The EEE 190 Undergraduate Project Proposal Template

Undergraduate Project Proposal
by

Roel M. Ocampo
xxxx-xxxxx
B.S. Electrical Engineering

John Richard Ereso Hizon
xxxx-xxxxx
B.S. Electronics and Communications Engineering

Ivan Benedict Nilo Calalang Cruz
xxxx-xxxxx
B.S. Electrical Engineering

Raymund Christian Koon Roque
xxxx-xxxxx
B.S. Electronics and Communications Engineering

Romarie U. Lorenzo
xxxx-xxxxx
B.S. Computer Engineering

Federico Mendoza Ang
xxxx-xxxxx
B.S. Computer Engineering

Siegfred Daquiang Balon
xxxx-xxxxx
B.S. Electronics and Communications Engineering

Advisers:

Professor Louis Poblete Alarcón
Professor Joel Joseph Sacro Marciano, Jr.
Professor Anastacia Ballesil Alvarez
University of the Philippines, Diliman

April 2012

Abstract

The EEE 190 Undergraduate Project Proposal Template

In a *single* well-written paragraph, this is what we'd like to do. Try to cover Need, Solution, Differentiation, Benefit (NSDB). Use the content of this template as an example for formatting your proposal document, **NOT** as a strict guide for the flow of your discussion and what your proposal must contain.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The ability to design and build logic and computational elements that operate at extremely low energy levels is seen as a very important enabler for systems in various application domains such as mobile devices, wireless sensor networks and bio-medical systems [1]. Devices operating at these low energy levels can also take advantage of alternative energy storage and scavenging methods that can lead to almost indefinite operational lifetimes [2], as well as new computing and system paradigms.

Technology scaling and supply voltage reduction have been responsible for the continued energy reduction and performance improvement in complementary static CMOS circuits, the most popular logic topology in use today. However, the increased leakage energy brought about by scaling and V_{DD} reduction is starting to limit the minimum energy that static CMOS circuits can achieve. One low-energy alternative to complementary static CMOS circuits is the sense amplifier-based pass transistor logic (SAPTL) topology.

1.1 Overview

This work presents the sense amplifier-based pass transistor logic (SAPTL) as a low energy alternative logic topology to fully complementary static CMOS logic. The SAPTL takes advantage of the inherent decoupling of logic functionality and circuit gain in pass transistor circuits in order to achieve ultra-low energy operation that is lower than static CMOS, especially in cases where leakage energy becomes a significant component of total circuit energy. This decoupling also allows the improvement of performance through threshold voltage reduction without increasing the total SAPTL energy consumption.

1.2 Proposal Flow and Organization

Chapter 3 presents an overview of the low energy design, starting with the basics of how energy is consumed in digital circuits. Techniques on how energy consumption can be reduced, such as sub-threshold leakage current mitigation, as well as the challenges involved, are mentioned. Metrics showing how these reductions affect the performance of these circuits are then introduced.

Pass transistor logic is introduced in Chapter 2 as a low leakage alternative to complementary static CMOS logic, since it operates inherently without built-in gain elements. The penalties of operating without gain, as well as ways to introduce more gain into the circuit are also presented.

Chapter 4 introduces the basic organization of the sense amplifier-based pass transistor logic (SAPTL) circuit, which is built using (1) an inverted pass transistors tree, or stack, and (2) gain elements in the form of drivers and sense amplifiers. This is followed by an analysis of the delay and energy of these SAPTL building blocks and the implications of combining these two blocks in order to implement boolean functions.

The timing behavior of SAPTL is shown in Chapter 5. A two-phase clocking scheme as well as two asynchronous handshaking schemes are presented. The energy and delay characteristics of logic functions using these timing schemes are then compared with their corresponding complementary static CMOS implementations, highlighting design areas where using SAPTL will be advantageous. Chapter 4 shows several examples of the SAPTL circuits, including practical design and implementation issues as well as measurement results of fabricated circuits.

Chapter 6 presents the Gantt Chart for the project, highlighting the milestones and the deliverables in full detail.

Chapter 2

Related Work

Pass transistor logic (PTL) has been the focus of many research projects [3] ranging from their use as static CMOS replacement circuits for specialized applications, to computer-aided design (CAD) methodologies that take advantage of various PTL characteristics such as area efficiency and regularity [4]. In this chapter, a sampling of various pass transistor logic implementations are described, with the intention of highlighting the relationship between logic functionality, gain stages, performance and energy.

2.1 Basic PTL Topologies

Various pass transistor logic implementations have been reported and can be broadly grouped into two classes based on the pass transistor network (PTN) used: (1) NMOS-only PTNs and (2) PTNs that use both NMOS and PMOS transistors [5]. NMOS-only pass transistor networks are less complex, resulting in (1) lower input and signal path capacitances and (2) smaller area. However, due to the threshold voltage drop needed to keep the NMOS network conducting, the high or logic '1' output voltage will be less than the supply voltage, V_{DD}. In order to recover a rail-to-rail output swing, a simple inverter or buffer can be added at the output of the PTN.

