Static power model for CMOS and FPGA circuits

Anas Razzaq | Andy Ye

Department of Electrical and Computer Engineering, Ryerson University, 350 Victoria Street, Toronto, Canada

Correspondence
Anas Razzaq, Department of Electrical and Computer Engineering, Ryerson University, 350 Victoria Street, Toronto, Canada.
Email: anas.razzaq@ryerson.ca

Funding Information
Natural Sciences and Engineering Research Council (NSERC)

Abstract
In Ultra-Low-Power (ULP) applications, power consumption is a key parameter for process independent architectural level design decisions. Traditionally, time-consuming Spice simulations are used to measure the static power consumption. Herein, a technology-independent static power estimation model is presented, which can estimate static power with reasonable accuracy in much less time. It is shown that active area only is not a good indicator for static power consumption, hence in this model, the effects of transistor sizing, transistor stacking, gate boosting and voltage change are considered. The procedure to apply this model to processors and FPGAs is demonstrated. Across different process technologies, compared to traditional spice simulation, this model can estimate the static power consumption of processor with an error of 1%–4%, while static power consumption of an FPGA system with an error of 1%–15%.

1 | INTRODUCTION

In the recent years, the demand for Ultra-Low-Power (ULP) applications has grown significantly. In ULP design, power consumption is one of the most important parameters and availability of power dissipation data is an absolute necessity for architectural level design decisions [1]. The performance of different systems can be compared either using simulation tools or using estimation models. In the literature studies, Elmore delay model [2] is available for delay estimation, similarly for area estimation models like VPR area model and COFFE area model [3] are available. Dynamic power can be estimated based on the capacitance of the system, but to the best of our knowledge, there is no reliable process independent static power model for CMOS or pass transistor devices. It is commonly thought that static power can be estimated based on the active area but in our work, we will show that this is not true in all cases. In the FPGA research community, a common practice for static power is to use spice simulation [4–6]. Spice simulations can not only be time-consuming but for each process technology, one has to generate new simulation results. In this work, we will propose a process independent static power estimation model that can reliably predict the static power consumption of heterogeneous systems to assist the process independent architectural level investigation of design trade-offs between fixed logic CPUs and reconfigurable logic FPGAs.

To meet performance requirements, designers have to decide between heterogeneous systems like ASIC, FPGA or CPU. For example, it is a well-known fact that FPGAs have lower latency and are more energy-efficient compared to general-purpose processors [7]. But currently, most power-limited applications such as wireless sensor networks and implantable devices, prefer custom ASICs [8] or low power micro-controllers [9]. In recent years FPGA manufacturers have offered some low power devices like iCE FPGA and Actel Igloo, these low power FPGAs have gained the attention of designers and few published results are using them for ULP applications [10]. Now, in a power-limited scenario, if a designer wants to replace a processor with a custom-designed low-power FPGA fabric, the first question that needs to be answered is that how many FPGA tiles can be turned on in the available power budget? and then decide if that number of tiles is enough for the targeted application.

Similarly, power dissipation also influences on the decision among homogenous devices as well. For example in FPGAs generally, NMOS pass transistor switch is used but, it was shown by [5] that for low voltage and low power applications Transmission Gate (TG)-based FPGAs can achieve 26% better power delay product (PDP) compare to NMOS pass transistor-based FPGAs.

In this work, we investigate procedures to estimate the static power consumption for both processors and FPGAs in a process independent manner. Moreover, for FPGAs we will...
demonstrate the procedure for NMOS-based FPGAs and TG-based FPGAs. To verify the validity of our model we measured the static power consumption of the DLX processor and an FPGA system implementing different benchmark circuits. Then we compare our model with the measured values. For simulations we used Predictive Technology Models (PTM) [11] for 130, 90, 65, 45 and 32 nm. Table 1 tabulates the physical parameters used for each process technology. Default values of physical parameters are used and one can refer to [12] for more explanation. DLX processor was selected because of its simple architecture and easily available open-source implementation.

Herein, Section 2 briefly reviews the accuracy of currently used performance estimation models for FPGAs. Section 3 explains the procedure to measure the static power consumption of DLX process and how our static power model can be used for CMOS circuits. In Section 4 methodology adopted to measure the static power consumption of FPGA Logic Element is explained. In the same section, we will also explain how to apply our static power model on pass transistor circuits. In Section 5, utilization of our static power model is shown with a case study of replacing CPU with an FPGA. In Section 6 we analyse the accuracy of the static power model by implementing different benchmark circuits on an FPGA system. Section 7 concludes the paper.

## 2 Accuracy of Performance Estimation Models for FPGAs

An FPGA device consists of billions of transistors. It is practically not possible to analyse the complete FPGA device using SPICE-based simulation solely, therefore higher level Computer-Aided Design (CAD) tools are required for the performance analysis [13]. CAD tools provided by the vendors can accurately and quickly estimate the performance of an application on an existing FPGA, but they cannot be used to study the novel FPGA architectures [14]. Verilog-to-Routing (VTR) [15] (previously known as Versatile Packing, Placement and Routing [VPR]) is one of the most widely used open-source FPGA CAD tool that allows a variety of options to explore new FPGA architectures. VTR uses a combination of both SPICE simulations and mathematical estimation models to compute area, delay and power of any circuit implemented on the targeted FPGA [16].

In VTR, the minimum width transistor area model is used to estimate the area of the FPGA device. However, it was shown by Ref. [17] that the error percentage of the VPR area model for some components is even greater than 50%. Table 2 shows the error percentage of the VPR area model for different components. Similarly, VPR uses the Elmore delay model to estimate the delay across different components in the FPGA device and the error percentage of Elmore delay is also considerably high ranging from 0.7% to 24%, as shown in Table 3.

VTR provides architects with useful information to evaluate their designs, therefore despite having this high error percentage VTR is still one of the most widely used CAD tools in academia for FPGA research. Accuracy of models is important for the simulations and final stage of the design, but it’s also desirable to have simple mathematical models to facilitate high-level decision between alternate designs. At an early stage of the design, a useful level of abstraction for application is more important than the absolute accuracy of these models [1]. In this work, we are investigating if the static power consumption of FPGAs and CMOS circuits can be estimated using mathematical models with reasonable accuracy.

