

Low Power Design for DSP Architecture

Vrinda Mehta, Bansari Mehta, Karthik Nadar, Sanskriti Pahinkar

Abstract — The use of high performance multi-core DSPs in telecommunications access, edge and infrastructure equipment is increasing exponentially. Many of these applications may be remote and the machines cannot be accessed by humans regularly, which emphasize the need for low power DSPs. In this paper, low power design considerations are described using synchronous and asynchronous architectures.

Index Terms — Asynchronous design, DSP architecture, Dynamic, Low power design, Power optimization, Static, Synchronous design

1 INTRODUCTION

“Power demands are increasing rapidly, yet battery capacity cannot keep up.” [in Ditzel et al.: Power-Aware Architecting for data-dominated applications, 2007, Springer][1]

A decade ago, with the advent of smaller geometry process nodes, migrating to them was an exciting prospect with faster clock speeds, double the number of chips per wafer and of course much lower power consumption.

However as the geometries decreased, it became difficult to achieve improvements in the product fundamentals of cost, performance and power without compromising on any one of them.

Transistor leakage current became a major contributor of power loss below the 90nm technology. As transistor size decreased, their threshold voltage also decreased which helped in reducing chip area with increased performance, but with the high cost of increased leakage current. The difficult problem, presented to the designers was to choose between increased clock frequency for faster processing but at higher power consumption or to reduce the power loss but at a slower processing rate.

Nowadays, the demand for smaller, thinner, faster, lighter systems with smaller batteries and enhanced battery life is great but improving the battery life is a huge challenge for researchers. Hence to optimize the usage we first need to identify the major sources of power consumption and then understand how to reduce them.[2]

In this paper, we have discussed various techniques that can be used to assuage or circumvent the power crisis by using two architectural designs – synchronous and asynchronous.

- Vrinda Mehta is currently pursuing bachelors degree program in electronic engineering in D.J.Sanghvi College of Engineering, India.
E-mail: vrindamht@yahoo.co.in
- Bansari Mehta is currently pursuing bachelors degree program in electronic engineering in D.J.Sanghvi College of Engineering, India.
E-mail: bansarimht@yahoo.co.in
- Karthik Nadar is currently pursuing bachelors degree program in electronic engineering in D.J.Sanghvi College of Engineering, India.
E-mail: karthik.nadar@gmail.com
- Sanskriti Pahinkar is currently pursuing bachelors degree program in electronic engineering in D.J.Sanghvi College of Engineering, India.
E-mail: sanskriti.pahinkar@gmail.com

2 POWER LOSS OPTIMIZATION

Power losses are primarily of two kinds: dynamic and static. The source of static power dissipation is leakage currents in the transistor while switching operations prove to be a major factor in dynamic losses.

2.1 Dynamic Power Loss Optimization

A major contributor of dynamic power loss is switching operations. Also the type of system used to design the processor-synchronous or asynchronous, plays an important role in decreasing the power consumed.

2.1.1 Synchronous Systems

Synchronous architecture is most commonly used in processors, which implies that all these processors are controlled by a common clock. This causes quite a few dynamic complications as the technology becomes faster and smaller. The major power culprits that need to be optimized are-clock tree losses and logic transition losses.[3]

2.1.1.1 Clock Tree Losses and Optimization

The modern processors utilize massive clock trees and operate at high speeds. Therefore power loss occurs due to the high speed as well as the large power consuming drivers that are necessary to minimize clock propagation delays through the processor and hence the skew.

This power consumption can be minimized using the following techniques:

- Using flip-flops each having an independent clock to restrict operation to the time when they are required.
- We use gated clock trees to cut-off clock to the modules which are not being utilized. In fig. 1, we can see that each module has an enable pin which can be deactivated when the module is not in use thereby reducing power loss.

