Incorporating Electronics in the Architecture Track of Computer Science, a Comparative Study

Hassan Farhat
hfarhat@unomaha.edu
University of Nebraska at Omaha
Omaha, NE 68182, U.S.A.

ABSTRACT

The latest draft report of the ACM/IEEE Joint Task Force on computer science makes several recommendations[1]. Of these, for computer architecture: a) a shift towards more use of Programmable Logic Devices (PLDs), and b) more emphasis on the study of power constraints.

While the shift towards use of PLDs requires minimal electrical knowledge, the study of power constraints requires better understanding of the electrical topics. With further constraints placed on the number of credit hours, detailed study the electrical topics may be skipped.

The contribution of the paper is to propose a set of electrical topics compiled so as to include under curricula constraints and offer pedagogy of teaching the topics. In addition, the paper presents a comparative study of the electronics topics found in digital design texts[2].

1. INTRODUCTION

Digital circuits design is a central topic of study for computer science and computer engineering. The topic spans a range of topics coverage from the high level of computer architecture [18, 19] to the lower level of Very Large Scale Integration (VLSI) [1]. In prerequisites to studying computer architecture, students are required to study digital design [2] to [12] and computer organization [14] to [17].

In 2001, the Joint ACM/IEEE Task Force has recommended a body of knowledge set for the Computer Science field [26]. An interim review with revisions of the report is found in [27]. The latest draft recommendation is found in [28].

In the latest draft report in computer architecture, the recommendation includes the following paragraph: "*In this KA, multi-core parallelism, virtual machine support, and power as a constraint are more significant considerations now than a decade ago. The use of CAD tools is prescribed rather than suggested."* [p. 201]. The report provides a set of guidelines that are not rigid but can be tailored to meet the need of a particular curriculum.

With respect to electrical topics, the shift towards use of programmable logic devices and CAD tools may represent one extreme where electrical knowledge may not be needed. While studying power as a constraint may represent the other extreme where more electrical topics are needed.

To illustrate, we refer to Fig. 1 and Fig. 2. Fig. 1 is a model of a simple computer as a finite state machine with datapath (FSMD).

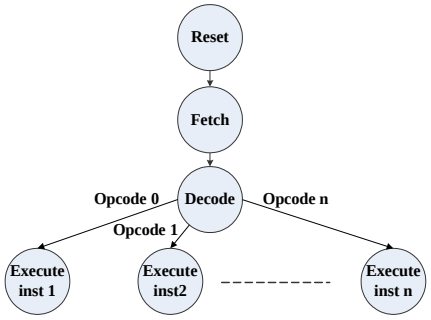

Fig. 1: Computer as FSMD

At the behavior level of hardware programming languages, the design of the FSMD can be realized as a nested set of conditional statements (if and case statements), Fig. 2. The figure shows a general behavioral construct for the design in VHDL. In fact, at the behavioral level, a complete design of an accumulator-based computer can be completed in less than 90 lines of code [25]. As can be seen, many of the electrical constrains that may influence the design are left for the computer aided design tools.

```
Use Library of modularized Module to generate RAM
PROCESS (clk, start)
BEGIN
  IF start = 1 THEN    state = Reset
  ELSIF clk'event and clk = 1 THEN
     CASE state IS
       WHEN reset    -- do reset part, initialize PC, etc.
       WHEN Fetch    -- do fetch part, bring instruction from memory
       WHEN decode -- do decode part, find type of instruction
            CASE ir(15 DOWNTO 10)
              WHEN "000000"      state = execute inst 1
              WHEN "000001"      state = execute inst 2
                 ...
              WHEN "111111"      state = execute inst n
            END CASE
       WHEN execute inst 1       -- execute instruction 1

       WHEN execute inst 2       -- execute instruction 2
            …
       WHEN execute inst n       -- execute instruction n
     END CASE
  END IF
END PROCESS
```

Fig. 2: Computer at behavioral level in VHDL

With the above, the computer design is presented as a simple software program with little or no discussion of electrical constraints. For more in depth understanding of the design one can model the computer at the structural level and study the electrical details of the structural units that compose the design. This leads to incorporating in the study, speed and power requirements.