Both the delay and area estimation models used in VTR are process independent but, to estimate the power VTR requires process technology file. Power consumption of each FPGA subcircuit is measured using SPICE simulation and then VTR estimates the total power of the circuit based on the utilization of each subcircuit [16]. Development of a new FPGA is an iterative process that involves transistor-level modifications (resizing) to tune the area, delay and power [3]. This means that each time the transistor size is changed in a subcircuit, a new SPICE simulation is required. The availability of a mathematical power estimation model can help in speeding up this iteration process, furthermore, it will also facilitate FPGA architects to evaluate their design independent of process technology. Although it’s out of the scope of this work but, integration of power estimation model in VTR can allow the following three modes to the users:

- A quick process independent power estimation.
- Initial measurement of power using accurate SPICE simulations and then utilise a mathematical model during the iterative performance tuning process.
- For more accurate results utilise only SPICE simulations (currently available feature).

### Table 1: Physical Parameters of Transistors (Model Generated using [11])

| Process (nm) | Lref (nm) | VTH (V) | VDD (V) | TOX (nm) Thin | TOX (nm) Thick | RDSW (Ω) |
|-------------|-----------|---------|---------|---------------|---------------|---------|
| 130         | 49        | 0.284   | 1.3     | 1.6           | 3.2           | 200     |
| 90          | 35        | 0.284   | 1.2     | 1.4           | 2.8           | 180     |
| 65          | 24.5      | 0.29    | 1.1     | 1.2           | 2.4           | 165     |
| 45          | 17.5      | 0.295   | 1       | 1             | 2             | 155     |
| 32          | 12.6      | 0.292   | 0.9     | 0.9           | 1.8           | 150     |

### Table 2: Error percentage of VPR area model [17]

| Components     | Area (λ²) | Error (%) |
|----------------|-----------|-----------|
| Inverter       | 1036      | 520       | 50       |
| Buffer (4x)    | 2112      | 1976      | 6        |
| Buffer (16x)   | 5292      | 7176      | 36       |
| 4-LUT          | 11,904    | 6240      | 48       |
| Full adder     | 6072      | 8112      | 34       |
3 | STATIC POWER MEASUREMENT OF DLX PROCESSOR

Many open source VHDL implementation of DLX processor are available, in this work we are using the implementation of [18]. We measured the static power consumption of the DLX processor across different process technologies by adopting the following procedure:

- We synthesised VHDL implementation of [18] using Synopsis Design Vision (we are not disclosing process technology here because of NDA).
- A detailed cell report was generated to find out the type and quantity of standard cells required to implement the DLX processor.
- Then we used PTM to design the cells that were reported by Synopsis. We designed cells in PTM 130, 90, 65, 45 and 32 nm process technologies.
- Using spice simulations static power consumption of individual cells was measured (average static power for all possible input combinations was measured).
- Once the static power consumption and quantity of each cell were known, we found the total static power consumption of the DLX processor by just added the static power consumption of individual cells.

Table 4 shows the static power consumption of DLX processor along with some of the building blocks across different process technologies. It is usually expected that the power consumption of a device reduces as the process technology shrinks but according to [19] static power dissipation per device increases with the process technology shrinking. Here it is also important to mention that in a shorter process technologies devices operate at a much faster rate and as a result the total energy consumption per device reduces with the process shrinking not necessarily being the power consumption [19].

But interestingly Table 4 shows that for different process technologies static power consumption of the DLX processor remains almost the same. To explain why this is happening we need to look at the subthreshold leakage current (cause of static power) equation which is given by:

\[
I_{\text{sub}} = \mu C_{\text{OX}} \frac{W}{L} (m - 1) (V_T)^2 \times e^{-\frac{V_{G}-V_{TH}}{mV_T}} \\
\times \left(1 - e^{-\frac{V_{DD}}{V_{TH}}}ight)
\]

(1)

Where,
- \(\mu\) is the zero bias mobility.
- \(C_{OX}\) is Oxide Capacitance.
- \(W\) is width of the transistor
- \(L\) is Length of the transistor.
- \(m\) is body effect coefficient.
- \(V_T\) is thermal voltage.
- \(V_G\) is Gate voltage.
- \(V_{TH}\) is threshold voltage.
- \(V_{DS}\) is drain to source voltage.

From the Equation-(1) it is clear that:

- \(I_{\text{sub}}\) increases with the reduction of \(V_{TH}\).
- \(I_{\text{sub}}\) increases with the increase in \(C_{OX}\).
- \(I_{\text{sub}}\) decreases with the reduction of Supply voltage \(V_{DD}\) (because \(V_G\) and \(V_{DS}\) will decrease).

With the technology shrinking \(V_{TH}\) is decreasing and \(C_{OX}\) is increasing (because of oxide thickness decreases), which causes an increase in \(I_{\text{sub}}\). On the other hand, with the technology shrinking it is necessary to decrease the supply voltage to suppress the power consumption. Now, two parameters \((V_{TH} \text{ and } C_{OX})\) are causing an increase in \(I_{\text{sub}}\) while the third parameter \(V_{DD}\) is causing a decrease in static power consumption, therefore, we do not observe much change in static power consumption from one process technology to another.

3.1 | Active area model to predict static power

Static power consumption of CMOS circuits is usually measured using spice simulations which is not only a time-consuming process but also requires transistor models for each process technology. In this part of our work, we are proposing

| Components | Wire segment length | Delay (ps) | Error (%) |
|------------|---------------------|------------|-----------|
| Buffered Routing Resources | 1 | 472 | 448 | 5.1 |
| | 4 | 850 | 777 | 8.6 |
| | 8 | 1340 | 1240 | 7.5 |
| | 16 | 2420 | 2270 | 6.2 |
| Pass-transistor Routing Resources | 1 | 471 | 359 | 24 |
| | 4 | 852 | 543 | 14 |
| | 8 | 1340 | 868 | 6.8 |
| | 16 | 2420 | 1460 | 0.7 |

| Process | Voltage (V) | Static power (mw) |
|---------|-------------|-------------------|
| 130 nm  | 1.3 | 0.09 0.16 2.42 |
| 90 nm   | 1.2 | 0.09 0.16 2.40 |
| 65 nm   | 1.1 | 0.08 0.15 2.18 |
| 45 nm   | 1  | 0.07 0.13 1.85 |
| 32 nm   | 0.9 | 0.10 0.18 2.46 |
a static power estimation method which is independent of process technology and can be used by designer for estimation of static power consumption at the early stage of design.

