartan-3AN [17]	XC3S700AN	Spartan-3AN Starter Kit	90 nm	1.2 V	5.3°C/W	11,776	21	91.3%
Xilinx Virtex-5 [23]	XC5VLX50T	ML505	65 nm	1.0 V	0.2°C/W	28,800	50	88.9%
Silicon Blue iCE65 [20]	iCE65L04	iCEman Eval Kit	65 nm	1.2 V	n/a	3,520	6	87.3%

TABLE 4: Power measurement results. Shown is total and normalised inactive and active power.

| Device | T_{min} | T_{max} | Inactive mode | | | | Active mode | |
			P_{total} (T_{min}) [mW]	P_{norm} (T_{min}) [μW/LUT]	P_{total} (T_{max}) [mW]	P_{norm} (T_{max}) [μW/LUT]	P_{total} (T_{max}) [W]	P_{norm} (T_{max}) [μW/LUT]
V2P30	26°C	74°C	24.0	0.88	216.0	7.89	11,190	455.3
S3E500	26°C	46°C	19.2	2.06	26.2	2.81	1,022	124.8
S3AN700	26°C	47°C	18.7	1.59	28.8	2.45	1,363	126.8
V5LX50T	30°C	54°C	349.0	12.12	412.1	14.31	2,545	99.4
iCE65L04	25°C	29°C	0.13	0.037	0.13	0.037	324	105.5

Figure 3 shows the surface temperature of the Xilinx devices over the duty cycle. The temperature is measured with an infrared thermometer as explained in Section 4. For all devices the temperature increases almost linearly. Virtex-II Pro has the steepest slope and reaches a surface temperature of 74°C at 60% duty cycle. Spartan-3E, Spartan-3AN and Virtex-5 reach a surface temperature of 46°C, 47°C, and 54°C, respectively, when on full duty cycle.

Figure 4 illustrates the inactive power consumption normalised per LUT over duty cycle. Each point on the line is measured when the temperature of the FPGA has stabilized for a given duty cycle as illustrated in Figure 2. We observe that inactive power consumption, which in our implementation is almost equivalent to static power, increases with the duty cycle. This is due to the rising temperature of the device caused by heat dissipated during the active part of the duty cycle. The Virtex-5 device, which is manufactured in 65 nm process, has a 12 times higher inactive power consumption under cold conditions (0% duty cycle) than the 130 nm Virtex-II Pro. This can be explained with worsened static power in the smaller process technology but unused hard IP blocks may also have an influence. The inactive power in Virtex-II Pro deteriorates quickly with higher duty cycles because of the high device temperature as illustrated in Figure 3. Virtex-5 and the Spartan-3 devices develop less heat which leads to a less progressive increase in inactive power. Of all four Xilinx FPGAs, Spartan-3 devices show the overall best efficiency during inactive phases.

Figure 5 shows the normalised active power consumption for all Xilinx devices. This is the instantaneous active power during on-phase of the duty cycle as illustrated in Figure 2. We can observe a notable improvement in active power for newer devices which is due to feature and voltage scaling in the process technology. The improvement from Virtex-II Pro (130 nm) to Spartan-3 (90 nm) is especially noteworthy. Compared to Virtex-5, the active power is reduced by more than a factor of 4. The active power consumption is relatively independent of duty cycle and temperature, although, this could change if static power becomes a more dominant component in active power. In current devices, we find that inactive power is considerably less than active power although the ratio increases from 0.2% to 14% between Virtex-II Pro and Virtex-5 in cold conditions, or from 1.5% to 16% in hot conditions. On the other hand, devices can be optimised for low leakage by trading off performance. The Silicon Blue device, for example, has an inactive-to-active power ratio of 0.04%.

The only device in our test that features a dedicated low-power mode is Spartan-3AN that provides a suspend mode. This mode reduces the power consumption of all auxiliary circuits powered on the VCC$_{aux}$ rail [22]. The logic state is preserved during suspend mode and the wake-up time
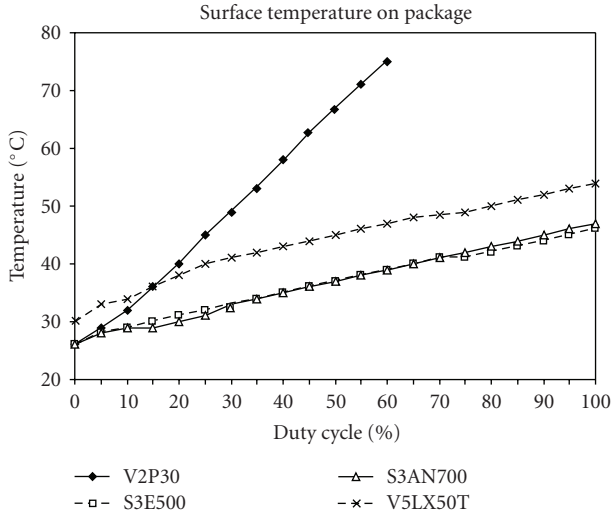
FIGURE 3: Variation of device surface temperature with duty cycle.

