DIGITAL CIRCUIT DESIGN MINI CONFERENCE

ELE202 Digital Circuit Design Laboratory

Cherry Auditorium, Kirk Engineering Building, December 5, 7 & 10, 2001
Department of Electrical & Computer Engineering, University of Rhode Island

Session I. Wednesday (Dec. 5) 8:00-8:50 am

1. 8:00 - 8:07 Don't Blink the Game. Pascale Delaunay, Shanika Doctor
2. 8:07 - 8:14 Electronic Safe Lock. Va Xiong
3. 8:14 - 8:21
4. 8:21 - 8:28 Up/Down Stop Watch. Kai-Chieh Yang
5. 8:28 - 8:35 Combinational Lock. Jeffrey Frost, Dan Kelsay
6. 8:35 - 8:42
7. 8:42 - 8:49

Session II. Wednesday (Dec. 5) 9:00-9:50 am

22. 9:00 - 9:07 Traffic Hazard Control Signal. Cynthia Calderwood, Tristan Trottier
23. 9:07 - 9:14 Digital Calendar. Eric G. Champion, Frank L. Maker III
24. 9:14 - 9:21 Light Up Children's Sneakers. Christopher Shaffer
25. 9:21 - 9:28 Dancing Lights. Thomas Nimmo, Joshua Ford
26. 9:28 - 9:35 Nurse Call Light. Kaylen Haley, Amy Wright
27. 9:35 - 9:42 Traffic Light Controller. Peter DeBellis, Martin Steele
28. 9:42 - 9:49 Lock with 10 Second Countdown Shawn Martin, Chris Leahy, Rob Mac Fadyen

Session III. Friday (Dec. 7) 8:00-8:50 am

8. 8:00 - 8:07 Electronic Locking System. Kyle Bissett
9. 8:07 - 8:14 Digital Dice. John Koch, Luis Torrico, John Coughlin
10. 8:14 - 8:21
11. 8:21 - 8:28
12. 8:28 - 8:35
13. 8:35 - 8:42 Traffic Light. Adam Cabral, Thomas Cavaliere
14. 8:42 - 8:49 Gameshow Buzzer. Jason Gail, Chirag Patel

Session IV. Friday (Dec. 7) 9:00-9:50 am

29. 9:00 - 9:07 One Click Mouse. Anna Wagner, Tim Swaby
30. 9:07 - 9:14 Timed Beeper. Alioune Fall, Josh Lentini, Teven Tran
31. 9:14 - 9:21 Electronic Lock Matt. McKeever, Neil Young
32. 9:21 - 9:28 Christmas Lights. Lee Ann Krafton, Clark Baird
33. 9:28 - 9:35 Timer With An Alarm. Jeremy Fournier, Lenny Vargas
34. 9:35 - 9:42 Reprogrammable Electronic Lock. Andrew Bonin
35. 9:42 - 9:49 Electronic Lock. Jeremy Russell

Session V. Monday (Dec. 10) 8:00-8:50 am

15. 8:00 - 8:07 Eat At Joe's. Elana Mayer, Nicole Baugh
16. 8:07 - 8:14 Victor Montero, Enrique Blanco
17. 8:14 - 8:21
18. 8:21 - 8:28 Decorative Lights with Four Functions. Matthew Girouard, Matthew Bilodeau
19. 8:28 - 8:35 Electronic Combination Lock. Eddie Ramirez, Daniel Schaefer
20. 8:35 - 8:42 Digital lock with Alarm. Meghan Collins, Curtis Horne
21. 8:42 - 8:49 Overriding a Lock. Manuel Galvao, Adriel Lopes

Session VI. Monday (Dec. 10) 9:00-9:50 am

36. 9:00 - 9:07 Arcade Game. Ellaine Abueg, Michel Beliard, Chuen-Song
37. 9:07 - 9:14 Stop Watch. Eric Lefurge
38. 9:14 - 9:21 The SmartSnooze Alarm Clock. Eugene Chabot, Michael Rossoni
39. 9:21 - 9:28 Dice slot machine. Francisco Faria, Rasiel Mejia, Victor Montero
40. 9:28 - 9:35 The Lights at Butterfield Brian. Cookson, David Gibb, Patrick Martin
41. 9:35 - 9:42
42. 9:42 - 9:49 Number Guessing Game. Michael Kuhn, Gregory Herring