It is generally thought that active area can be a good predictor for static power consumption. But this statement is not true in most cases. Consider the example of transistors connected in series, their power does not increase proportionally to the active area because of stacking effect [20]. Similarly, if we have \( N \) number of transistors connected in parallel, if all the transistors are in OFF state then their static power is proportional to area, but even if one of them is in ON state then total static power consumption will be almost zero (\( V_{DS} \) will be 0). So on average for parallel transistors as well active area is not a good indicator for static power consumption. Just to strengthen this point we will use an active area model to predict static power for different CMOS logic gates and will see how far off it is in predicting static power. We are using the following steps to estimate static power from the active area:

1. Use COFFE [3] area model to estimate active NMOS-Area (\( A_N \)) and PMOS-Area (\( A_P \)) in the circuit.
2. Measure Static power consumption of NMOS (\( P_N \)) and PMOS (\( P_P \)) in inverter configuration (PMOS is sized according to PN ratio). Table 5 tabulates \( P_N, P_P \) and PN ratios for different process technologies we are using in our work.
3. Use following equation to estimate the static power

\[
P_{\text{Static}} = A_N \times P_N + \frac{1}{PN \text{ Ratio}} A_P \times P_P
\]  

(2)

Static power prediction using Equation 2 for different logic gates and DLX processor is shown in Table 6. The model works well for smaller gates (one to two input), but as mentioned earlier this model does not account for stacking effect or parallel circuit effect, so it overestimates the static power for logic gates with three or more inputs. Moreover, the rate of increase of active area with sizing (width) is much slower compared to leakage current (see Equation-1), so static power model based on active area underestimate the effect of sizing. But interestingly since it underestimates for some logic gates and overestimates for others this simple static power model only has 1%–15% error for DLX processor across different process technologies.

### 3.2 Our process independent static power model

This model accounts for the effect of transistors in series or parallel. In this model, we assume transistor as a resistor, the value of resistance is very high in the OFF state, while in the ON state its value is very low (almost 0 \( \Omega \)). We will start explaining our model with the example of an inverter shown in Figure 1. In inverter there can be two possible states, we calculate static power consumption for each state (Figures 1b and 1c) and then take average of all the states (Figure 1a). Applying the same concept we can estimate the static power consumption of any logic gate. For illustration purposes, another example of a 2-input NAND gate is shown in Figure 2. Although at this point we are ignoring lots of factors such as gate leakage, effect of \( V_{DS} \) on transistor’s resistance, etc but ignoring those factors make problem very simple to solve and at the later stage, we will add correction factor. From Figure 2 average static power of the NAND gate will be:

\[
P_{\text{Pull-up}} = \frac{(V_{DD})^2}{2 \times R_{OFF}} = P_P
\]  

(3)

\[
P_{\text{Pull-down}} = \frac{5 \times (V_{DD})^2}{4 \times R_{OFF}} = \frac{5}{2} P_N
\]  

(4)

Most of the basic logic gates are a combination of series and parallel transistors. If one of the pull-up networks or pull-down networks consists of series-connected transistors then the other one will have parallel-connected transistors or vice versa (obviously not true for complex gates like Exor or Exnor and for those gates detailed analysis explained in Figure 2 has to be followed). Therefore, rather than deriving equations for individual logic gates, we will devise generic equations for N transistors connected in series or parallel. Applying the procedure explained in Figures 1 and 2, static power consumption for 2, 3, and 4 NMOS transistors connected in series and parallel is shown in Table 7. Derivation for PMOS-based circuit follows the same procedure. Average static power for PMOS circuits would also have similar equations except \( P_N \) will be replaced with \( P_P \). Based on Table 7, the most closest generalized equation of static power for single transistor as a function of \( N \) can be written as:

\[
P_{\text{Parallel}} = \frac{W}{2^{N-1}} P_{P/N}
\]  

(5)

\[
P_{\text{Series}} = \frac{1.2 \times W}{N} P_{P/N}
\]  

(6)

Here, depending on the configuration \( P_{P/N} \) is the power of either PMOS or NMOS used in the unit inverter. \( W \) in the

| Process | NMOS | PMOS | PN Ratio |
|---------|------|------|----------|
| 130 nm  | 8.81 | 5.52 | Exact    | 2.75    | 3 |
| 90 nm   | 8.85 | 5.69 | Exact    | 2.7     | 3 |
| 65 nm   | 8.55 | 5.13 | Exact    | 2.7     | 3 |
| 45 nm   | 7.71 | 4.66 | Exact    | 2.85    | 3 |
| 32 nm   | 8.92 | 7.21 | Exact    | 2.6     | 3 |
| Devices | 130 nm | | | | 90 nm | | | | 65 nm | | | | 45 nm | | | | 32 nm | | |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Meas. (nw) | Model (nw) | Error (%) | Meas. (nw) | Model (nw) | Error (%) | Meas. (nw) | Model (nw) | Error (%) | Meas. (nw) | Model (nw) | Error (%) | Meas. (nw) | Model (nw) | Error (%) | Meas. (nw) | Model (nw) | Error (%) | Meas. (nw) | Model (nw) | Error (%) |
| AND2 | 64 | 61 | 5 | 65 | 62 | 5 | 60 | 58 | 3 | 52 | 52 | 0 | 72 | 70 | 3 |
| AND3 | 68 | 89 | 31 | 69 | 90 | 30 | 64 | 85 | 33 | 53 | 77 | 45 | 70 | 101 | 44 |
| AND4 | 67 | 121 | 75 | 69 | 123 | 78 | 65 | 116 | 78 | 51 | 105 | 106 | 66 | 136 | 106 |
| EXOR2 | 134 | 137 | 2 | 134 | 139 | 4 | 122 | 130 | 7 | 104 | 118 | 13 | 139 | 159 | 14 |
| NOT | 25 | 19 | 24 | 26 | 19 | 27 | 24 | 18 | 25 | 22 | 16 | 27 | 31 | 21 | 32 |
| NOT(4s) | 101 | 36 | 64 | 104 | 37 | 64 | 96 | 34 | 65 | 87 | 31 | 64 | 122 | 42 | 66 |
| NAND2 | 34 | 42 | 24 | 34 | 43 | 26 | 31 | 40 | 29 | 27 | 37 | 35 | 48 | 37 |
| NAND3 | 35 | 70 | 100 | 34 | 72 | 112 | 32 | 67 | 109 | 25 | 61 | 144 | 35 | 79 | 126 |
| NAND4 | 34 | 103 | 203 | 33 | 104 | 215 | 32 | 98 | 206 | 23 | 89 | 287 | 23 | 114 | 396 |
| OR2 | 36 | 45 | 25 | 36 | 45 | 25 | 32 | 42 | 31 | 27 | 38 | 41 | 36 | 53 | 47 |
| OR3 | 37 | 77 | 108 | 35 | 79 | 126 | 31 | 73 | 135 | 24 | 66 | 175 | 30 | 92 | 207 |
| OR4 | 34 | 116 | 241 | 31 | 118 | 281 | 27 | 109 | 304 | 20 | 99 | 395 | 26 | 140 | 438 |
| OR | 56 | 63 | 13 | 56 | 64 | 14 | 51 | 60 | 18 | 45 | 54 | 20 | 59 | 74 | 25 |
| OR4 | 54 | 96 | 78 | 53 | 98 | 85 | 47 | 90 | 91 | 40 | 82 | 105 | 50 | 114 | 128 |
| DLX Proc. | 2.42 (mw) | 2.43 (mw) | 0.4 | 2.40 (mw) | 2.48 (mw) | 3 | 2.18 (mw) | 2.32 (mw) | 6 | 1.85 (mw) | 2.10 (mw) | 14 | 2.46 (mw) | 2.83 (mw) | 15 |
equation is the ratio between the width of the transistor in the current circuit and the width of the transistor used in a unit inverter. \( W \) is added in the equation on the bases of simulation results and leakage current Equation-(1), both indicate that static power is proportional to the width of the transistor.