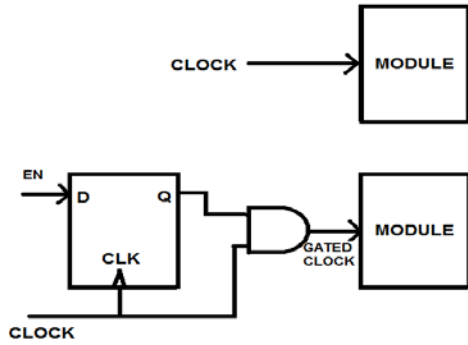


Fig. 1: Gated Clock Tree

- We implement additional combinational circuitry to perform parallel operations instead of using a single module working at a faster clock speed. In fig. 2, we observe that the Multiply and Accumulate (MAC) unit requires to be clocked every cycle so the clock speed needed is greater. Also the multiplier utilizes larger amount of power as it processes the huge amount of data provided to it at once.

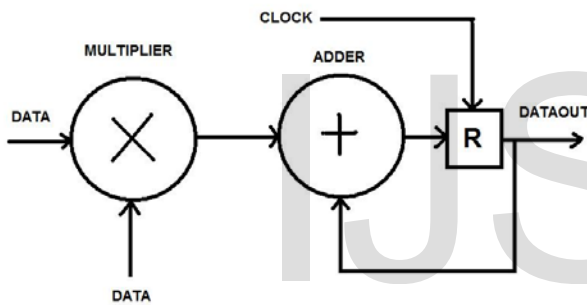


Fig. 2: Single Stage Sequential feedback MAC

- The fig. 3 consists of a multi-stage multi-cycle (4-stage) MAC unit which processes data in smaller quantities in the different stages and hence it requires low-power multipliers and adders. Also since there are four stages, the accumulator needs to be clocked once every four cycles which further reduces the power used. Although one may say that the multi-stage MAC may use more space in the die, this is not true as the single MAC is much larger as it processes more data at once. The simpler circuitry and slower gates of the multi-stage MAC also help in reducing its size.

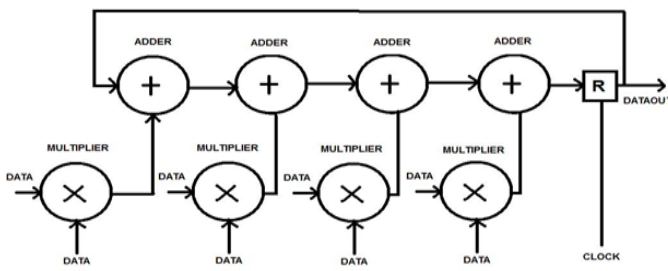


Fig. 3: Multi-stage, multi-cycle MAC

2.1.1.2 Logic Transition Losses and optimization

Power is consumed during charging and discharging every time the logic circuit transitions from one state to the other. The techniques that can be utilized to minimize the power consumption are as follows:

- Optimize gate placement:**
 Today's back-end tools allow the user to write the software code on a language such as VHDL and then let the tools create the physical design with regard to that code. While this increases the speed at which the functional silicon is realized, the placement of the gates and interconnecting wires is by no means optimal. The wires may be placed too close together or may be too long which increases their capacitance which eventually leads to increased power loss. The designers are removed from the designing process in these back-end tools whereas if these tools can be improved to allow human interface and visualization then the placement will become optimal as the human brain can easily figure out where the wire lengths can be shortened to improve power wastage. Moreover, since the wires are shorter, the size of the gates connected to these wires also reduces, thus decreasing the entire circuit size.
- Optimize signal routing:**
 Data and address buses in processors are placed close together and change states frequently. Their close placement causes leakage due to formation of capacitors between the closely placed lines. Additionally, the frequent change of state especially in lower order address buses causes greater losses in the processor. A design methodology that allows the designer to work closely with the router allows optimization of routing signals thereby decreasing their length and increasing the space between them so as to lower their overall capacitance.
- Disconnection of circuits that change state needlessly:**
 By use of clock-gating, circuits that change their states when not required can be eliminated leading to power savings.
- Decrease number of high frequency gates:**
 The use of complex circuits such as look-ahead adders instead of ripple-carry adders increases the performance of the system but at the expense of greater area and power consumption. This is further worsened by the use of large gate drivers and buffers that are required to speed up transitions. By the use of multi-stage, smaller circuits that operate in parallel this can be reduced as while they would be assumed to occupy more area that is not the case, as the gates used are smaller and slower as compared to the faster and larger gates utilized in the power-hungry circuits.[3]