1

Don't Blink the Game

Pascale Delaunay, Shanika Doctor. Department of Electrical and Computer Engineering, University of Rhode Island

The group is designing a two player reflex game. There is a main light which is green , this is the light the players will react to. There are two other LED's which light up to indicate which player got the point. We will be using a binary up/down counter and a 14-bit binary counter to implement the random light pulses. We will also be using a dual D Flip-Flop, and a few AND gate to implement the push controls and the light control. We still have to work out disabling the other button once one player pushes theirs. The game is designed so that the buttons are active only when the light is on. Also if both players press the button at the same time both lights light up indicating a tie.

2

Electronic Safe Lock

Va Xiong, Department of Electrical and Computer Engineering, University of Rhode Island

This is a simple implementation of a safe that is controlled by electronic code. The input is a 4-bit word that is stored as the pass-code into 4 Flip-flops. To open the lock, the correct pass-code must be entered which is stored into another 4 Flip-flops. The Pass-code and the entered code are each displayed with 4 LED’s. The codes are then compared and the status of the lock is displayed by a LED which stays ON if locked, OFF if unlocked, and blinking if reset. This concept can be further extended to a more complex circuit, which may have a larger word for pass-code for greater security but the logic remains the same.

4

Up/Down Stop Watch

Kai-Chieh Yang. Department of Electrical Engineering, University of Rhode Island, Kingston, RI

This project is a design of a stopwatch using single button as inputs and a seven-segment LED display as an output. This stopwatch differs from a normal stop watch by its countdown feature. Like a general stop watch, when the push-button is pressed, it starts to count. After the button is released and pressed again, the stopwatch stops. In this state, if we press the button and hold it for more than 2 seconds, we set the stopwatch to countdown mode and reset it at the same time. Otherwise, we only reset the stopwatch. The “long” input (press and hold for more than 2 seconds)is ineffective while the stopwatch is running or has been reset. The designing process of this project is quite systematic. First, draw a simple block diagram to outline the circuit. Second, use state diagram and state table to determine what the circuit should do at certain condition. Then use K-map to realize the details of the circuit. Basically, the circuit contains a clock generator, a push-button, a BCD up/down counter, two binary counters (one for detecting the “long” signal, the other for BCD counter), a BCD to seven-segment driver, a even-segment LED display, and a system controller(the main circuit).

5

Combinational Lock

Jeffrey Frost Dan Kelsay. Department of Electrical Engineering, University of Rhode Island, Kingston, RI

To enable the lock the "armed" state binary sequence 1011 is entered into the system by a switch. Each bit in the code is entered in sync with the clock. An LED is connected to the clock so the user will know when the clock is changing. The same code is used again to disarm the lock from its armed state. However if a mistake is made in entering the sequence from either the armed or disarmed state then the system resets itself back to it's original state: either armed or disarmed. Another LED is connected to the outputs of the flip-flops to light up when the lock is armed. Three J-K flip-flops where used along with a series of 2 input AND, OR and 3 input AND gates, inverters and buffers, and resistors. When armed and an outside source disturbs the lock then a buzzer will go off.

8

Electronic Locking System

Kyle Bissett. Department of Electrical Engineering, University of Rhode Island, Kingston, RI

This project is driven by 2 D Flip-Flops to determine state. A second set of flip-flops hold the current combination. The purpose of this project is to be able to take input from the DIP switches, and determine the next state from the current state and whether or not the input matches what is currently in memory. If the state is ready, the input becomes the new combination, and the lock is undone. If the state is unlocked, the state will go to ready if a match is made and locked if there is not. If the state is locked, the state will go to unlocked if a math is made and remain locked otherwise. The status of locked and unlocked is displayed by LEDs.