To verify the equations we simulated for both series and parallel connected transistors, results are shown in Figure 3. Accuracy of Equation-(5) for PMOS parallel circuit can be seen from Figure 3a, the equation fits exactly to the measured values across all process technologies. For the other three configurations, slight variation across different process technologies was observed. These variations can be associated with the factors that we ignored in the initial stage of our model. We didn’t consider the effect of gate leakage and as gate leakage in NMOS is much more compared to PMOS [21], therefore more variation is observed in NMOS. Simulation results also showed that the actual reduction in static power due to the stacking effect was much more than what we estimated in Equation-(6) [22]. Based on simulation results, we will add a correction factor that results in on average minimum error across all process technologies.

\[
\begin{align*}
PMOS &= \frac{W}{2^{N-1}} P_p \\
NMOS &= \frac{1.2^{N-1} W}{2^{N-1}} P_N
\end{align*}
\]

For series.

\[
\begin{align*}
PMOS &= \frac{1.2}{N^2} W P_p \\
NMOS &= \frac{1.8 \times 1.2}{N^2} W P_N
\end{align*}
\]

Table 8 shows the power estimation of different logic gates and DLX processor using our static power model. Error for larger gates (4-inputs) is still in the range of 2%-31% but significant improvement is achieved compared to the area-based model presented in Table 6.

4 | STATIC POWER MEASUREMENT OF AN FPGA LOGIC ELEMENT

An FPGA architecture consists of an array of identical tiles. Each FPGA tile consists of a Logic Block (LB), connection blocks (CB), and switch block (SB). Figure 4 shows the typical FPGA tile used in VPR. In this work, we are using Look-up-Table (LUT) size (K) of four and cluster size (N) of 4. Because K of four or five is the most area-efficient and beyond the cluster size of four there is no significant improvement in the total area of an FPGA [23]. The approach we are using to measure the static power consumption of the FPGA Logic Element (LE) is very similar to the approach we used for the

![Figure 1](image1.png)

**FIGURE 1** Static power calculation of inverter using Model-2

![Figure 2](image2.png)

**FIGURE 2** Static power calculation of NAND using Model-2
We used two supply voltages in LE by higher voltage, which results in boosted voltage at the gate. Multiplexers is relatively easy because SRAM cells can be powered (of transistors). Gate boosting in the SRAM controlled multiplexers that are connected at the input of BLE.

| Inputs | $N = 2$ | $N = 3$ | $N = 4$ | $N = 2$ | $N = 3$ | $N = 4$ |
|--------|---------|---------|---------|---------|---------|---------|
| 0000   | $2P_N$  | $2P_N$  | $2P_N$  | $4P_N$  | $6P_N$  | $8P_N$  |
| 0001   | $4P_N$  | $3P_N$  | $2.7P_N$| $0$     | $0$     | $0$     |
| 0010   | $4P_N$  | $3P_N$  | $2.7P_N$| $0$     | $0$     | $0$     |
| 0011   | $0$     | $6P_N$  | $4P_N$  | $0$     | $0$     | $0$     |
| 0100   | $NA$    | $3P_N$  | $2.7P_N$| $NA$    | $0$     | $0$     |
| 0101   | $NA$    | $6P_N$  | $4P_N$  | $NA$    | $0$     | $0$     |
| 0110   | $NA$    | $6P_N$  | $4P_N$  | $NA$    | $0$     | $0$     |
| 0111   | $NA$    | $0$     | $8P_N$  | $NA$    | $0$     | $0$     |
| 1000   | $NA$    | $NA$    | $2.7P_N$| $NA$    | $NA$    | $0$     |
| 1001   | $NA$    | $NA$    | $4P_N$  | $NA$    | $NA$    | $0$     |
| 1010   | $NA$    | $NA$    | $4P_N$  | $NA$    | $NA$    | $0$     |
| 1011   | $NA$    | $NA$    | $8P_N$  | $NA$    | $NA$    | $0$     |
| 1100   | $NA$    | $NA$    | $4P_N$  | $NA$    | $NA$    | $0$     |
| 1101   | $NA$    | $NA$    | $8P_N$  | $NA$    | $NA$    | $0$     |
| 1110   | $NA$    | $NA$    | $8P_N$  | $NA$    | $NA$    | $0$     |
| 1111   | $NA$    | $NA$    | $0$     | $NA$    | $NA$    | $0$     |
| Average| $2.5P_N$| $3.62P_N$| $4.3P_N$| $P_N$   | $0.75P_N$| $0.5P_N$|

The Logic Element based on NMOS Pass Transistors (LE-NMOS): As we know that, NMOS pass transistor-based multiplexer is not good in passing ‘1’ and requires gate boosting to overcome high static power consumption of downstream CMOS circuit [13]. As can be seen from Figure 4 that the multiplexers used in routing resources (CB, SB and local routing) are SRAM controlled (SRAM connected to gate of transistors). Gate boosting in the SRAM controlled multiplexers is relatively easy because SRAM cells can be powered by higher voltage, which results in boosted voltage at the gate. We used two supply voltages in LE-NMOS implementation, $V_{SRAM}$ is used to power-up SRAM cells in routing resources, while rest of the circuit is powered by $V_{DD}$. SPICE simulations showed that the best PDP is achieved when $V_{SRAM}$ is approximately $V_{TH}$ higher than $V_{DD}$, hence we are using this value for $V_{SRAM}$. The multiplexer used in a LUT is not SRAM controlled. Therefore, [6, 20] used a modified buffer at the output of the LUT which acts as a level-restorer (LR). CMOS circuit following LR receives a strong ‘1’ and as a result, does not consume very high static power. We are also using LR for LUT in LE-NMOS, circuit diagram of LR is shown in Figure 6.