[1] ACM/EEE 2013 ironman draft version

[2] Part of this work is presented in IEEE EIT08 conference [23]

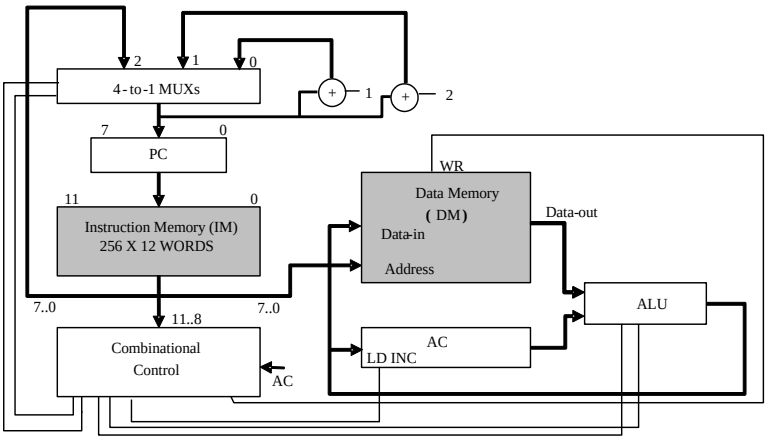

Fig. 3: Schematic of the single-cycle computer

In a traditional hardware track in computer science, a digital design course is offered as a first course in the track. At the gate and transistor level, the study requires prerequisites of electrical and electronics concepts acquired through a sequence of classes in electrical engineering.

To include the needed electrical topics under number of credit hours constraint is a major challenge to the computer science educator. This is particularly true as found in many digital design texts and under the further added constraints by the ACM/IEEE draft report. To properly understand power consumption, timing delays, and interfacing at the structural level, we need electrical knowledge below the logic gate level.

Fig. 3 shows an example of a single-cycle computer designed from several hardware units [20]. In [22] we showed such a design is accomplished using an easy to learn package, Multisim, [13]. A comparative study of computer aided design packages is found in [21]. The need for interfacing study in computer graphics is discussed in [24]. To understand a system electrical details including power consumptions and timing constraints, one needs to understand: a) the electrical properties of the individual units, and b) how they are interfaced.

In earlier work [23], we proposed a set of electrical topics that can be included under curricula constraints. We feel the latest recommendations further support the need to include a set of electronics topics. Our work is related to previous work. We expand on previous work and incorporate power topics discussions. We propose a set of electrical topics to include in a first course in digital design. The topics are proposed so as to complement many of the existing textbooks on digital design. Many of the textbooks lack some of the proposed topics. Either previous knowledge is assumed, or the coverage is not sufficient for a computer science student with no familiarity with electrical topics.

We attempt to present a set of needed topics intended to help the computer science educator in complementing exiting textbooks. Due to the new set of constraints, the educator may choose to cover a subset of the proposed topics. By including the proposed electrical material, the student avoids the need to take additional courses in electrical and computer engineering. To aid and speedup the learning process, extensive use of Multisim (software package) is incorporated. This accelerates the learning process through circuit simulation at the electrical level and, hence, avoids the time consuming process of actual design.

The paper is organized as follows. Section 2 includes, the needed electrical topics. These topics are compiled so as to better understand the electrical properties of logic gates and assume the student had no previous knowledge in electricity. In the remaining sections, we propose teaching methodology samples of the topics in the context of theory and Multisim. In section 3, we discuss how to incorporate Transistor-Transistor-Logic (TTL) and interfacing in Multisim. In section 4 we discuss power consumption as it relates to TTL and data sheets. Based on Moore's law, as the number of circuit elements increases exponentially, power constraints become more important in design decisions. Sections 5 and 6 deal with the need for alternative designs to cope with power constrains. The sections cover Complementary Metal Oxide Semiconductor (CMOS) technologies. In section 7 we include a comparative study of the electrical topics found in popular existing digital design textbooks. The intention of the comparative study is to aid the new educator. The conclusions are given in section 8.

2. THE ELECTRICAL TOPICS GROUPS

Logic circuits are designed using two dominant technologies: transistor-transistor-logic (TTL) and complementary metal-oxide-semiconductor (CMOS). To accommodate the various design platforms (handheld devices, laptops and desktops for example) CMOS designs are offered based on different voltage sources. These include the traditional 5 V technology and the low-voltage CMOS technology (3.3 V, 2.5 V, and 1.8 V).

Today digital designs incorporate different technologies in realization of a digital system. To understand the interfacing requirements and other electrical constraints in design, a designer refers to a unit data sheet. As a result, in considering the proper electrical coverage, an important minimal objective is the need to understand and interpret the various data sheets of primitive elements in combinational and sequential circuits. We compiled a sample of needed topics as presented in Table 1.

The table is composed of 5 groups. Based on the students background some groups maybe skipped. As can be seen in the table, group 1 covers the basic level, Ohm's law. Some textbooks may start at this level. However, the textbooks may not be thorough enough as they may assume previous experience with electrical and electronics concepts leaving gaps in understanding the design process.

Group 2 covers three basic important semiconductor elements diodes, Bipolar Junction Transistors (BJT) and Metal-oxide-semiconductor Field-effect transistor (MOSFET). For gate design, the general functions of the semiconductor components are covered as non-linear devices. We then cover first order approximation of devices.

