Document: 2004-327
Division: Computers in Education

A Hybrid Software/Hardware Approach for Teaching Digital Logic Design for Computer Science Undergraduates

Mahmoud K Quweider, Anna Hernandez

CS/CIS Department
University of Texas, Brownsville
80 Fort Brown
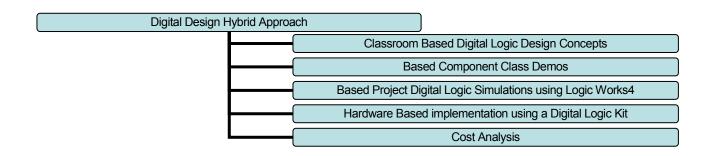**Brownsville, TX 78521 U.S.A.**
1-956-574-6605

Abstract
As a means to stress basic concepts in digital logic design for Computer Science majors at our minority based institution, a hybrid approach was creatively followed to enforce key concepts in digital logic design. The approach allows students, who are enrolled in the senior project course, to simulate a digital logic project of their choosing and implement it both in software and hardware. The approach was found to be enriching and very effective in enforcing concepts taught in the class as well as in drawing the students' attention to issues of cost and real world implementation by using a hardware-based kit for the projects realization; additionally, students where asked to run a component and cost analysis of their projects as well. The success of the approach will be departmentalized for students who wish to conduct their research in the digital logic design area.

Introduction
The University of Texas at Brownsville is a minority based university with almost 85% being of Hispanic background. In order to foster and encourage retention of students in the Computer Science area, we have improved many key courses by either creating a companion web site that complements the course or by improving the course content through several lab modules with a hands-on approach. One of the courses that were targeted for improvements is Digital Logic and Computer Architecture and the following senior project in the same area. Students who were interested in digital logic design as a topic for their senior project were allowed to choose a topic such as the design of a digital clock, a digital calculator or a simple electronic game; the students prepared for the project by implementing in software, in a gradual approach and under the instructor's supervision, many of the basic functional units making up their projects. This software simulation stage allowed the students to grasp many of the key concepts underlying their project. The stage additionally isolated the more subtle and tricky problems that usually arise in a pure hardware implementation such as module

interfacing and debugging. As the software stage was verified, students were asked to create components and cost analysis through which they came up with the list of components needed for the hardware implementation and the cost of these components. This stage was followed by the actual hardware implementation, which was largely based on their working in the software implementation, using specialized hardware-kits which resemble lab in a suitcase. The kits include basic probes, power supplies, clocks, switches and resistors among other components. The advantage of the use of the dual approach was prominent as students wasted little time on the actual implementation and were focused on hardware issues which are hard to resolve in a simulated environment. Figure 1 shows the approach taking by the authors in which the last two components, hardware implementation and cost analysis, were added to the traditional approach to enhance and improve retention and enforce the key concepts.

Digital Design Hybrid Approach
Classroom Based Digital Logic Design Concepts
Based Component Class Demos
Based Project Digital Logic Simulations using Logic Works4
Hardware Based implementation using a Digital Logic Kit
Cost Analysis

Figure 1. Hybrid Approach for Digital Logic and Computer Architecture

Software Implementation
We demonstrate the software implementation side of the hybrid approach through the design of a digital clock circuit, which was part of a senior project conducted by Anna Hernandez based on concepts from the books by Bignell[1] and Mano[2] respectively. The design of the circuit consists of several devices, or functional units provided by Logic Works4 (a digital logic software simulator.) Among these devices are the 4-bit binary counters and the 7-Segment displays which we use here for illustration purposes. A 4-bit binary counter is a register that outputs the binary number sequence from 0000 (0 decimal) to 1111 (15 decimal) upon the application of input pulses. The device consists of four inputs in addition to a load , a clear and a clock input. The four inputs are used to load a specific number whenever the load switch is activated; the clear switch forces the counter to reset or go back to zero. The clock switch is used to apply the input pulses which will force the counter to circle through its finite states of 0 to 1 to 2, all the way to 15 and then back to 0.

Binary Counter

CLK   C0

D3   Q3
D2   Q2
D1   Q1
D0   Q0

LD
CLR

Figure 2. Logic Works4 Binary Counter

The 7-Segment Display consists of 7 inputs, one for each line segment of the display number. Four of these devices must be used in order to display the hours and minutes appropriately. For example, the minutes go from 00 to 59 and therefore two 7-Segment Display devices are used, one device for the unit of the minutes (1 to 9) and a second device for the tens of minutes (10, 20, 30, 40, 50). The following example should help illustrate the use of the two devices in displaying the minutes:

Tens   Ones

dot   dot
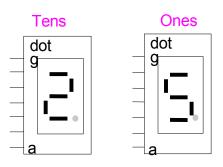g   g

a   a

Figure 3. Logic Works4 7-Segment Display Displaying 25 Minutes

Since the function of this device is to display the number which the counter is outputting, a special device that works as a decoder was created because the binary counter has four outputs while the 7-Segment Display has seven inputs. The decoder consists of four inputs and seven outputs. The inputs are connected to the four outputs of the binary counter while well the seven outputs are connected to the 7 inputs of the display device. This device translates the output of the counter and indicates to the display device which line segments to turn on in order to display the number.

a

f   g   b

e   c

d

Figure 4. Labels for the 7-Segment Display

Each line segment is a separate function. For example, if a zero is to be displayed, then the line segments of a, b, c, d, e, and f have to be turned on. Therefore, the use of the truth table and of the K-Map is needed to simplify each function and design the circuit of the device.

INPUTS				UNITS							NUM
W	X	Y	Z	A	B	C	D	E	F	G	
0	0	0	0	1	1	1	1	1	1	0	0
0	0	0	1	0	1	1	0	0	0	0	1
0	0	1	0	1	1	0	1	1	0	1	2
0	0	1	1	1	1	1	1	0	0	1	3
0	1	0	0	0	1	1	0	0	1	1	4
0	1	0	1	1	0	1	1	0	1	1	5
0	1	1	0	1	0	1	1	1	1	1	6
0	1	1	1	1	1	1	0	0	0	0	7
1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	1	1	1	1	0	1	1	9
1	0	1	0	1	1	1	1	1	1	0	X
1	0	1	1	0	1	1	0	0	0	0	X
1	1	0	0	1	1	0	1	1	0	1	X
1	1	0	1	1	1	1	1	0	0	1	X
1	1	1	0	0	1	1	0	0	1	1	X
1	1	1	1	1	0	1	1	0	1	1	X

Table 1. Truth Table for the Seven Segment Display

Since the circuit design of a digital clock is too complex, the project was split into several parts. The first part to be created was the unit for the display of minutes while the second part was the unit for the display of hours. First, a binary counter, a decoder device (labeled UNIT), and a display device were connected together to display the ones part of the minutes. For this device, whenever the counter reaches the number nine, it would load a zero and restart as shown in figure 5.
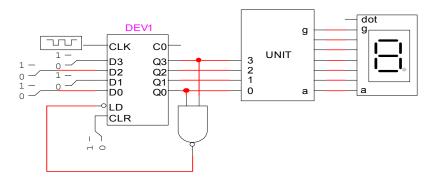

Figure 5. Minute Display for Digital Clock

As we can see by the design, the output Q_3 and Q_0 are connected with an 'AND' gate whose output is connected to the load switch. In view of the fact that the binary representation of the number 9 is 1001, this means that whenever the counter reaches the number nine, the load switch will get activated and load the number zero. The final circuit design consists of connecting different sections that implement the minutes and the hours using the basic concept mentioned above as shown in figure 6.
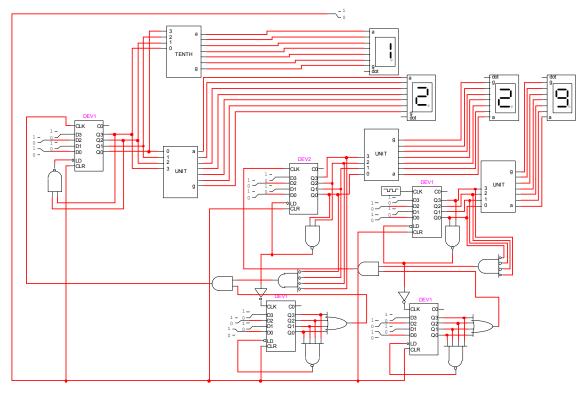
Figure 6. Software Simulation of the Digital Clock

At this stage the digital clock is fully functional in software form. The main basic ideas for the design have been emphasized and well understood. Following this stage, the actual implementation commenced and focused on hardware related issues such as the use of the power supply, wiring, use of resistors and other electrical components. These issues are taken care of in the hardware implementation.

Hardware implementation and Cost Analysis
Now that the software simulation phase is finished, the final phase of this project is the implementation of the circuit in a digital tool kit. In order to complete this phase, research for integrated chips, electrical components and wires was conducted using the Jameco electronics website. The Jameco website[3] provides detailed information of all the different types of devices available in their stock. In order to select which devices were needed for this particular project, a search was conducted for counters, NAND gates, NOR gates, LED (Light Emitting Devices), resistor and wires. The following device list describes the components that were needed for the implementation of the digital clock circuit as well as the cost of the components:

Product Number	Description	Quantity	Price	Total
74LS02	Quad 2-input NOR gate	1	$0.25	$0.25
74LS10	Triple 3-input NAND gate	1	$0.29	$0.29
74LS47	BCD to 7-seg. Decoder/Driver (O.C.)	3	$0.99	$4.95
74LS76	Dual JK flip-flop w/preset & clear	1	$0.69	$0.69
74LS90	Decade counter	2	$0.49	$0.98
74LS92	Divide-by-12 counter	1	$0.45	$0.45
LDD51121	Red LED	4	$1.25	$5.00
108329	540 pcs. 1/8 watt Resistor Kit	1	$24.95	$24.95
	Total:			$37.56

Table 2. Component and Cost Analysis for the Digital Clock

Hardware implementation provided detailed information about each component. A sample of these components and their description follows: Product Number 74LS02, a Quad 2-input NOR gate, contains four independent gates each of which performs the logic NOR function. The output of this device is connected to the LED of the tens of the hour to display the number 1 whenever the hours are from 10 to 12. The chip has fourteen pin inputs of which pin number 7 is the ground connection and pin number 14 is the voltage (power supply) connection as shown in figure 7.
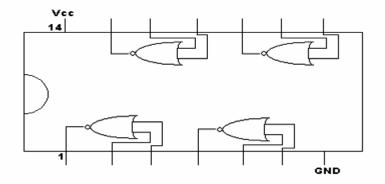

Figure 7. Quad 2-input NOR gate (Product Number 74LS02)

Product Number 74LS10, Triple 3-input NAND gate, contains 3 independent gates each of which performs the logic NAND function. The chip has fourteen pin inputs of which pin number 7 is the ground connection and pin number 14 is the voltage connection.
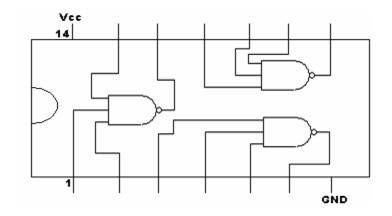
Figure 8. Triple 3-input NAND gate (Product Number 74LS10)

Product Number 74LS47, BCD to 7-segment Decoder/Driver (O.C.), was used for controlling the LED's directly. This device receives information from the counter where it decodes the information received and sends the decoded output to the LED to display the desired number. The chip has sixteen pin inputs of which pin number 8 is the ground connection and pin number 16 is the voltage connection. The inputs for this chip include pin numbers 1, 2, 6, and 7, which are connected to the counter for the receiving of information. The outputs for this chip include pin numbers 9, 10, 11, 12, 13, 14, and 15, which are connected to the LED in order to transmit and display the desired number. The LT pin is the Lamp Test input and when a low signal is applied to it, all segment outputs are turned on. RBI pin, Ripple Blanking Input, and BI/RBO pin, Blanking Input or Ripple Blanking Output, can be used to control the display intensity by varying the duty cycle of the blanking signal. For the implementation, BI/RBO pin is not being used and LT and RBI are connected to the voltage. The connection between the outputs for this chip and the LED is accomplished with the use of resistors.
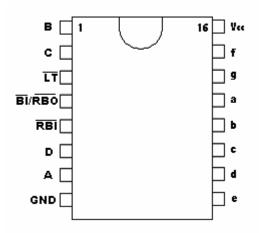

Figure 9. 7-segment Decoder/Driver (O.C.) (Product Number 74LS47)

Product Number 74LS76, Dual JK flip-flop with preset and clear, contains two independent J-K flip-flops with individual J-K, clock, preset, and clear inputs. The main purpose of this device is to control the change in the hours output depending on the output from the counter which controls the tens of minutes. In other words, when the counter outputs the number 5 this device will become active and will not change the current hour until the output from the counter changes from 5 back to 0. The chip has sixteen pin inputs of which pin number 13 is the ground connection and pin number 5 is the voltage connection.
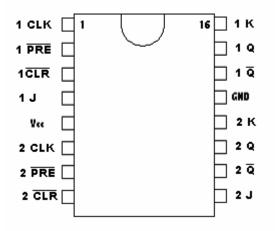

Figure 10. Dual JK flip-flop with preset and clear (Product Number 74LS76)

Product Number 74LS90, Decade counter, contains four master-slave flip-flops and additional gating to provide a divide-by-two counter and a three-stage binary counter for which the count cycle length is divided-by-five for this specific counter. The main purpose of this device is to count from 0 to 9 and to activate the next counter when the number 9 is reached. This is accomplished by connecting output QD, of this device, to Input A of the next counter. To reset from 9 to 0, output QA is connected to Input B of the same device. This chip has fourteen pin inputs of which pin number 10 is the ground connection and pin number 5 is the voltage connection.
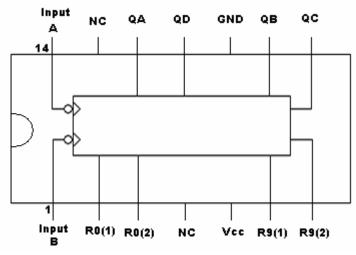

Figure 11. Dual Decade counter (Product Number 74LS90)