The Logic Element based on Transmission Gate (LE-TG): Another way of implementing multiplexer is using Transmission Gate (TG). Transmission gate switch can provide full rail to rail swing (equally good for passing '0' or '1') and unless very high performance is required TG-based FPGAs can be operated without gate boosting. Therefore, we are using a single voltage source for LE-TG and at the LUT’s output, a normal buffer is used instead of an LR.

Measurement results for both LE-NMOS and LE-TG are shown in Table 9. Unless otherwise stated throughout this paper LE performance refers to the performance of a complete logic cluster. Refer to Figure 4, in Table 9, delay measurement is from the input of routing multiplexer to the output buffer of Basic Logic Element (BLE), similarly reported static power includes power of four BLEs and sixteen routing multiplexers that are connected at the input of BLE.

4.1 Static power model for FPGA blocks

We will use the model described in Section 3.2 to estimate the static power of FPGA’s blocks. For CMOS circuits in an FPGA such as inverters and buffers, we can directly apply the equations (7-10), in this section we will explain the procedure for pass transistor-based circuits and how to deal with gate boosting effect.

4.1.1 2x1 Multiplexer

Circuit diagram of both NMOS-based and TG-based 2x1 Multiplexer is shown in Figure 7, considering pass transistor as a three-input device Table 10 shows the static power estimation for each state of NMOS pass transistor (N-Pass), NMOS-based 2x1 Multiplexer ($2 \times 1_{NMOS}$) and TG-based 2x1 Multiplexer ($2 \times 1_{TG}$). Average static power for each of them can be written as:

$$P_{N-Pass} = \frac{P_N}{2}$$  \hspace{1cm} (11)

$$P_{2\times1_{NMOS}} = P_N$$  \hspace{1cm} (12)

$$P_{2\times1_{TG}} = P_N + P_P$$  \hspace{1cm} (13)

Simulation results showed that $P_{2\times1_{TG}}$ does not require any correction (as shown in Table 11, $P_{2\times1_{TG}}$ has error of less than 1%), but $P_{2\times1_{NMOS}}$ requires a correction factor. $2 \times 1_{NMOS}$ is consuming on average 0.8 times of $P_N$ across different process.
technologies. The reason for this lower than anticipated static power consumption can be associated with the fact that NMOS-based multiplexer produces weak ‘1’ and maximum value of $V_{DS}$ for OFF transistor will be less than $V_{DD}$, resulting in less average static power consumption. In our model, for $P_{2\times1_{NMOS}}$ we will use following equation:

$$P_{2\times1_{NMOS}} = 0.8 \times P_N \quad (14)$$

4.1.2 | Multi-Tree Multiplexer

The LUT of an FPGA is implemented using encoded Multiplexer while the routing network is implemented using 2-level Multiplexer [3, 5, 17]. It can be observed from the Figure 8 that, to implement N input encoded multiplexer, we need N−1, 2x1 multiplexers, while to implement N input 2-level multiplexer, we need $\frac{N}{2} + 1$, 2x1 multiplexers. According to [24], if larger multiplexer is built by the combination of the smaller multiplexer, then the power of a larger multiplexer can be found by adding up the power of the smaller multiplexers. Using the same concept, the static power consumption of an encoded multiplexer ($P_{MUX_{Enc}}$) and a 2-level multiplexer ($P_{MUX_{2Lev}}$) can be found using the following equations:

$$P_{MUX_{Enc}} = (N - 1) \times P_{2\times1_{NMOS}} \quad (15)$$
$$P_{MUX_{2Lev}} = \left( \frac{N}{2} + 1 \right) \times P_{2\times1_{NMOS}} \quad (16)$$

4.1.3 | Effect of sizing

Multiplexers in an FPGA needs to be sized according to the load they are driving and it’s important for the static power model to account for the transistor sizing. Figure 9 shows the effect of transistor sizing on the static power consumption of a multiplexer, the graph shows that the static power increases linearly with the increase in size and this trend is consistent across all the process technologies. Therefore, we can write the generalized equation for the static power consumption of a multiplexer as:

$$P_{Max-W} = P_{MUX} \times W \quad (17)$$
| Devices | 130 nm | 90 nm | 65 nm | 45 nm | 32 nm |
|---------|--------|--------|--------|--------|--------|
|         | Meas. (nw) | Model (nw) | Error (%) | Meas. (nw) | Model (nw) | Error (%) | Meas. (nw) | Model (nw) | Error (%) | Meas. (nw) | Model (nw) | Error (%) | Meas. (nw) | Model (nw) | Error (%) |
| AND2    | 64     | 61     | 5       | 65     | 62     | 5       | 60     | 58     | 3       | 52     | 52     | 0       | 72     | 71     | 1       |
| AND3    | 68     | 57     | 16      | 69     | 58     | 16      | 64     | 54     | 16      | 53     | 49     | 8       | 70     | 66     | 6       |
| AND4    | 69     | 53     | 23      | 69     | 54     | 22      | 65     | 50     | 23      | 51     | 45     | 12      | 66     | 61     | 8       |
| EXOR2   | 134    | 119    | 11      | 134    | 121    | 10      | 122    | 113    | 7       | 104    | 102    | 2       | 139    | 140    | 1       |
| NOT     | 25     | 25     | 0       | 26     | 26     | 0       | 24     | 24     | 0       | 22     | 22     | 0       | 31     | 31     | 0       |
| NOT(4s) | 101    | 101    | 0       | 104    | 104    | 0       | 96     | 96     | 0       | 87     | 87     | 0       | 122    | 122    | 0       |
| NAND2   | 34     | 36     | 6       | 34     | 36     | 6       | 31     | 34     | 10      | 27     | 31     | 15      | 35     | 41     | 17      |
| NAND3   | 35     | 31     | 11      | 34     | 32     | 6       | 32     | 30     | 6       | 25     | 27     | 8       | 35     | 35     | 0       |
| NAND4   | 34     | 27     | 21      | 33     | 28     | 15      | 32     | 26     | 19      | 23     | 24     | 4       | 23     | 30     | 30      |
| NOR2    | 36     | 30     | 17      | 36     | 31     | 14      | 32     | 29     | 9       | 27     | 26     | 4       | 36     | 37     | 3       |
| NOR3    | 37     | 29     | 22      | 35     | 30     | 14      | 31     | 28     | 10      | 24     | 25     | 4       | 30     | 36     | 20      |
| NOR4    | 34     | 27     | 21      | 31     | 28     | 10      | 27     | 26     | 4       | 20     | 23     | 15      | 26     | 34     | 31      |
| OR2     | 56     | 56     | 0       | 56     | 57     | 2       | 51     | 53     | 4       | 45     | 48     | 7       | 59     | 51     | 14      |
| OR3     | 54     | 55     | 2       | 53     | 56     | 6       | 47     | 52     | 11      | 40     | 47     | 18      | 50     | 47     | 6       |
| OR4     | 49     | 53     | 8       | 47     | 54     | 15      | 42     | 50     | 19      | 34     | 42     | 24      | 41     | 42     | 2       |
| DLX Proc | 2.42 (mw) | 2.32 (mw) | 4       | 2.40 (mw) | 2.32 (mw) | 3       | 2.18 (mw) | 2.12 (mw) | 3       | 1.85 (mw) | 1.86 (mw) | 1       | 2.46 (mw) | 2.51 (mw) | 2       |
4.1.4 | SRAM cell