Group 1: Basic Electricity	Group 2: Semiconductor Elements Function and Characteristics
Ohms law	Diodes, Light-emitting diodes, seven segment displays
Kirchhoff's laws	Bipolar junction transistor
Mesh analysis	MOSFET, NMOS, PMOS
RC circuits	Characteristics curves
Thevenin equivalent	First order models
Group 3: Logic Gate Design	**Group 4: Interpreting Data sheets**
Diode, TTL and CMOS design	Fan-outs
Open-drain and tristate design	Power dissipation (dynamic and Statistic)
Input-output transfer characteristics	Noise margin
Transmission gates	Speed
Group 5: Interfacing	
Unused inputs	
Same series interfacing	
Different technologies interfacing	
Input and output interfacing	
startup issues	

Table 1: Sample topics coverage

In group 3 we consider design of the basic logic gates. Each gate design is considered in the different technologies, diodes, Transistor-Transistor-Logic (TTL), N-channel MOS (NMOS) and complementary MOS (CMOS). The group also considers realization of important gates with non-logic outputs such as tristate and open-drain. It includes as well coverage of transmission gates used in discussion of multiplexer circuits. To understand data sheets we then study the input-output characteristic of the inverter circuit.

With the discussion of the previous groups, interpreting data sheets can be covered clearly. In particular, voltage and current constraints can be covered as related to actual designs. Further, based on the discussion of group 3, students can perform actual computations of the different data columns found in data sheets. This includes, the columns for V_{OH}(min), V_{OL}(max), V_{IH}(min) and V_{IL}(max). It also includes computations of currents into and out of logic gates. Based on the concepts of the current and voltage computations, further important discussions can be carried (fanouts, speed, power dissipation, and noise margins). Additional power dissipation concepts can be discussed (statistic and dynamic) where the load model can be a resistor as in TTL designs or a capacitor as in CMOS designs.

In the final group, interfacing, we cover interfacing by first discussing interfacing of chips from the same technology and series. We then cover interfacing between different series. Interfacing between different series is important given the many low-voltage chips provided by CMOS. We then discuss input and output interfacing. This includes switch and LED interfacing which includes load resistors computations. The final topic in the group is discussion of startup issues where initial memory element values (flip-flops) can be covered.

3. USING MULTISIM

We use Multisim to simulate electrical circuits designs. Simulation avoids the time consuming process of performing actual design. The software has an easy to use graphical user interface where design and simulation can be performed using the same interface. The package can simulate designs from simple RC circuits to simple computer designs [22]. Using simulation helps the students: a) concentrate on concepts, and b) verify these concepts; resulting in a faster learning process.

We illustrate the use of the package in five applications: 1) circuit response to an RC circuit and simulated oscilloscope; 2) load line construction of a common-emitter bipolar junction transistor; 3) transistor-collector characteristic curves; 4) input-output voltage transfer curves of a TTL inverter; and 5) interfacing. The five concepts are clarified significantly when simulated. This is especially true for a computer science student with no formal background in electrical and electronics circuits.

The RC circuit simulation: For discussion of group 1 example we look at Fig. 4 designed in Multisim. The circuit shows an RC circuit with a switch. By pressing the spacebar on the keyboard, the SPDT switch can be connected to the R2 resistor or to the positive side of the voltage source. Hence the circuit can be used to simulate charging and discharging the capacitor C1. The circuit can be used to enforce the concepts of measuring RC constants (approximately 0.63 VCC). This is illustrated in Fig. 5 where the RC circuit response is shown using a simulated oscilloscope. The figure shows two time bars that can be moved to read the RC constant value.

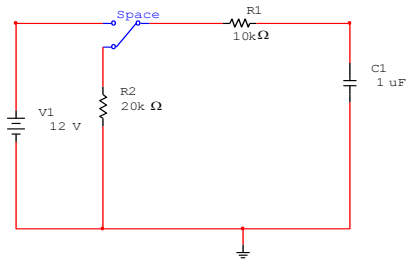

Fig. 4: Using Multisim to simulate RC circuit response

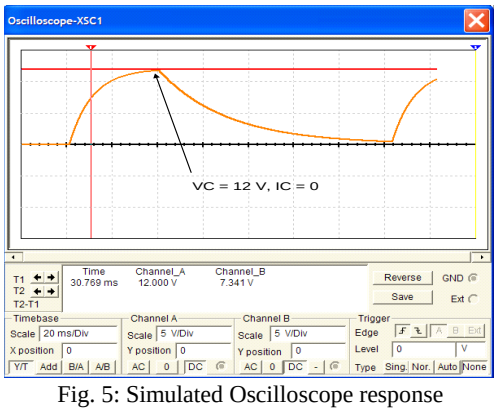

Fig. 5: Simulated Oscilloscope response

Load line construction: Fig. 6 shows an example circuit in Multisim. The design is of a common emitter Bipolar-Junction Transistor (BJT) circuit with several SPT switches and ammeters. The circuit can be used to verify the load line construction concepts from group 2. Here the VCC voltage is fixed through the use of the switches labeled A through J while the IB values are changed using the switches labeled 0 through 9 (the IB value changes due to the different resistors). The corresponding IC and VCE values are read on the simulated ammeters and voltmeters, respectively.