9

Digital Dice

John Koch, Luis Torrico, John Coughlin.
Department of Electrical Engineering, University of Rhode Island, Kingston, RI

For our final design project we created a six-faced digital die with the use of 7 LED lights. The design includes the use of 3 MC14027B JK-Flip Flops to generate random number, the outputs from the Flip Flops, 3 MC14071B OR gates, and 2 MC14081B AND gates to assign the number to the corresponding LED. When a button is pushed the circuit will choose a number and the correct LED's will light up on the die face indicating the selected number. The lights are arranged in the typical die fashion with three LED's on the left, three on the right and one in the middle.

13

Traffic Light

Adam Cabral and Thomas Cavaliere.
Department of Electrical and Computer Engineering

In our ELE 202 design project, we designed a circuit that controls the traffic lights at a four way intersection. We wanted this light to simulate a stoplight where a side road crosses a main road. The main road will have a green light until a vehicle pulls up to the intersection on the side road. When this occurs, a sensor is set off enabling a counter. After a delay, the light changes. The lights on the main road will turn red and the side road will turn green. At this point, another counter starts. After the delay the lights change back to their default position. This happens whether or not the sensor is still depressed (meaning cars are still on the side road). We designed the circuit using one MC14520 which is a dual up counter. We used a pushbutton for our sensor and a JK flip flop. Our circuit performs as anticipated.

14

Gameshow Buzzer

Jason Gail, Chirag Patel. Dept. of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

My group is designing a gameshow buzzer. It has 3 switches all hooked up to one clock. When a question is asked, the clock is started at set amount of time. if the time runs out(counting down), then a light will be triggered, meaning that everyone has run out of time. If a one of the three switches (buzzers) are pressed, then the timer stops, and a light corresponding to that buzzer is then lit. The timer consists of a two digit linked counter with a variable speed, and a selectable starting time. The clock will be stopped using a nor network, and a series of and gates to be sure that only one buzzer can be set off at a time.

15

Eat At Joe's

Elana Mayer (EE), Nicole Baugh (BE/IEP). Dept. of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

Our project was designed using a state machine of simple logic. The point of our project is to have letters light up on the 7-segment LED displays: J, O, E, ', S. The letter will scroll down the five LED displays, one letter at a time. What we;d like to prove is that a device like this can run off of just simple logic instead of a RAM or memory chip. In our design we are using three JK Flip-Flops, a good amount of AND and OR gates, and a timer for the JK-FF clocks. So far we have run in to a few snags but we are working out the last few bugs. We have seen that our letters will scroll however, they stop at the second to last LED. a few of the LED segment also do not give us the right letter. What I realized our problem was was that we might have burned out he display. So today i grabbed ten 470 ohm resistors and five new displays. I will work on it over the weekend and have the bugs worked out too.

18

Decorative Lights With Four Functions

Matthew Girouard, Matthew Bilodeau. Department of Electrical and Computer Engineering, University of Rhode Island, Kingston, Rhode Island

We have developed a four-functioned decorative light set. Each time the user presses a debounced push button, which is simulated by a DIP switch, the lights will switch to the next function, eventually looping around to the first function. The functions are as follows:

1. All of the lights are on (using 4 LEDs, 1111, 1111, 1111, 1111)
2. All of the lights blinking in unison (1111, 0000, 1111, 0000)
3. Alternating every other light on (0101, 1010, 0101, 1010)
4. Chasing lights

We have implemented this circuit by using 3 AND gates, 2 J-K FF, 4 OR gates, 1 2 to 4 decoder (4 AND gates) and 1 inverter. So far the circuit is fully implemented and operational with the exception of the chasing lights. We expect to have the chasing lights fully operational for the presentation.

