DESIGNING FOR LOW POWER

A Lattice Semiconductor White Paper

March 2010

Lattice Semiconductor
5555 Northeast Moore Ct.
Hillsboro, Oregon 97124 USA
Telephone: (503) 268-8000
www.latticesemi.com

Introduction

Power consumption is becoming an increasingly important variable when it comes to calculating OPEX and carbon footprint for telecom infrastructure projects. For example, on average, each fully loaded 3G cell site costs approximately $1600/yr US or $3200/yr in Europe. This suggests a typical European operator running 20000 cell sites would consume 58MW, which translates to about $62M USD per year. In addition to these costs, the level of consumption per cell site leads to an estimated 11 tons of carbon dioxide emissions per cell site per year. For these operators, power equals cost. FPGAs are becoming one of the most important facets of basestation architectures, and so the spotlight has fallen on them to minimize power consumption.

For example, to minimize power consumption the LatticeECP3 FPGA family uses variable channel lengths, optimized low-power transistors, and improved routing defaults and algorithms. As a result, the ECP3's static power consumption was reduced by 80% and total power consumption by over 50% for typical designs, compared to competitive SERDES-capable FPGAs.

A system designer considering the use of FPGAs looks at four power components:

Pre-Programmed Static (Quiescent) Device Power Consumption

Pre-Programmed Static Quiescent Device Power Consumption is the amount of power consumed by the FPGA prior to the device being programmed. For quiescent device power consumption the FPGA is in a non-programmed state, yet has been powered. It is important that the device not consume significant power during this time, as conceptually the FPGA device could draw excessive power and potentially shut down power supplies, preventing the board from successfully initializing itself and the system. The FPGA supplier therefore has to carefully design transistors that have a low static

power footprint, without compromising in areas where higher performance is required (e.g. I/O and SERDES)

Inrush Programming Current: (the surge current/power required when programming a device until programming in complete)

Inrush Programming Current has in the past been an issue for FPGA vendors. Inrush programming current had actually been larger than a typical application's power consumption, effectively sizing the power supplies/regulators. This was of course not desirable, and Lattice put considerable energy into designing its products so that programming current (inrush) was well under any typical application's power consumption. Lattice specifies and tracks inrush in its datasheets and Power Calculator tool.

Post-Programmed Static Power Consumption: Power being consumed by the device with a 'zero MHz' frequency

Post-Programmed Static Power Consumption is a very significant component of FPGA power consumption. This is due to the large number of transistors on FPGAs (typically 8 to 10 times for an equivalent ASIC logic implementation, config and muxing not included), all of which have a small amount of leakage current. The leakage on these transistors (pass muxes for switches, RAM cells, etc.) typically is "always on" and drawing power, whether the transistor is used or not. Typically, post-programmed static power is equal to or greater than Quiescent Device Power. There are some recent innovations addressing power grids and removing power to these specific transistors, which will reduce this static power component.

Dynamic Power Consumption: Incremental power consumed by a non-zero frequency component. (i.e. $P=kcV^2f$)

Dynamic Power Consumption tracks kcV^2f, and typically is under the designer's control. Depending on the type of design being implemented (always on, always processing, datapath-type of design vs. a wake up, process and go back to sleep type of design, etc.), *either dynamic power or post-programmed static power is the most significant component in power analysis.*

Power consumption is closely linked to temperature. As the FPGA heats up, power increases due to increased leakage of the transistors. In extreme cases a device can get so hot that the transistors can not turn off, a situation referred to as thermal runaway. It is essential that power analysis be included as part of the design process for any FPGA. Using power analysis, one can have extremely high confidence the design will work within the design environment. Temperature can be controlled using various techniques such as fans, heat sinks, modification of the design, I/O standards, etc.

Modeling "Environment Aware" Power Consumption

In addition to FPGA architectural improvements, the advent of software-based tools has proven very valuable in the low power design process. Lattice's Power Calculator (Figure 1), for example, includes an environment-aware power model, graphical power displays and a variety of useful reports. Thermal resistance options model real world thermal conditions, including heat sinks, airflow and the printed circuit board's complexity, while graphical power curves illustrate operating temperature profiles. Software-based power calculators can be used in the pre- and post-FPGA design process to analyze the power expected when the PCB returns from assembly.