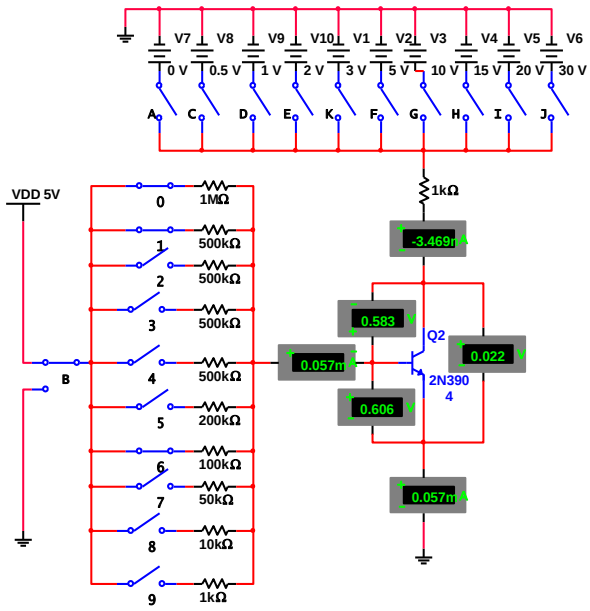

Fig. 6: Use of circuit in characteristic and load line concepts

Transistor-collector characteristic curve: Fig. 6 can be used, as well, to generate transistor-collector characteristic curves by fixing the IB value and adjusting VCC so as to obtain different VCE values. Fig. 7 shows the result using Excel.

Voltage transfer curves of a TTL inverter: In Fig. 8 we construct the design of a standard TTL NAND gate. The design can be converted to an inverter gate by connecting the two inputs to generate a single input circuit. The circuit can be used to form the input-output characteristic curves found in group 3. This is done by adjusting the input voltage values, and reading the corresponding output voltage values. On plotting voltage values we obtain the input/output characteristic curve shown in Fig. 9.

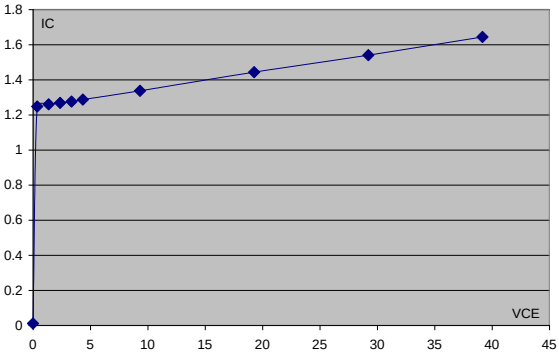

Fig. 7: Transistor-collector characteristic curve

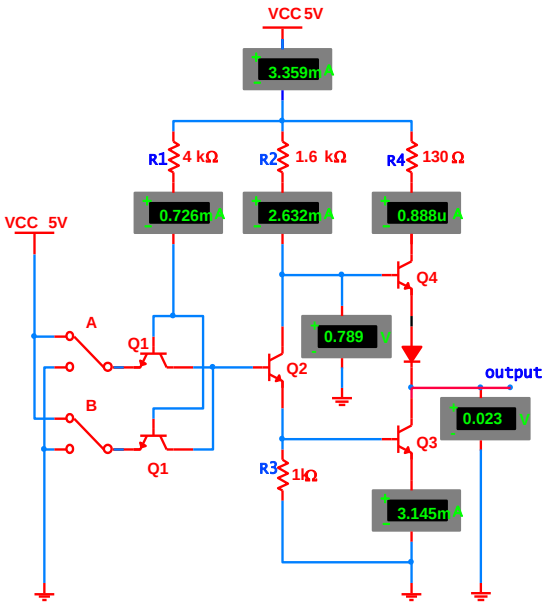

Fig. 8: Simulated NAND gate

Interfacing: An example of Multisim use from group 5, interfacing, is shown in Fig. 10. The figure shows interfacing of 2 V logic to 5 V logic. For the driver HIGH output (Fig. 10 (b)) the HIGH output is not recognized as a HIGH by the load (U6). As a result, the circuit response is the same for both inputs (Fig. 10 (a) and (b)).