Figure 10 shows that an SRAM cell consists of two back to back connected inverters and two pass transistors. Therefore, its total static power can be estimated as:

\[ P_{SRAM} = 2 \times P_{N-\text{pass}} + 2 \times P_N + \frac{2 \times P_P}{PNRatio} \]  (18)

It's important to mention here that, to save the static power consumption of SRAM cells, transistors with the thick oxide are used in the SRAM cells [4], \(P_N\) and \(P_P\) in Equation-18 refer to thick oxide transistors. Moreover, PMOS transistors used in an SRAM cell are of unit size and we need to divide \(P_P\) with the PN ratio of process technology.

To avoid loading effect in SRAM cells sometimes an additional inverter is added at the output. In our design, we are only using such a setup for SRAM in LUT. Its static power can be calculated by:

\[ P_{SRAM-LUT} = P_{SRAM} + P_{INV} \]  (19)

Where, \(P_{INV}\) is Power of Inverter.

4.1.5 | Gate boosting effect

As mentioned earlier, the SRAM cells used in routing resources are powered with higher voltage and we need to account for the effect of this voltage boosting in our model. Figure 11 shows the effect of supply voltage on the power consumption of an SRAM cell, it can be observed that with every increase of 0.1 voltage power consumption doubles. This trend is very consistent across all process technologies under test. Therefore,

\[ P_{SRAM-Boost} = P_{SRAM}2^{10V_B} \]  (20)

\((V_B)\) refers to the boosted voltage and normally it's equal to \(V_{TH}\).
Boosting of Multiplexer is much more tricky because only the gate voltage is changing and $V_{DS}$ remains the same. Referring back to subthreshold leakage current Equation-1, leakage current increases exponentially by increasing gate voltage $V_G$. Moreover, in boosting situation gate voltage is higher than the normal rated voltage, this higher gate voltage will also increase gate leakage current exponentially [21] and as result overall rate of increase of leakage current would be more than just $e^{\Delta V_G} (V_B = \Delta V_G)$. Considering these two factors, we use the following equation for the boosted multiplexer

$$P_{\text{Mux-Boost}} = P_{\text{MUX}} \times W \times e^{3 \times V_B} \quad (21)$$

Figure 12 shows the graphical comparison between measured and estimated static power of boosted multiplexer. Except for 32 nm process technology, Equation 21 estimates static power of boosted multiplexer with maximum error of 26%. But for 32 nm process technology error is high at 68%.

Table 11 shows the comparison between measured and estimated static power of different components used in the LE, while the graphical comparison between measured and estimated power of both LEs (NMOS and TG) across different process technologies is shown in Figure 13. Our static power model estimates power very accurately for non-boosted components and as LE-TG has no gate boosting therefore, the error percentage for it is less than 1%. But, since for boosted components such as 2x1 multiplexer error percentage is up to 68% and a major part of LE-NMOS consists of the boosted circuit, therefore the estimated static power of LE-NMOS has relatively high error percentage ranging from 1%-14%.

### 5 | REPLACING DLX PROCESSOR WITH FPGA

The power budget plays a big role in deciding the logic capacity of an FPGA device. For example, over 5.5 million logic elements are available in Altera's Stratix 10 FPGA [25] while Altera's low power version MAX 10 FPGA contains only 2000

---

**Table 9** Performance comparison between LE-NMOS and LE-TG

| Process (nm) | VDD (V) | LE-NMOS Power (μW) | Delay (ps) | PDP (fJ) | LE-TG Power (μW) | Delay (ps) | PDP (fJ) |
|--------------|---------|--------------------|------------|----------|----------------|------------|----------|
| 130          | 1.30    | 19.19              | 424        | 8.13     | 15             | 381        | 5.68     |
| 90           | 1.20    | 14.33              | 345        | 4.94     | 15             | 296        | 4.51     |
| 65           | 1.10    | 13.87              | 306        | 4.25     | 14             | 252        | 3.58     |
| 45           | 1.00    | 10.59              | 271        | 2.87     | 13             | 235        | 2.95     |
| 32           | 0.90    | 12.31              | 329        | 4.05     | 17             | 241        | 4.14     |

---

**Figure 7** Circuit diagram pass transistor circuits
to 5000 logic elements [26]. Before implementing an application on an FPGA device, the designer needs to know if the required logic capacity for the application can be achieved in the available power budget or not. Consider a case where a designer wants to replace a DLX processor with a reconfigurable logic FPGA with a constraint that the FPGA fabric should operate in the power budget of a DLX processor. The designer wants to find the maximum logic capacity that can be achieved in the given power budget. To validate and show the utilization of our static power, we will answer this question using both measured values and estimated values.