19

Electronic Combination Lock

Eddie Ramirez, Daniel Schaefer. Department of Electrical Engineering, University of Rhode Island

This project is designed to have the user enter in a four input combination twice, whereas the second is dependent upon the first for the lock to open. After entering the first four number combination and pressing the debounce switch, an LED will blink to signify that the combination was entered correctly. Once the second combination is entered and the denounce switch is pressed again, the LED will permanently illuminate as long as the second combination was entered correctly. The eight DIP switches are wired to a multiplexer, which compares the first four numbers within the lock to the first set of numbers entered by the user using four separate exclusive OR gates. The user enters the combination on four DIP switches and presses the debounce switch to enter the second four numbers. The four outputs of the exclusive OR gate are connected to a four input AND gate. The output yielded from the AND gate is then entered in a D Flip Flop, so as to avoid the user from entering a different set of numbers without pressing the debounce switch. This is then entered into a counter, and finally to the LED. The counter is used to cause the LED to blink after the first correct entry. The lock closes after the user presses the debounce switch a third time.

20

Digital lock with Alarm

Meghan Collins, Curtis Horne, Univerisity of Rhode Island Dept. of Electrical &Computer Engineering Kingston, Rhode Island

When the circuit is initially activated, the red LED lights up indicating the lock is locked. The lock combination is already pre-set in the lock. When the right combination is put in twice the green LED flashes and the lock is ready to accept a new combination. In order to lock it, a wrong combination has to be put in. While the lock is locked if a wrong combination is entered, the alarm will sound. Within the circuit the user is at first enabling a four bit register and storing a four bit number in it. When the user want to enter a new combination there is only one set of switches and an enter button. To lock it a wrong combination is enter. To send a number to be compared an enter button has to be pressed. If a wrong combination is entered an alarm will sound. To disable the alarm the lock has to be unlock.

21

Overriding a Lock

Manuel Galvao, Adriel Lopes. Department of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

The purpose of our project, *Overriding a Lock,* is basically a program that displays the proper combination for a four-bit lock. The idea came into mind; if a certain individual looses or fails to remember the combination of the lock he/she can use this electrical hardware to implement the proper lock combination. Surely, a thief may also find this electrical hardware very useful when it comes to their work related tasks. This overriding lock uses a 2-input AND gate (4081), a dual up counter (4520), 4-bit magnitude comparator (4585), and a latch (4508). The overriding lock works by using a counter to find a certain combination. The combination is stored in the latch. Once we turn on the power, counter two begins to loop. By holding the push button counter two stores the binary combination to the latch. As it stores the combination, counter one automatically begins to loop. This overriding lock then compares the counter one loop and the latch in order to illuminate the LED logic indicator in which in this case relates to the opening of the lock. Moreover, once the lock is open we can then change the lock combination and repeat the process if one looses or fails to remember the new combination.

22

Traffic Hazard Control Signal

Cynthia Calderwood, Tristan Trottier. Dept. of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

For this project, we designed a traffic light that would be used on the side of the road in order to direct traffic around a construction sign or to warn drivers to be cautious with nearby roadwork. The sign consists of a series of LED's in the form of a two-headed arrow. There are three inputs to the circuit including a left signal, a right signal, and a caution signal. Each input is controlled through a DIP switch. When the left signal is turned on, the lights will create a cycle that will sequentially turn on in order to form the left side of the arrow. The right side of the arrow will light sequentially when the right signal is on. For the caution signal, the center lights with no arrowheads will flash in time with the clock. The circuit was designed by using D-flip-flops, two and three input AND gates and three input OR gates.

23

Digital Calendar

Eric G. Champion and Frank L. Maker III. Dept. of Electrical and Computer Engineering, University of Rhode Island, Kingston, Rhode Island

The group is designing a digital calendar, which will consist of three separate counting devices. The first of these counting devices will receive a clock signal and will increment a Seven Segment display by one each time. There will be two 7-segment displays for the days as double digits will be required. When the first display reaches 9 the next one will be incremented as the first one continues through the cycle back to zero. All of the counting devices will work in a similar fashion depending on how many digits are to be displayed. There will be 2 seven segment displays for the days, 2 for the months, and 2 for the last two digits in the year. A standard up-counting operation will not suffice in this instance because of varying amount of days in the separate months. To overcome this obstacle an output decoder had to be developed in order to determine which month the calendar was in, and henceforth it would count the correct number of days before incrementing the displays to the next month. Once the number of years equals 13 the year counter will be incremented by one, and the entire process will start counting from January again. The counters increment on what the day can not be (i.e. if there is 31 days in the month the counter will increment when the days equals 32).