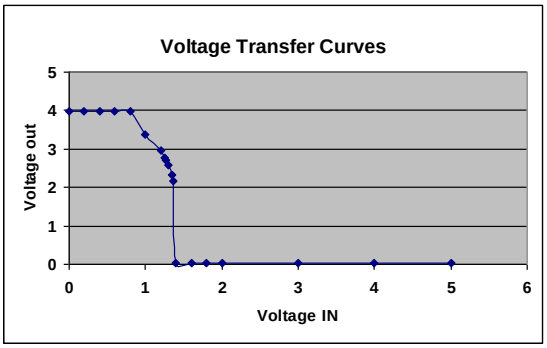

Fig. 9: Sample input/output characteristic curve

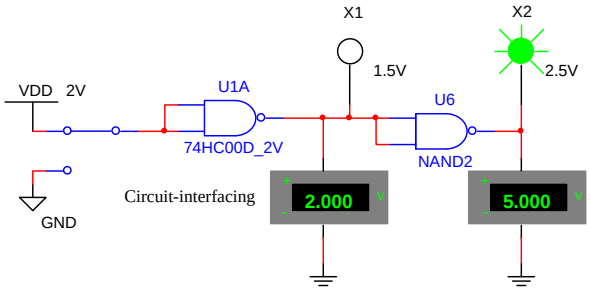

Fig. 10 (a): Input 1; output of U1A = 0V (logic 0) and output of U6 = 5V

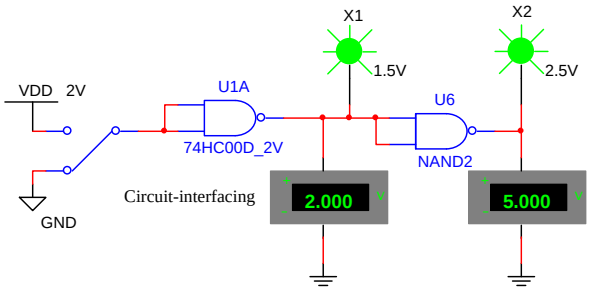

Fig. 10 (b): Input 0; output of U1A = 2V (logic 1); output of U6 should be 0V (logic 0)

We expand next on discussion of power consumption as presented in group 4. Based on the ACM/IEEE task force [28], more emphasis is placed on power consumption.

4. POWER CONSUMPTION CONSTRAINTS

With the increase in switching components on a VLSI chip and the increase in components with processing units, the latest ACM/IEEE recommendations include coverage of power consumption topics.

Consider Fig. 11. In order for a digital circuit to work, it requires a certain amount of electrical power. When a digital circuit uses power to operate, we say the circuit dissipates (or consumes) power. We use the figure in illustrating power constraints. By definition the power consumed by the circuit is $P = V_{CC} . I_{CC}$.

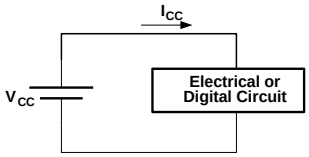

Fig. 11: Example circuit for power discussion

Multisim can be used to speedup the discussion of power consumption. We illustrate how power consumption can be computed as applied to TTL, in particular the NAND circuit shown in Fig. [9]. TTL data sheets supply two types of currents I_{CCH} and I_{CCL}, corresponding to current demands on output HIGH and LOW, respectively. The currents are given based on open circuit outputs; that is no load is connected to the output. Three columns for current are given, minimum, typical and maximum.

Normally the currents are different. Power dissipation is based on the average current $I_{CC}(avg) = \frac{I_{CCH} + I_{CCL}}{2}$. The average power dissipated is $P_D(avg) = I_{CC}(avg) \times V_{CC}$. In computing I_{CC}, the column for typical current is used.

Data sheets provide power dissipation as well, 40 *mW/chip*. The power dissipation as well as the current values can be verified and theoretically computed using Multisim (refer to Fig. [9] where I_{CC} is read below the V_{CC} value). The power computation is computed as

$$P_D(avg) = n\left(\frac{I_{CCH} + I_{CCL}}{2}\right)V_{CC} \quad (1)$$

where n is the number of gates on a chip.

The need for alternative design technology is then discussed based on the increase in number of switching elements based on Moore's law (predication). With the number of switching elements exceeding 2 billion elements, this leads to today's CPU designs using Complementary Metal Oxide Semiconductor (CMOS).

5. METAL OXIDE SEMICONDUCTOR TRANSISTORS

Here we discuss the Metal Oxide Semiconductor (MOS) transistor. In the next section we propose discussion of dynamic power in the context of the proposed topics coverage.