In this case study, we are using a simple FPGA architecture consisting of LE-NMOS presented in Section-4, routing channel width (W) of 20, directional wires in routing channels, logic cluster input pin connectivity $F_{\text{cnt}} = 0.5W$ and switch block flexibility $F_S = 3$. Furthermore, it is assumed that at the rated voltage, for both the processor and the FPGA static power consumption consists of 50% of the total power consumption. Table 12 shows number of FPGA tiles that can be turned ON with the power budget of a DLX processor. The required number of tiles predicted using our static power model is very close to measured values, maximum error percentage across all process technologies is 9%. Based on these results we can say that our static power model can be used in the early stage of design for process independent architectural level decisions.
6 | STATIC POWER MEASUREMENT OF BENCHMARK CIRCUITS IMPLEMENTED ON AN FPGA

We are utilising VTR CAD tool to measure the static power consumption of a whole FPGA system. Figure 14 shows the flow diagram of VTR to measure the power consumption an FPGA system. We will briefly discuss some of the major steps, for the detailed explanation of each step refer to [15, 16, 27].

1) Verilog Benchmark Circuit: User needs to provide a Verilog code of the circuit he wants to implement on the
FPGA. In the VTR package, some benchmark circuits that come from a variety of real applications are already included. The 12 benchmark circuits we are using in this work are listed in the Table 13.

2) Architecture Description File: Templates of different fully designed architecture description files are also included in the VTR package. These architectures could either be modified to test new architectures or can be used unmodified. We are using the same architecture described in Section 5, which consists of four BLEs per cluster, 10 inputs per cluster, and four inputs for each LUT.

3) SPICE CMOS Technology File: This file describes the properties of NMOS and PMOS transistors. We are using the same PTM files for 130, 90, 65, 45 and 32 nm described in Section 1.

In the first step of VTR CAD flow, hardware description code is synthesised and mapped into logic primitives (LUTs, Flip Flops, etc.) defined by the architecture description file. SPICE simulations (external software required) are performed to measure the power consumption of each subcircuit (primitive). Then based on the utilization of each subcircuit and net actively VTR calculates the total power of the targeted circuit.

Our static power model initially estimates the static power consumption of each subcircuit following the procedure explained in Section 4 then uses the subcircuit count and utilization reported by the VTR to estimate the static power of the whole FPGA system. It is important to mention here that we are verifying our model using a simple FPGA architecture but since the basic building blocks of all the FPGA devices are the same, therefore our model can generically be applied to other FPGA architectures as well. Multiplexers, buffers and SRAM cells are the three main components used in different sizes to design an FPGA device. In Section 4 it was shown that for these basic blocks (multiplexers, buffers and SRAM cells) our static power model can estimate power with reasonable accuracy. Some FPGA devices also have additional functional blocks such as multipliers, adders, etc. These functional blocks are mostly designed using CMOS logic and Section 3 showed the validity of the proposed static power model for CMOS logic circuits.

Table 13 shows the comparison between the measured and estimated static power of 12 benchmark circuits implemented on the FPGA. The results show that across all the process technologies under test our static power model shows reasonable accuracy with the error ranging from 1%–15%.

7 | CONCLUSION

In this study, we presented process independent static power estimation model. We showed that circuit configuration (transistors in parallel/series) plays a big role in the static power consumption and merely using the active area as the static power estimation will result in a high percentage of error, for example for 4-input NOR gate the error was almost 241%–438%. For that reason, our static power model considers circuit configuration and estimates power much more accurately. For the basic CMOS logic gates, our model predicts static power consumption of individual gates with an accuracy of 70% or more. For pass

![Figure 13](image1.png)

**Figure 13** Comparison between measured and estimated static power of LE

| Devices                     | 130 nm | 90 nm | 65 nm | 45 nm | 32 nm |
|-----------------------------|--------|-------|-------|-------|-------|
| **FPGA tile static power (uw)** | Meas. | Model (%) | Meas. | Model (%) | Meas. | Model (%) | Meas. | Model (%) | Meas. | Model (%) |
| 57                          | 52     | 9     | 43    | 42    | 41    | 40     | 32    | 34     | 38    | 42     | 11    |
| **DLX static power (uw)**   | 2422   | 2319  | 4     | 2399  | 2316  | 3     | 2179  | 2117   | 3     | 1853   | 1     | 2458   | 2510  | 2     |
| **No. of tiles**            | 42     | 45    | 7     | 56    | 55    | 2     | 53    | 52     | 2     | 58     | 5     | 65     | 58    | 9     |
| Benchmark Circuit | 130 nm | 90 nm | 65 nm | 45 nm | 32 nm |
|-------------------|--------|-------|-------|-------|-------|
|                   | Meas. (mw) | Model (mw) | Error (%) | Meas. (mw) | Model (mw) | Error (%) | Meas. (mw) | Model (mw) | Error (%) | Meas. (mw) | Model (mw) | Error (%) | Meas. (mw) | Model (mw) | Error (%) |
| and_latch         | 0.0052 | 0.0048 | 9 | 0.0039 | 0.0038 | 1 | 0.0038 | 0.0037 | 1 | 0.0029 | 0.0031 | 8 | 0.0033 | 0.0038 | 15 |
| Bgm               | 2137   | 1945  | 9 | 1654   | 1626  | 2 | 1589   | 1562  | 2 | 1264   | 1324  | 5 | 1547   | 1680  | 9 |
| blob_merge        | 370    | 337   | 9 | 286    | 262   | 2 | 275    | 270   | 2 | 219    | 229   | 5 | 268    | 291   | 9 |
| Boundtop          | 2.058  | 1.856 | 10 | 1.508  | 1.488 | 1 | 1.462  | 1.446 | 1 | 1.114  | 1.198 | 8 | 1.289  | 1.471 | 14 |
| Ch_intrinsics     | 1.674  | 1.510 | 10 | 1.227  | 1.210 | 1 | 1.189  | 1.176 | 1 | 0.906  | 0.974 | 8 | 1.049  | 1.196 | 14 |
| diffeq1           | 134    | 122   | 9 | 103    | 101   | 2 | 99     | 97    | 2 | 78     | 82    | 5 | 95     | 104   | 9 |
| diffeq2           | 1.870  | 1.686 | 10 | 1.371  | 1.352 | 1 | 1.329  | 1.314 | 1 | 1.012  | 1.088 | 8 | 1.171  | 1.336 | 14 |
| LU8PEEng          | 12,184 | 11,160 | 8 | 9672   | 9507  | 2 | 9263   | 9090  | 2 | 7488   | 7770  | 4 | 9348   | 9984  | 7 |
| mkpktMerge        | 1.421  | 1.281 | 10 | 1.041  | 1.027 | 1 | 1.010  | 0.998 | 1 | 0.769  | 0.827 | 8 | 0.890  | 1.015 | 14 |
| mkMADapter        | 31     | 28    | 9 | 23     | 23    | 2 | 23     | 22    | 2 | 18     | 19    | 6 | 21     | 23    | 11 |
| orl200            | 403    | 367   | 9 | 312    | 307   | 2 | 300    | 295   | 2 | 239    | 250   | 5 | 293    | 318   | 8 |
| Sha               | 91     | 83    | 9 | 70     | 69    | 2 | 67     | 66    | 2 | 53     | 56    | 5 | 64     | 70    | 9 |
transistor-based circuits, our model considers the effect of gate boosting and can estimate the static power consumption of FPGA systems with an accuracy of more than 85%.