2.1.2 Asynchronous Systems

Synchronous systems currently dominate the market as asynchronous systems are much harder to design. This is because here a great deal of attention has to be paid to avoid changes

in the dynamic states as this leads to race conditions and hazards. However, as technology advances, these problems are being overcome leading to production of asynchronous processors that can match up to their synchronous counterparts. Synchronous design is based on two assumptions:

- All signals are binary
- Time is discrete.

These assumptions simplify design in synchronous systems but asynchronous systems work without the second assumption and hence deliver superior performance.[4]

Fig. 4 shows an asynchronous system where the different modules communicate without clock. These systems use handshaking signals for communication and hence eliminate the global clock.

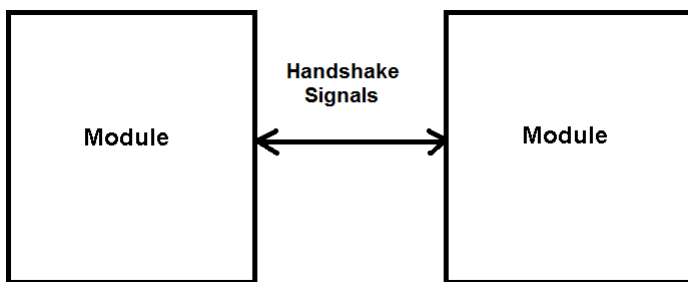


Fig. 4: Asynchronous System with Handshaking

2.1.2.1 POWER OPTIMIZATION

Asynchronous circuits are advantageous over synchronous circuits in the field of low power applications because they eliminate clock and with it most of the power losses caused by it. Hence asynchronous design optimizes power utilization using the following methods:

- Removal of clock trees: Synchronous processors have large clock trees to link the different blocks and keep them synchronized. To drive these clock trees requires high-power buffers which increase the power consumption. These clock trees are high capacitance networks and contribute to power loss. Also clock changes state twice every cycle - on the rising and the falling edges and this increases power loss. Additionally, clock trees do not help in any computation and hence by making processors clock-less an enormous amount of power is saved.
- Elimination of inter-state elements used in pipelining: Synchronous processors are heavily pipelined with a large amount of stages being executed at once. This pipelining requires an enormous number of inter-stage elements such as flip-flops and state elements which are essentially useless for computation. Asynchronous design discards these elements and thus saves both the power consumed by them and also the space occupied by them. Additionally, when a branch instruction is discarded, due to pipelining, a bubble is created. With the help of a additional hardware, synchronous processors eliminate the bubble, to achieve effective throughput, as the hardware fills each bubble on the next clock cycle. This increases clock skew

and hence greater power dissipation.

Asynchronous processors use the bubble as an empty space and fill it up with the next stage immediately. Therefore additional hardware is redundant and power loss decreases.[5],[6]

- Increase in usable clock period: All the inter-stage flip-flops require set-up and hold times and hence a significant portion of the time between the clock edges is wasted, thus reducing the time allowed for computation. The decrease in the usable clock period implies that in synchronous design, the inter-stage elements need to be operated at a faster rate than a single clock period. This requires buffers with even greater driving capacity leading to increased power loss. Asynchronous circuits, on the other hand, use smaller, slower and lower power circuits and eliminate the inter-stage elements. Moreover, due to the slower gates, High-Voltage Threshold (HVT) transistors can be utilized which drastically cuts down the leakage current and power wastage and reduces the die area.
- Shortening interconnecting wires: The discussions above shows that a lot of silicon is saved as elements not necessary for computing are discarded. Due to this, the wires used to interconnect modules are reduced in size. Shorter wires have lesser capacitance and require smaller buffers to drive them and hence reduce wastage of power.[6]

2.2 Static Power Loss Optimization

- High-performance processors are designed via two processes: general purpose silicon process and low-leakage silicon process. The former has larger leakage or standby power but better speed and performance than the latter.