Fig. 12 shows a 2D view of a MOS transistor (Enhancement-Mode Metal-Oxide-Semiconductor Field-Effect-Transistor, E-MOSFET).

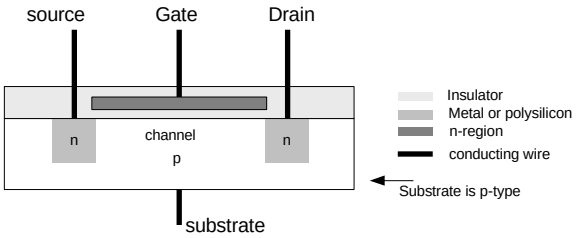

Fig. 12: n-type E-MOSFET Transistor

Similar to a TTL transistor, the circuit is composed of three inputs (source, gate and drain). We show a 3D view of the gate. The gate is characterized by a length and width. It is surrounded by an insulator (SiO_2, glass) as shown in Fig. 13.

Fig. 13: 3D view of gate part of E-MOSFET transistor

In the discussion, brief explanations are made about the transistor as a controlled switch that connects the source to the drain. The gate input functions as the switch control. This is shown in Fig. 14 part (b). In part (a) the insulator material results in a capacitor model between the substrate and the gate. By applying a positive voltage on G we induce a negative channel of electrons in substrate below the gate. This results in a closed switch by inducing a conduction path from source to drain. Similarly, a 0 voltage on the gate input removes the conduction path and keeps the switch open.

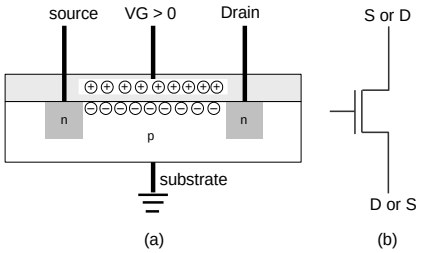

Fig. 14: (a) Capacitor effect, MOSFET with conducting channel, (b) MOSFET as a switch

The above transistor design is called n-type E-MOSFET (abbreviated as nMOS). The n is used to indicate that conduction is due to negative carriers. Similar design can be completed with the substrate semiconductor of type n, the gate and source of type p, resulting in pMOS, Fig. 15.

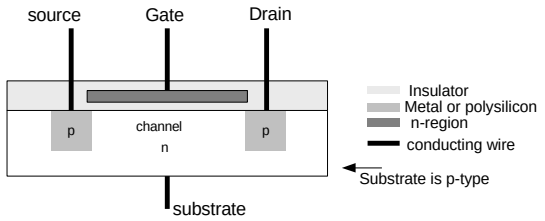

Fig 15: p-type E-MOSFET

We discuss next power consumption in CMOS circuits

6. DYNAMIC POWER CONSUMPTION

Consider Fig. 16. The figure represents the design of a NAND gate in CMOS. The design is called CMOS (Complementary Metal Oxide Semiconductor) since the design incorporates both nMOS and pMOS transistors (alternative designs that use nMOS only use more power are not as common). In the figure, alternative switch representation for the two types are shown in Multisim.

We introduce dynamic power to the students by breaking the design, as is customary, into two networks, a pull-up and a pull-down networks. The pull-up network is composed of the pMOS transistors Q1 and Q2. Similarly, the pull-down network is composed of the nMOS transistors Q3 and Q4[3].

By inspecting all steady state input combinations we show that at no time, there is a direct path from the power source (V_{DD} for CMOS circuits) to ground. As a result, the current is 0 (the gate substrate forms a capacitor; the steady state current gate current is 0). Hence the power dissipation is approximately zero. We contrast this with the TTL alternative design of the NAND gate. The power consumed when the gate inputs/outputs are not changing is called static power dissipation. Hence, we emphasize the advantage of CMOS over TTL in terms of static power consumption.

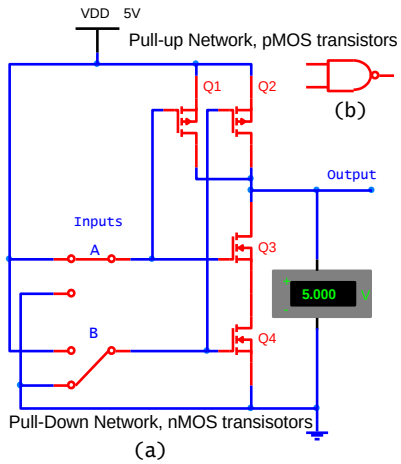

Fig. 16: (a) CMOS design of a NAND gate, (b) NAND gate

We then introduce and compute dynamic power consumption. In designs of CMOS circuits, dominant power loss is due to dynamic power[4]. This occurs as the gate inputs/outputs are changing. When interfacing TTL circuits, the dominant model used is a load resistor. When interfacing CMOS circuits, however, the dominant model used is a capacitive load.