**ACKNOWLEDGMENT**

This research was supported by a Discovery Grant from the Natural Sciences and Engineering Research Council (NSERC) of Canada.

**ORCID**

Anas Razzaq [https://orcid.org/0000-0002-7708-9299](https://orcid.org/0000-0002-7708-9299)

**REFERENCES**

1. Butts, J.A., Sohi, G.S.: A Static power model for architects. In: Proceedings 33rd Annual IEEE/ACM International Symposium on Microarchitecture. MICRO-33 2000, pp. 191–201. Monterey, CA (2000)
2. Elmore, W.C.: The transient response of damped linear networks with particular regard to wideband amplifiers. J. Appl. Phys. 19(1), 55–65 (1948)
3. Chiasson, C., Betz, V.: COFFE: fully-automated transistor sizing for FPGAs. In: International Conference on Field-Programmable Technology (FPT), pp. 34–41 (2013)
4. Li, F., et al.: Low-power FPGA using pre-defined dual-Vdd/dual-Vt fabrics. In: ACM/SIGDA International Symposium on Field Programmable Gate Arrays - FPGA, pp. 42–50 (2004)
5. Chiasson, C., Vaughn, B.: Should FPGAs ABANDON the pass-gate? In: Field-Programmable Logic and Applications, pp. 1–8 (2013)
6. Kim, J.H., Anderson, J.H.: Synthesizable standard cell FPGA fabrics targetable by the verilog-to-routing CAD flow. ACM Trans. Reconfigurable Technol. Syst. (TRETS). 10(2), 1–23 (2017)
7. Grossmann, P., Leeser, M.E., Onabajo, M.: Minimum energy operation for clustered island-style FPGAs. In: ACM/SIGDA International Symposium on Field Programmable Gate Arrays - FPGA, pp. 157–166 (2013)
8. Liu, X., et al.: An ultra-low power ECG acquisition and monitoring ASIC system for WBAN applications. IEEE J. Emerg. Select. Top. Circ. Syst. 2(1), 60–70 (2012)
9. Iyer, R.: Visual IoT: architectural challenges and opportunities. IEEE Micro. 36(6), 45–47 (2016)
10. Consul.Pacareu, S. et al.: High performance DT-CNN camera device design on ACTEL IGLOO low power FPGA. In: 2011 20th European Conference on Circuit Theory and Design, ECCTD 2011, pp. 37–40 (2011)
11. Predictive Technology Model. (2019) http://ptm.asu.edu/
12. Cao, Y.: Predictive technology model of conventional CMOS devices. In: Predictive Technology Model for Robust Nanoelectronic Design, pp. 7–23. Springer US (2011)
13. Betz, V., Rose, J., Marquardt, A.: Architecture and CAD for Deep-Submicron FPGAS. Kluwer Academic Publishers United States (1999)
14. Goeders, J.B., Wilton, S.J.E.: VersaPower: power estimation for diverse FPGA architectures. In: International Conference on Field-Programmable Technology (FPT), pp. 229–234 (2012)
15. Murray, K.E., et al.: Vtr 8: high performance CAD and customisable FPGA architecture modelling. ACM Trans. Reconfigurable Technol. Syst. (TRETS) (2020)
16. Luu, J., et al.: Vtr 7.0: next generation architecture and CAD system for FPGAs. ACM Trans. Reconfigurable Technol. Syst. (TRETS). 7(2) (2014)
17. Khan, F.F., Ye, A.: An evaluation on the accuracy of the minimum width transistor area models in ranking the layout area of FPGA architectures. ACM Trans. Reconfigurable Technol. Syst. (TRETS), 11(1), 1–23 (2018)
18. Chiatante, E., Calvino, A.T.: DLX microprocessor: design & development. Master’s Dissertation. Politecnico di Torino (2018)
19. Roy, K., Mahmoodi.Meimand, H., Mukhopadhyay, S.: Leakage control for deep-submicron circuits. VLSI Circ. Syst. 5117, 135 (2003)
20. Rahman, A., Polavarapu, V.: Evaluation of low-leakage design techniques for field programmable gate arrays. In: ACM/SIGDA International Symposium on Field Programmable gate arrays - FPGA, pp. 23–30 (2004)
21. Lee, D., Blaauw, D., Sylvester, D.: Gate oxide leakage current analysis and reduction for VLSI circuits. IEEE Trans. Very Large Scale Integr. Syst. 12(2), 155–166 (2004)
22. Weste, N.E.H., Harris, D.M.: Chapter 5: power. In: CMOS VLSI design: a circuits and systems Perspective, 4th ed. Addison-Wesley (2010)
23. Ahmed, E., Rose, J.: The effect of logic block granularity on deep-submicron FPGA performance and density. IEEE Trans. Very Large Scale Integr. 12(3), 288–298 (2004)
24. Anderson, J.H., Najm, E.N.: Active leakage power optimization for FPGAs. IEEE Trans. Comput. Aided Des. Integr. Circ. Syst. 25(3), 423–437 (2006)
25. Intel® Stratis® 10 FPGAs (2016). https://www.intel.com/content/www/us/en/products/programmable/fpga/stratis-10.html
26. Altera Max 10 FPGA (2016). https://www.intel.com/content/www/us/en/products/programmable/fpga/max10.html
27. Goeders, J.: Power Estimation for VTR. (2013). https://vtr.readthedocs.io/en/latest/vtr/power_estimation

How to cite this article: Razzaq A, Ye A. Static power model for CMOS and FPGA circuits. *IET Comput. Digit. Tech.* 2021;15:263–278. [https://doi.org/10.1049/cdt.2.12021](https://doi.org/10.1049/cdt.2.12021)