Product Number 74LS92, Divide-by-12 counter, contains four master-slave flip-flops and additional gating to provide a divide-by-two counter and a three-stage binary counter for which the count cycle length is divide-by-six. The main purpose of this device is to count from 0 to 5 and to activate the next device, designated to control the hours, when the number 5 is reached. This is accomplished by connecting the output QC of this device to the clock of the J-K Flip-Flop.   In order to reset the device, the output QA and the input B, in the same device, must be connected together to get the desired result. The chip has fourteen pins of which pin number 10 is the ground connection and pin number 5 is the voltage connection.
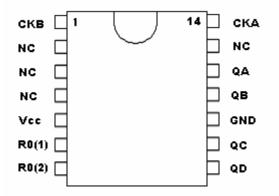

Figure 12. Divide-by-12 counter (Product Number 74LS92)

Product Number LDD51121, Red LED, is a double digit LED display unit that is connected to the BCD-to-7-segment decoder. The main purpose of this device is to display the current counter number which has been translated by the decoder device. Since this device is active low, pin number 13 and 14 are connect to the voltage.
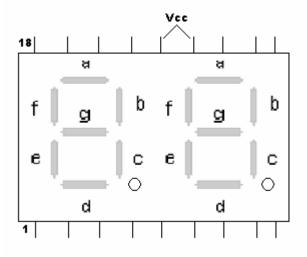

Figure 13. Double digit Red LED (Product Number LDD51121)

One additional piece of information provided by the hardware implementation for this device is the segments for which each pin activates as shown in table 3.

PIN NO.	SEGMENT
1	E of Digit 1
2	D of Digit 1
3	C of Digit 1
4	DP of Digit 1
5	E of Digit 2
6	D of Digit 2
7	G of Digit 2
8	C of Digit 2
9	DP of Digit 2
10	B of Digit 2
11	A of Digit 2
12	F of Digit 2
13	Voltage of Digit 2
14	Voltage of Digit 1
15	B of Digit 1
16	A of Digit 1
17	G of Digit 1
18	F of Digit 1

Table 3. Pin activation for Each Segment

Having gathered all information about the microchips, resistors and wire connectors, the implementation of all components to the digital tool kit was the next logical step. For the units of the minutes, the microchips needed were a decade counter and a BCD-to-7 segment Decoder/Driver. The output of the last device, the BCD-to-7 segment Decoder/Driver, is connected to a Dual Red LED of the second digit. For the tens of minutes, the microchips needed were a divide-by-12 counter and a BCD-to-7 segment Decoder/Driver. The output of the last device is connected to a Dual Red LED, but to the first digit. For the hours section, a Dual JK flip-flop, decade counter, Triple 3-input NAND gate, Quad 2-input NOR gate, and the BCD-to-7 segment Decoder/Driver are connected together, and whose output is connected to a Dual Red LED. Figure 14 shows the actual kit which was used to create the digital clock. At this stage the two approaches, software and hardware, have been implemented and verified. The hardware implementation proceeded smoothly as it built on top of the software implementation and focused on the power, wiring and connection issues. As many of the students indicated, this stage was very crucial in clarifying many issues that are not tackled in the software simulations such as what type of resistors to use, nature of active low and active high devices and module interfacing just to mention a few.

Figure 14. Digital Clock implementation using Hardware Kit

Conclusion

The hybrid approach of software/hardware has shown its effectiveness in enforcing key digital logic design concepts by actual hardware implementation using the digital logic kits. The approach promotes greater understanding by incorporating the theory and the software simulation with an actual hand-on approach. In addition to allowing the student to research, using the Internet, the component and cost analysis has the extra advantage of allowing the student to acquaint themselves with the logic families, their capabilities and variations. Another advantage of the hybrid approach is to allow the student to devise components hierarchically when there is a shortage in the number of available components needed for the hardware implementation.

Bibliography

1. Bignell, James and Donovan, "Digital Electronics," 4th edition, Delmar Thomson Learning, 2000.
2. Mano, Morris and Kime, Charles "Logic and Computer Design Fundamentals," 2nd edition, Prentice Hall, 2001.
3. Jameco Electronics Website at http://www.integrated-circuits.com.

Biographical Information

MAHMOUD K QUWEIDER
Mahmoud Quweider is an assistant professor and the chair of the Computer Science Department at the University of Texas, Brownsville (UTB). He has a PH.D in Engineering Science from the University of Toledo, Ohio and several Masters from the same university. Prior to joining UTB, he has worked as a Software Design Engineer for companies such as Pixera, 3 Com and Mercantec.

ANNA HERNANDEZ
Anna Hernandez is currently working for the University of Texas, Brownsville as an IT specialist. She obtained her Bachelor degree in Computer Science from the University of Texas, Brownsville where she worked under the supervision of Dr. Quweider for her senior project.