In capacitive loads, the steady state current is zero. This part of the discussion can be illustrated by referring to the RC circuit (Fig. 4 and Fig. 5). The capacitor can be considered as the load capacitance. In such a circuit, the steady state current is 0 since the capacitor voltage is 12 V or 0 V (capacitor is fully charged (switch connected to 12 V) or fully discharged (switch connected to R2)). Hence, in this mode, after charging the capacitor, the power consumed is 0 W. This is not the case if a resistor replaces the capacitor simulating a TTL circuit.

While minimal power is consumed in steady state, CMOS circuits consume power as the capacitive loads are charging. For CMOS gates, the load inputs are modeled as capacitors (first order approximation). While a capacitor is charging, the consumed power is called *dynamic power* consumption, $P(t) = V_{DD} \times I_{DD}(t)$ (I_{DD} is used for CMOS).

For an RC circuit, the current I_{DD} is a function of time as indicated in the equation. Hence one is interested in the average power consumption over a time interval, T. The average power can be computed using the equation

$$Paveg = \frac{1}{T}\int_0^T p(t)dt = \frac{V_{DD}}{T}\int_0^T i(t)dt \qquad (2)$$

With $$i(t)dt = \frac{dQ}{dt}dt = \frac{d(CV_c)}{dt}dt = C\frac{d(V_c)}{dt}dt = CdV_c \qquad (3)$$

(V_C is the capacitor voltage, and $Q = C \times V_C$ is the charge accumulated on the capacitor) we obtain

$$Paveg = \frac{V_{DD}}{T}\int_0^T i(t)dt = \frac{CV_{DD}}{T}\int_0^T dV_c = \frac{CV_{DD}}{T}(V_c(T) - V_c(0)) \qquad (4)$$

Hence $$Paveg = \frac{V_{DD}}{T}\int_0^T i(t)dt = \frac{CV_{DD}}{T}\int_0^T dV_c = \frac{CV_{DD}^2}{T} \qquad (5)$$

Assuming the capacitor, was fully discharged at time 0 and fully charged at time T, we have $V_C(0) = 0$ and $V_C(T) = V_{DD}$. The total energy delivered to the RC circuit by the power supply is $$E(t) = T \times \frac{CV_{DD}^2}{T} = CV_{DD}^2 \qquad (6)$$

Equation (5) for power consumption is called dynamic since it is a function of charging/discharging the capacitor. Once this is done, since the current I_{DD} is reduced to 0 A, no additional power consumption is caused. If the output is changing at a frequency, *f*, then the total dynamic power consumption can be computed as follows. Over one period of time *T*, the capacitor charges (output transition from 0 to 1, and then discharges, transition from 1 to 0 (no energy is delivered to the circuit by VDD). Hence, for a circuit with outputs changing at a rate, *f*, the dynamic power consumed is

$$P_{dynamic} = f \times C \times V_{DD}^2 \qquad (7)$$

The power consumed by a CMOS circuit is dominated by dynamic power consumption. To illustrate this we look at the ALVC 7400 package for example. It contains 4 NAND gates, designed using low power CMOS with $V_{DD} = 3$ V. The typical current supplied to the package is 2.5 μA per gate. Hence the typical power consumed is

$$P = V_{DD} \times I_{CC} = 3\text{ V} \times 2.5\ \mu\text{A} = 7.5\ \mu\text{W}$$

[3] The transistors symbols show alternative transistors representation.

[4] Today, due to the thin gate oxide insulator, leakage currents play important roles as well.

When compared to the power consumed by TTL, TTL uses over 1000 times more static power than CMOS.

The power consumption values are based on the output of a gate connected to a standard capacitive and resistive load. To account for dynamic power consumption we need to use the CV^2 equation. As a result, knowledge of the C value is needed. In addition, one needs to know how often the output is changing. Alternatively, one can use a current rating found in the data sheets, ΔI_{CC}. This represents the current as a result of charging the load capacitor. To compute the dynamic power consumed we use $P = V_{CC} \times \Delta I_{CC} \times f$, where f represents the fraction of time the output is changing.

7. COMPARATIVE STUDY OF TEXTBOOKS

We looked at several textbooks in the field to determine the electrical coverage in each. None of the textbooks covered all the topics found in group1. Outside group 1 and semiconductor characteristic, [12] is the most comprehensive. For transmission gates coverage, [7] provides the best coverage. For interfacing and sizing the resistor values, [8] provides very detailed discussion. Ref. [8] also devotes: a) a special chapter "Practical consideration for Digital Design", b) a chapter on interfacing to analog devices, and c) an appendix on basic electricity. We feel the basic electricity discussion is incomplete (KVL, KCL, and RC circuits discussions are missing for example).