One common form of the NMOS-only PTL topology is the complementary pass transistor logic (CPL) [4]. CPL has complementary inputs and outputs, and usually has CMOS inverters at its outputs to restore the degraded PTL voltage swing due to the NMOS threshold voltage (V_{TH}) drop. Fig. 2.1b shows a CPL implementation of an OR/NOR gate. A number of arithmetic building blocks have been reported using CPL [4, 6, 7], taking advantage of the low input capacitance and reduced transistor count to increase performance.

A variation of CPL is the double pass transistor logic (DPL) [8, 7]. It is composed of

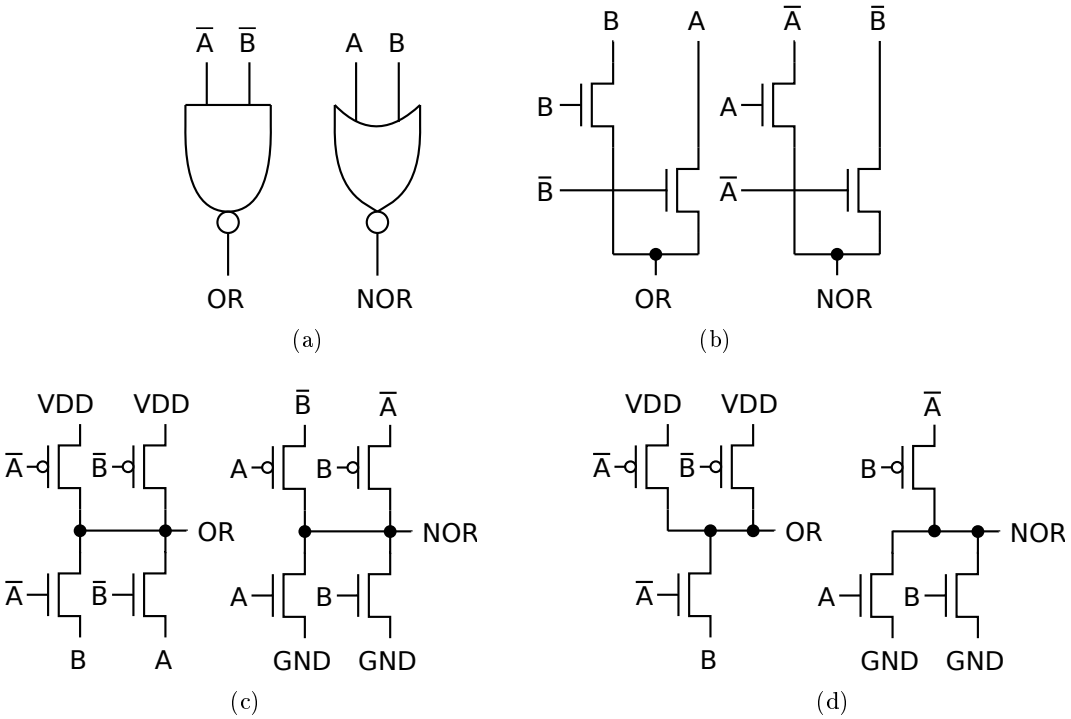

Figure 2.1: OR/NOR implementations in (a) CMOS, (b) CPL, (c) DPL and (d) DVL.

an NMOS PTN and its PMOS equivalent, as shown in Fig. 2.1c. The additional PMOS PTN (1) doubles the signal transmission path, compensating for the performance degradation due to increased path capacitance and (2) allows full swing at the output of the PTN, improving the performance at low supply voltage even with limited threshold voltage scaling. DPL however, has increased input capacitance due to the redundant signal paths present.

Dual value logic (DVL) [9, 10], shown in Fig. 2.1d, is a pass transistor-based logic style derived from DPL by eliminating redundant and slower branches while still maintaining full-rail voltage swing at the output of the PTN. Note that since DPL and DVL circuits use both NMOS and PMOS PTNs, no buffering or voltage swing restoration circuitry is required at their outputs.

2.1.1 Decoupled Operation

Separate optimization strategies for the pass transistor network and the gain stages can be employed due to the reduced coupling between functionality and gain. For example, consider the case of subthreshold leakage and performance. The energy consumed by the CPL multiplexer in Fig. 2.1 can be expressed as:

$$E = E_{dynamic} + E_{leakage} \tag{2.1}$$

where $E_{dynamic}$ is the energy needed to charge and discharge the capacitances in the circuit, while $E_{leakage}$ is the leakage energy dissipated by the circuit. $E_{dynamic}$ and $E_{leakage}$ for a certain operation period, T_{op}, and output swing, dV, can be further expressed as:

$$E_{dynamic} = \alpha\, C_{int,driver}\, V_{DD}^2 + \alpha\, C_{PTN}\, V_{DD}\, dV + \alpha\, (C_{int,buffer} + C_{load})\, V_{DD}^2 \tag{2.2}$$

$$E_{leakage} = V_{DD}\, I_{leak,driver}\, T_{op} + V_{DD}\, I_{leak,buffer}\, T_{op} \tag{2.3}$$

Since the subthreshold leakage energy has been confined to the gain elements, and assuming sneak paths can be eliminated as described in Chapter 4, the various low leakage techniques presented in Chapter 3 can be applied selectively to these circuits, such as increasing threshold voltages, forced stacking or adding header and footer switches. Since these techniques are local to the gain elements, their impact on the pass transistor network are reduced.