24

Light Up Children's Sneakers

Christopher Shaffer. Department of Electrical and Computer Engineering. University of Rhode Island Kingston, RI

The sneakers lights consist of four LED's that blink in two different patterns. The patterns are selectable through the use of a switch (y), in this project simulated using a DIP switch. All the lights are off unless activated by a pressure sensor in the sneaker, simulated using a de-bounced push button. The first sequence of flashes follows this pattern: L0 off, L1 on, L2 on, L3 off; then L0 on, L1 off, L2 off, L3 on; and L0 off, L1 on, L2 on, and L3 off. The second sequence follows this pattern: the two right most lights on, then the two left most lights on, then all lights on. For the design of the circuit I choose to use 2 D Flip-Flops to control the four states of the lights. The design also includes a next state decoder, and the (y) switch was connected directly into the output decoder which controls the lights. There are a total of eight chips used to implement this design, one D Flip-Flop chip; one Inverter chip; two, two input AND gate chips; two three Input AND gate chips; one two input OR gate chip; and one three input OR gate chip.

25

Dancing Lights

Thomas Nimmo, Joshua Ford. Dept. of Electrical and Computer Engineering, University of Rhode Island, Kingston, Rhode Island

This project's goal is to make a state machine to control a 5x5 array of LED's with 4 states. Each state will show a stick figure in different stages of dance. Our first idea for implementation was to design a different circuit for each row. But we decided that certain sections could be hardwired saving time and resources. If we are successful with our initial design we plan to add a device to clock the circuit with an external sound input.

26

Nurse Call Light

Kaylen Haley, Amy Wright. Department of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

Our Nurse Call Light is designed with two push-button switches and an LED. The patient, to signal either an emergency or an assistance situation, uses one button and the other is the Reset button used by the nurse. A blinking LED represents an emergency situation and a constantly lit LED represents the need for assistance. When the patient pushes the button, the call light is sent into an emergency state and a second button push within 2 seconds sends it into the assistance state. The only way to turn the call light off is if a nurse pushes the Reset. The first button push sends an enable signal, which is ANDed with a 1Hz count from a 4-bit counter. The result is wired to the clock input of a second 4-bit counter and after a 2-second count the patient push-button is disabled. If the button was pushed a second time before the 2-second count was over, the emergency light turns off and the LED stays lit. Otherwise the emergency signal remains on.

27

Traffic Light Controller

Peter DeBellis, Martin Steele. Department of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

Our idea is to design a traffic light controller. The situation involves the intersection of a primary and secondary road. The "normal" or resting state of this system will be a green light for the primary road, and a red light for the secondary. Pedestrian crossing buttons will be located at all 4 corners of the intersection, in addition to pressure sensors on the secondary road. A signal from a car or pedestrian will cause the primary light to cycle to yellow, then red, allowing flow onto the main road for 16 seconds. The system will then begin its return to the resting state by having the secondary light cycle to yellow, then red, where it will remain until another signal for access is input. The system will consist of 2 D-FF's, along with some counters to allow pausing in states. Also, a push button will be used to simulate pressure sensors and pedestrian buttons.

28

Lock with 10 Second Countdown

Shawn Martin, Chris Leahy, Rob Mac Fadyen Department of Electrical Engineering, University of Rhode Island

Our lock design is a simple 4 bit combination lock. When the device powers up a combination must be set. When the enter but is pressed the user has 10 seconds, displayed by a countdown of lights, to enter the correct combination and press enter. If the attempt is successful, the user is greeted by green lights, however a failure in combination or time limit results in red lights. With a successful attempt, the user may enter in a new combination.