When zero standby power is required, as is the case with battery operated devices, we use High-Voltage Threshold transistors. However there are cases, when there is a large amount of switching activity required, where using SVT logic rather than HVT saves overall power. Hence the Multiply and-Accumulate (MAC) unit, where fast switching is imperative, uses SVT logic whereas the low activity areas such as RAMs use the HVT logic.

Also the high leakage transistors should be detached from the power supply, during periods of inactivity, to remove leakage current altogether.[3]

Certain parts of the processor need lower voltage levels as compared to the rest of the circuit. We implement on-chip voltage regulators for this purpose.

The problem with linear voltage regulators is that they dissipate the excess power in the form of heat thereby wasting a lot of energy. Use of on-chip switching buck converters reduces power dissipation as they are extremely efficient in extracting more charge from the battery. Buck converters can also be switched off when the battery has discharged to such a level that conversion is no longer possible.

Another manner to decrease power loss is to implement voltage scaling with the use of dc-dc converters and performance monitoring circuits. This methodology reduces

the supply voltage to the modules not required to be operated at full speed. Voltage scaling is where the modules that require similar amount of supply are grouped together and provided with that supply which remains constant. The other approach is dynamic voltage scaling where the voltage given to the modules varies depending on their usage at that time.

Although circuits work slowly as their supply voltage is reduced, the aim should be to complete all the operations as fast as possible so that the processor can be put into sleep mode for a longer amount of time thus enabling more power saving.[7] These approaches can be used with synchronous and asynchronous design.

[2] Majid Sarrafzadeh, Foad Dabiri, Roozbeh Jafari, Tammara Massey, Ani Nahapetian, "Low power light-weight embedded systems", Computer Science Department University of California, Los Angeles, Electrical Eng. Dept. / University of Texas at Dallas Electrical Eng. and Computer Sci. Dept. / UC Berkeley, ISLPED'06, 2006.

[3] Octasic Semiconductors, White paper on "Power and Performance Optimizations in High-Performance Multi-Core DSPs", 2008

[4] Mohit Arora, Freescale Semiconductor, "Ultra Low Power Designs Using Asynchronous Design Techniques (Welcome to the World without Clocks)".

[5] Tony Werner, Venkatesh Akella, "Asynchronous Processor Survey", University of California, Davis, IEEE, pp. 67, 1997.

[6] Octasic Semiconductors, White paper on, "Asynchronous Processor Design Evolution", 2010.

[7] Silicon Laboratories Inc. , White paper on "Designing Low-Energy Embedded Systems from Silicon to Software", 2012.

TABLE 1

Transistor Options to Control Standby or Leakage Power in a General Purpose Silicon Process Design

Transistor Type	VT	Description
High-Voltage Threshold (HVT)	~0.34 V	<ul style="list-style-type: none"> • Least leakage current • Least switching speed
Standard-Voltage Threshold(SVT)	~0.27 V	<ul style="list-style-type: none"> • %5 times greater leakage current as compared to HVT • 35% higher switching speed
Low-Voltage Threshold (LVT)	~0.21 V	<ul style="list-style-type: none"> • 25 times more leakage current than HVT • 75% faster switching speed

3 CONCLUSION

With the ever increasing demand for more compact devices with higher performance, the need for chips with low power consumption technology is on the rise. It is necessary for chip developers to look into this field of technology as a long term solution for producing more energy efficient systems. Moreover energy conservation in any form is the order of the day. This paper highlights a few techniques of achieving the same.

ACKNOWLEDGMENT

We would like to show gratitude to our professor, Mr. Sunil Karamchandani, for his valuable insight and support. We would also like to thank another professor, Mr. Tushar Savant for his guidance. We would also to thank our colleague Mr. Heril Chheda for his constructive inputs.

REFERENCES

[1] Lothar Thiele, "Embedded Systems", Swiss Federal Institute of Technology, Computer Engineering and Networks Lab, pp. 9.1-9.16, 2013.