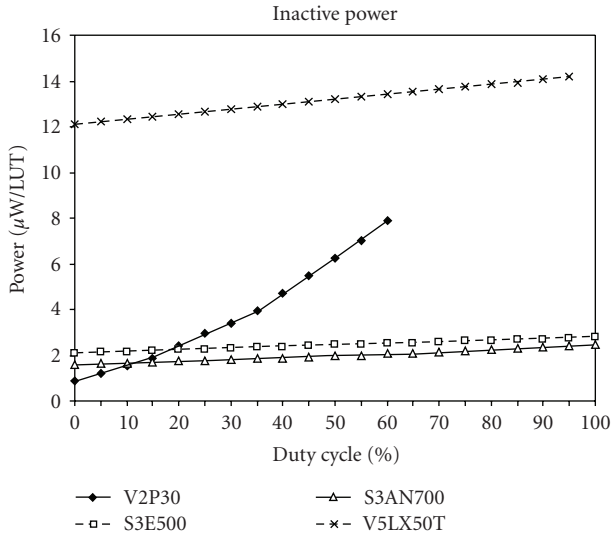FIGURE 5: Variation of instantaneous active power consumption with duty cycle.

FIGURE 4: Variation of inactive power consumption with duty cycle.

TABLE 5: Total core power in a Xilinx Spartan-3AN 700 FPGA for active, inactive, and suspend modes. All values are measured at 25°C.

Power	Active mode	Inactive mode	Suspend mode
P_{int} [mW]	1349	18.7	18.1
P_{aux} [mW]	44	43.6	5.8
P_{total} [mW]	1393	62.3	23.9

ranges between $100 \mu s$ and $500 \mu s$. Table 5 illustrates the core power consumption in all modes. Compared to the inactive mode, the suspend mode reduces the power consumption by a factor of 3.

In an overall comparison, the Silicon Blue iCE05 FPGA is the most power efficient device. It consumes slightly more active power than Virtex-5 but significantly less inactive power than all other devices. However, it has the lowest
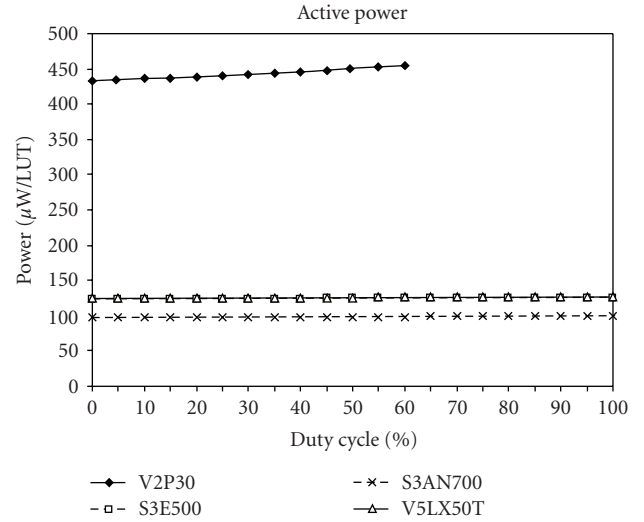
logic capacity and was specifically optimised for low leakage and has therefore less performance. The device performance is not addressed in this fabric characterisation method but can be evaluated using the additional power benchmarks presented in [4]. From Xilinx devices, we can also observe a general trend of reduction of active power with advancing process technology while inactive power is becoming more challenging. Dedicated low-power modes are needed to address the issue of power consumption during periods of inactivity. Spartan-3AN is the only device in our experiment that features a low-power mode. The inactive power reduction by a factor of 3 represents a good improvement but a power consumption of 23.9 mW in suspend mode is too high for most power budgets in mobile applications.

All our experiments are performed on just one particular device of each type and we therefore have no information on the variability of the results depending on process variation. To make the measurements more representative, the benchmark can be repeated and averaged with several devices of the same type.

6. Conclusions and Future Work

In this paper, we provide a new, application independent methodology for the fabric characterisation of fine-grain FPGAs. This methodology is useful in evaluating the active and inactive power consumption as well as advanced low-power modes. We describe procedures for measuring active and inactive power and temperature on FPGAs using a simple experimental setup. The key to this setup is the use of random number generators as a highly active circuit.

To illustrate our methodology, we perform the fabric characterisation for four Xilinx FPGAs and one Silicon Blue FPGA. Our comparison of Xilinx and Silicon Blue shows that active power is similar for the same technology node but inactive power is significantly improved in the case of Silicon Blue. This is because the Silicon Blue device is specifically

optimised for low leakage trading off device performance. Another optimisation is the reduction of internal switching activity in inactive states. Our measurements also show how advances in process technology reduce the active power by more than a factor of 4. We further observe an increase of inactive power by up to one order of magnitude. However, modern devices generate less heat per activity and suffer less from temperature-based deterioration of inactive power. Additionally, we measure one specific low-power mode in Spartan-3AN that reduces the inactive power by a factor of 3. These improvements are noteworthy, but advances through process technology alone are not enough to meet the strict power constraints in mobile devices. In particular, the power consumption during inactive periods needs to be addressed. To meet mobile power requirements, future devices need flexible and more effective low-power modes as well as techniques addressing the leakage in next-generation process technologies.