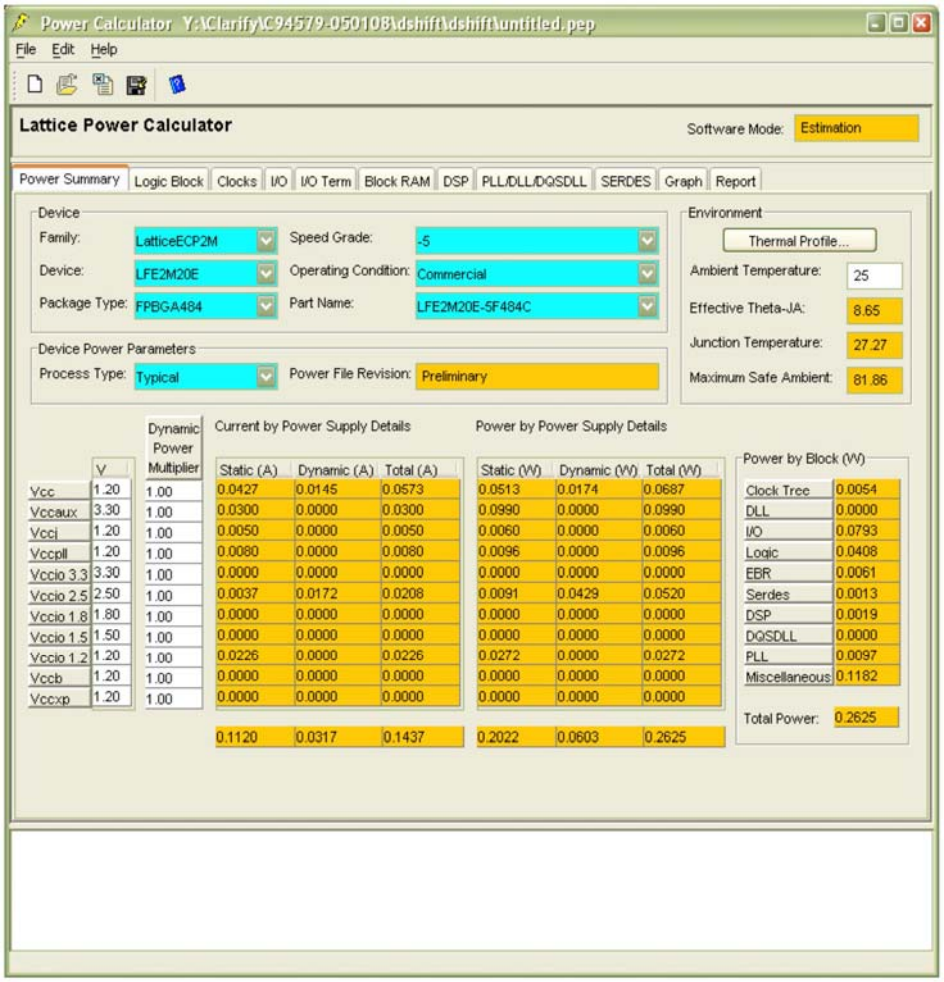

Figure 1 - The Power Calculator can be used at any point in the design cycle. Early estimates of toggle rates can later be replaced with simulation results. The Power Calculator accurately calculates current and power for all architectural elements and provides a thermal profile to model real system conditions

Summary

Does successful low power design rely on Architecture, or Process? The answer is not what traditional conventional wisdom has held to be true. Merely going to an advanced process node does not guarantee lower power: the tradeoffs made during the architecture and circuit design phase are crucial to the end result. If an FPGA is optimized for performance, it will inevitably lead to higher overall power consumption. A tradeoff has to be made in placing high performance circuitry where it is absolutely

needed, while optimizing other areas of the chip for lower power. Traditionally FPGA vendors relied solely on going to the next process node to get quantifiably smaller power footprints. However, the LatticeECP3 has shown that it is possible to have significantly lower power at 65nm than competitive devices at 40/45nm. At the heart of this paradigm shift lays ingenious circuit and transistor design.

###