29

One Click Mouse

Anna Wagner, Tim Swaby. Dept. of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

The group designed a single click mouse for a PC. It was designed to be used by people with severe disabilities who have little or no use of their hands, such as paraplegia and quadriplegia. These people are therefore unable to use a standard PC mouse, but with our project they can. Our project is made up of two basic parts: a standard PC mouse and the control device which displays the options to the user, takes their input, and relays it to the mouse. Our control device has six LEDs which each represent a certain function of a mouse (up, right, down, left, left click, and right click.) Using flipflops, the circuit circles through the LEDs at a reasonable rate. When the user sees the function they desire to use, they push the button and the mouse performs that action until the button is released. In order to do this, the input (button) is XORed with the power running to the enable line of the clock. This clock runs the LEDs, so when the button is pushed and held, resulting in a logic 1, the XOR outputs a logic 0, stopping the enable signal and stopping the clock. When the clock stops, the LEDs are frozen, and only the LED representing the desired function will be on. All the directions (up, down, etc.) will be ANDed with the button input running to the mouse, so they will only be implemented (move the mouse) if both are logic 1 (button is pushed and LED is on). This creates a way for people with disabilities to use a computer like everyone else with, literally, the push of a button. And although the mouse can only move in one direction at a time, it gives some freedom to people who otherwise wouldn't be able to use a mouse at all.

30

Timed Beeper

Alioune Fall, Josh Lentini, Teven Tran
Department of Electrical and Computer Engineering, University of Rhode Island

For this project, we decide to design a timed Beeper, which can count up to 15 seconds, 30 seconds, 1 minute, and 2 minutes, with an interval of 4 seconds after the preset time is elapsed. We will be using 2 to 4 And gates and 8 Bit-static shift register or 14 binary counter and an oscillator. The reset button can force the oscillator to go very fast, it can take few seconds to terminate the counting if the highest time's duration is chosen and the pushbutton will operate when the circuit starts. Pushing the reset button, it will oscillate at the frequency fixed and the polyester capacitor and the red LED will be driven by the And gates and flash at the same oscillator frequency. The reset and start buttons are the main goal of this project.

31

Electronic Lock

Matt McKeever, Neil Young. Department of Electrical and Computer Engineering, University of Rhode Island

An electronic lock is used everyday, so there is always room for improvement. Therefore the group decided they would develop an electronic lock This lock has many different and innovative features. These features include a combination lock that includes the letters A, B, C, and D, to give the user 24 different combinations. Another feature is a timer that gives the user a certain amount of time to input the correct code. If the user fails to input the correct combination, the lock automatically resets the inputs to zero. The circuit is built with nine 4175 flip-flops, one 4011 NAND gate, 6 4081 AND gates, and four 4071 OR gates. The uses for this circuit in the real world include locks for homes and businesses, and locks on steering wheels to prevent inebriated drivers from driving home.

32

Christmas Lights

Lee Ann Krafton, Clark Baird. Department of Electrical and Computer Engineering, University of Rhode Island

Our project was to develop a model of Christmas lights that have four patterns. The lights are either all on, all off, flashing, or alternating. Our design is complete on paper and uses 5 2-input AND gates, 2 3-input AND gates, 4 D-Flip Flops, 1 Exclusive OR gate and 3 OR-gates. The sequencing runs off the clock pulses. The basic circuit is designed with the clock as an input to the flip flops and one x-input. There are two output and then we just wired the other LED's in series. As of now our design is working on Xlinix but we have not completed the actual protoboard at this time.

33

Timer With An Alarm

Jeremy Fournier and Lenny Vargas. Dept. of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

This is going to be a timer that we can program to start at a specific time using the DIP switches. Once the timer counts down to 0, an alarm will go off. We will be using the concept of Lab #6 to complete the timer. We will be using two proto-boards and linking them to the main proto-board in the lab to use the alarm and DIP switches. We will be using two seven segment displays to represent seconds and tens of seconds. We will be using two of each CMOS chip: 4551, 4511, and 4510.

34

Reprogrammable Electronic Lock

Andrew Bonin. Dept. of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

The purpose of this project was to design a 4 bit electronic lock with the ability to change combination codes. The state of the lock is represented by four LED's, three red and one green. The locked state is represented by one red LED. The user has two chances to enter the correct code with the use of the DIP switches. If an incorrect code is entered a second red LED will light up and then a third. If the correct code is entered the green LED will light up. Then the user will have the option to change the security code. If they wish to do so, they simply need to set the DIP switches to the desired code and press the second debounced button. The code is then stored in a series of D FF's. The state machine is comprised of two J/K FF's, AND gates, OR gates, and an inverter. In order to compare the code stored in the D FF's to the code entered by the user I used a 4 bit magnitude comparator.