Our proposed methodology is part of a power benchmarking framework [3] which also covers further application-specific test cases that are representative of computations in mobile devices. Current and future work includes studying a wide range of devices and characterising hardened IP blocks, and their performance in various low-power applications.

References

[1] I. Kuon and J. Rose, "Measuring the gap between FPGAs and ASICs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 2, pp. 203–215, 2007.

[2] T. Tuan, A. Rahman, S. Das, S. Trimberger, and S. Kao, "A 90-nm low-power FPGA for battery-powered applications," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 2, pp. 296–300, 2007.

[3] "Groundhog benchmark suite," Tech. Rep., Imperial College, London, UK, 2009, http://cc.doc.ic.ac.uk/projects/GROUNDHOG/.

[4] P. Jamieson, T. Becker, W. Luk, P. Y. K. Cheung, T. Rissa, and T. Pitkänen, "Benchmarking reconfigurable architectures in the mobile domain," in *Proceedings of IEEE Symposium on Field-Programmable Custom Computing Machines*, pp. 1–8, Napa, Calif, USA, April 2009.

[5] C. Piguet, *Low-Power CMOS Circuits*, CRC Press, Boca Raton, Fla, USA, 2005.

[6] O. S. Unsal, J. W. Tschanz, K. Bowman, et al., "Impact of parameter variations on circuits and microarchitecture," *IEEE Micro*, vol. 26, no. 6, pp. 30–39, 2006.

[7] V. George, H. Zhang, and J. Rabaey, "The design of a low energy FPGA," in *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED '99)*, pp. 188–193, San Diego, Calif, USA, August 1999.

[8] L. Shang, A. S. Kaviani, and K. Bathala, "Dynamic power consumption in Virtex^{TM}-II FPGA family," in *Proceedings of the 10th ACM International Symposium on Field Programmable Gate Arrays (FPGA '02)*, pp. 157–164, Monterey, Calif, USA, February 2002.

[9] "*Virtex-II Pro and Virtex-II Pro X Platform FPGAs: Complete Data Sheet*," Xilinx Inc., May 2007.

[10] A. Gayasen, K. Lee, V. Narayanan, M. Kandemir, M. J. Irwin, and T. Tuan, "A dual-VDD low power FPGA architecture," in *Proceedings of the 14th International Conference on Field Programmable Logic and Its Application (FPL '04)*, vol. 3203, pp. 145–157, Leuven, Belgium, August-September 2004.

[11] R. Tessier, S. Swaminathan, R. Ramaswamy, D. Goeckel, and W. Burleson, "A reconfigurable, power-efficient adaptive Viterbi decoder," *IEEE Transactions on VLSI Systems*, vol. 13, no. 4, pp. 484–488, 2005.

[12] J. Becker, M. Hübner, and M. Ullmann, "Power estimation and power measurement of Xilinx Virtex FPGAs: trade-offs and limitations," in *Proceedings of the 16th Symposium on Integrated Circuits and Systems Design (SBCCI '03)*, pp. 283–288, Sao Paulo, Brazil, September 2003.

[13] S. Lopez-Buedo, J. Garrido, and E. Boemo, "Thermal testing on reconfigurable computers," *IEEE Design and Test of Computers*, vol. 17, no. 1, pp. 84–91, 2000.

[14] J. Lamoureux and S. J. E. Wilton, "On the trade-off between power and flexibility of FPGA clock networks," *ACM Transactions Reconfigurable Technology and Systems*, vol. 1, no. 3, pp. 1–33, 2008.

[15] "*Power vs. Performance: The 90 nm Inflection Point*," White Paper, Xilinx Inc., 2006.

[16] "*40-nm FPGA Power Management and Advantages*," White Paper, Altera Inc., 2008.

[17] "*Spartan-3 Generation FPGA User Guide v.1.2*," Xilinx Inc., April 2007.

[18] "*MachXO Family Data Sheet*," Lattice, June 2009.

[19] "*Igloo Handbook*," Actel, April 2009.

[20] "*iCE65 Ultra Low-Power Programmable Logic Family Data Sheet*," SiliconBlue, June 2009.

[21] D. B. Thomas and W. Luk, "High quality uniform random number generation using LUT optimised state-transition matrices," *The Journal of VLSI Signal Processing*, vol. 47, no. 1, pp. 77–92, 2007.

[22] "*Using Suspend Mode in Spartan-3 Generation FPGA*," Xilinx Inc., May 2007.

[23] "*Virtex-5 Family Platfrom Overview LX and LXT Platforms v2.2*," Xilinx Inc., January 2007.