Ref. [11] contains good introductory discussion of CMOS design at the layout level. The textbook skips TTL discussions and does not contain data sheets and logic interfacing discussions.

Ref. [9] includes good coverage of both TTL and CMOS circuits. It also contains a good discussion on characteristic curves. Ref. [9] is a classical text on the topic but assumes previous knowledge in electricity and does not include data sheets discussion. Ref. [10] is a good textbook that is comparable to [8]. Ref. [3] assumes previous knowledge in basic electricity and semiconductor technology. Otherwise the textbook provides detailed and practical discussion on digital electronics topics.

We think the combination of [12] and [11] provide the most comprehensive discussions. Ref. [12] provides the best digital electronics topics. Ref. [11] provides the best coverage on semiconductor technology.

8. CONCLUSION

In this paper we presented a set of electrical topics that can be included in the architecture track of computer science. With the latest ACM/IEEE task force recommendation, more discussion can be devoted to power constraints. We proposed coverage of the topics in Multisim. We also presented a brief comparative study of textbooks in the field. The proposed work is intended in helping the computer science educator; it incorporates a set of electric concepts under the computer architecture track. With the increasing power constraints the trend is to concentrate discussions on CMOS circuits.

9. REFERENCES

[1] N. Weste and D. Harris, "CMOS VLSI Design: A Circuits and Systems Perspective 4th edition", Addison Wesley, 2011.

[2] S. Brown and Z. Vranesic, Fundamentals of Digital Logic with VHDL Design with CD-ROM, 3rd edition, McGraw-Hill, 2009.

[3] N. Cook, Digital Circuits with PLD Integration, Prentice Hall, 2001.

[4] A. Dewey, Analysis and Design of Digital Systems with VHDL, PWS publishing, 1997.

[5] T. Floyd, Digital Fundamentals: A System Approach, Prentice Hall, 2013.

[6] D. Gajski D, Principle of Digital Design, Prentice Hall, 1997.

[7] R. Katz and B. Gaetano, Contemporary Logic Design, 2nd edition, Prentice Hall, 2005.

[8] W. Kleitz, Digital Electronics A Practical Approach with VHDL, 9th edition, Prentice Hall, 2012.

[9] M. Mano, Digital Design, 3rd edition, Prentice Hall, 2003.

[10] R. Tocci, N. Widmer and G. Moss, Digital Systems Principles and Applications, 11th edition, Prentice Hall, 2011.

[11] J. Vyemura, A First Course in Digital Systems Design an Integrated approach, Brooks/Cole, 2000.

[12] J. Wakerly, Digital Design Principles and Practices 4th edition, Prentice Hall, 2006.

[13] Electronics Workbench, www.electronicsworkbench.com

[14] J. D. Carpinelli, Computer Systems Organization and Architecture, Addison-Wesley, 2000.

[15] D. Patterson and J. Hennessy, Computer Organization and Design, Revised 4th edition, Morgan Kaufmann, 2011.

[16] W. Stallings, Computer Organization and Architecture, 9th Edition, Prentice Hall, 2013.

[17] D. Harris and S. Harris, Digital Design and computer Architecture, Morgan Kaufman, 2007.

[18] J. P. Hayes, Computer Architecture and Organization 3rd edition, McGraw-Hill, 1998.

[19] J. Hennessy and D. Patterson, Computer Architecture: A Quantitative Approach 5th Edition, Morgan Kaufmann, 2011.

[20] H. Farhat, "Integrating Hardware and Software in Early Research in Computer Science", Proc. ISCA CAINE, Honolulu, Hawaii, USA, 2005, pp. 390-394.

[21] H. Farhat, "A comparative study of Max+PLUS II and Quartus schematic capture tools", International Conference on Systems Engineering, 2006, pp. 119-124.

[22] H. Farhat, "An interactive computer Design in Multisim", Proc. ISCA CAINE, 2008.

[23] H. Farhat, "Integrating Electronics in Computer Science under Curricula Constraints, a Comparative Study", IEEE International Conference on Electro Information Technology, 2008.

[24] H. Farhat, "Incorporating Output Interface in Studying Computer Graphics in Computer Science", to appear Proc. FECS, 2013.

[25] J. Hamblen and M. Furman, Rapid Prototyping of Digital Systems, a tutorial approach, Kluwer Academic Publishing, 2001.

[26] http://www.acm.org/education/curric_vols/cc2001.pdf

[27]http://www.acm.org/education/curricula/ComputerScience2008.pdf

[28] http://ai.stanford.edu/users/sahami/CS2013/

Received: August 2013. Accepted: August 2014.