The delay of the CPL multiplexer on the other hand, can be expressed as:

$$D = D_{driver} + D_{PTN} + D_{buffer} \tag{2.4}$$

A first order model [11] for the delay of the pass transistor network and the CMOS inverters serving as drivers and buffers gives:

$$D \approxeq \frac{C_{int,driver}\, V_{DD}}{I_{ON,driver}} + \frac{C_{PTN}\, dV}{I_{ON,driver}} + R_{eff,PTN}\, C_{PTN}\, \ln 2 + \frac{(C_{int,buffer} + C_{load})\, V_{DD}}{I_{ON,buffer}} \tag{2.5}$$

The pass transistor network can be optimized for performance by using lower V_{TH} devices, reducing the effective resistance, R_{eff}, of the PTN. This decoupling of subthreshold leakage and performance is normally not possible in static CMOS circuits since the leakage path is the same as the signal path.

Chapter 3

Problem Statement and Objectives

Chapter 4

Preliminary Findings

Chapter 5

Methodology

Chapter 6

Project Schedule and Deliverables

Table 6.1 shows the sample table. You can control its position by right clicking the float and choosing Settings.

Row 1, Column 1	Row 1, Column 2	Row 1, Column 3	Row 1, Column 4	Row 1, Column 5

Table 6.1: Sample Table

Bibliography

[1] T. Sakurai, "Perspectives on power-aware electronics," in *Solid-State Circuits Conference, 2003. Digest of Technical Papers. ISSCC. 2003 IEEE International*, pp. 26–29 vol.1, 2003.

[2] G. Chen, M. Fojtik, D. Kim, D. Fick, J. Park, M. Seok, M.-T. Chen, Z. Foo, D. Sylvester, and D. Blaauw, "Millimeter-scale nearly perpetual sensor system with stacked battery and solar cells," in *ISSCC Digest of Technical Papers*, vol. 53, pp. 288–289, February 2010.

[3] K. Taki, "A survey for pass-transistor logic technologies-recent researches and developments and future prospects," in *Design Automation Conference 1998. Proceedings of the ASP-DAC '98. Asia and South Pacific*, pp. 223–226, Feb 1998.

[4] K. Yano, T. Yamanaka, T. Nishida, M. Saito, K. Shimohigashi, and A. Shimizu, "A 3.8-ns cmos 16x16-b multiplier using complementary pass-transistor logic," *Solid-State Circuits, IEEE Journal of*, vol. 25, pp. 388–395, Apr 1990.

[5] D. Markovic, B. Nikolic, and V. G. Oklobdzija, "A general method in synthesis of pass-transistor circuits," *Microelectronics Journal*, vol. 31, no. 11-12, pp. 991 – 998, 2000.

[6] I. Abu-Khater, A. Bellaouar, and M. Elmasry, "Circuit techniques for cmos low-power high-performance multipliers," *Solid-State Circuits, IEEE Journal of*, vol. 31, pp. 1535–1546, Oct 1996.

[7] P. Cheung, M. Scotti, J. Blake, B. Brewer, R. Grisenthwaite, D. Hitchcox, and P. Shepherd, "High speed arithmetic design using cpl and dpl logic," in *Solid-State Circuits Conference, 1997. ESSCIRC '97. Proceedings of the 23rd European*, pp. 360–363, Sept. 1997.

[8] M. Suzuki, N. Ohkubo, T. Shinbo, T. Yamanaka, A. Shimizu, K. Sasaki, and Y. Nakagome, "A 1.5-ns 32-b cmos alu in double pass-transistor logic," *Solid-State Circuits, IEEE Journal of*, vol. 28, pp. 1145–1151, Nov 1993.

[9] V. Oklobdzija and B. Duchene, "Pass-transistor dual value logic for low-power cmos," in *VLSI Technology, Systems, and Applications, 1995. Proceedings of Technical Papers. 1995 International Symposium on*, pp. 341–344, May-2 Jun 1995.

[10] V. Oklobdzija and B. Duchene, "Synthesis of high-speed pass-transistor logic," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, vol. 44, pp. 974–976, Nov 1997.

[11] J. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective*. Prentice Hall, 2 ed., 2003.