35

Electronic Lock

Jeremy Russell. Department of Electrical and Computer Engineering University of Rhode Island, Kingston RI

I will be presenting an electronic lock. It will have two LED's and a will take a for bit word for its input. The inputs will come from a DIP switch. When the system is locked both LED's light up. To unlock the lock you must go to waiting state. If you input the correct combo here the lock will then unlock. When it is unlocked both lights are on. If you input the wrong combo in the unlocked state then the system will lock. If you input the correct combination in the unlocked state then you will return to the waiting state, where you can then change the combination. To input the combination I used 4 DIP switches into a latch to hold the combination. To transition from the states – unlocked, locked, and waiting – I used two D-Flip Flops, using a debounced push button to act as the clock. To compare the combo held in the latch to the inputs from the DIP switch I used a comparator chip (MC14585B). I learned that you should never over think the problem. It can make you very confused.

36

Arcade Game

Ellaine Abueg, Michel Beliard, Chuen-Song Chen.
Dept. of Electrical and Computer Engineering,
University of Rhode Island, Kingston, RI

Our group is designing an arcade game in which a ring of LEDs light up one by one, in accordance to the clock cycle. The purpose of this game is to stop the light, by pressing a button, on the red LED. The closer to the red LED that the light stops, the more points are earned. It consists of three turns in which the level of difficulty increases when the maximum points are earned, in this case 3. The game's level of difficulty increases by increasing the rate of the clock cycle. The earned points will be accumulated and shown in a 7-segment LED display. In order to implement each portion of the project, we used the following chips: For the LEDs to light up one by one, we used a decade counter which transferred a "high" to each of the LEDs connected to its outputs. For each LED worth points, we connected it to an 8-bit priority encoder. For the red LED, we connected it directly to "3" on the encoder. The two LEDs on each side of the red LED are worth 2 points, therefore they were connected to an OR gate and then to the corresponding "2" on the encoder. Finally, the third LED on each side of the red LED are worth 1 point. They too, were connected to an OR gate, then to "1" on the encoder. All of the other green LEDs are worth 0 points so they weren't connected to anything. The 8-bit priority encoder was required for the decimal value of points to be "translated" into binary, which could then be read by the D-Flip flop. This D-Flip flop was used as a register to hold the totals for each previous round(s). This D-FF was connected to a full-adder that added all of the points gained so far. The full-adder was then connected to a BCD to 7-segment display chip that was connected to the 7-segment LED. This is used to display total points earned so far.

37

Stop Watch

Eric Lefurge. Dept. of Electrical Engineering, University of Rhode Island, Kingston RI

I have decided to design a digital circuit for a stop watch that can count up to 10 seconds. The watch is capable of pausing its count, resetting, and counting backwards, while displaying its count on a seven segment LED display output. Using a debounce switch to input a signal the owner has a choice of two kinds of switch control – a short depression and a long depression. Using a short depression which is the pushing and releasing of the debounce button within one second. This choice allows the owner to start, stop, and reset the counter. While in pause mode, a long depression in pause mode will cause the counter to toggle between counting forwards and backwards.

38

The SmartSnooze Alarm Clock

Eugene Chabot, Michael Rossoni. Department of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

Have you ever waken up late for class, or just slept right through due to laziness? This alarm clock is made for the slacker in you! It contains a unique SNOOZE button which will decrease the default timer by one minute every time it is pressed. We designed both the clock and alarm. Our clock contains a series of 4-bit up counters to keep the time, and a D flip-flop to keep track of the AM/PM state. To keep our design simple, we used blocks of DIP switches to set the alarm. The DIP switch settings will be compared with the actual time every minute, and will enable the audible alarm when the settings match the actual time. The alarm will go off until the SNOOZE button is pressed. The SNOOZE button is also connected to a 4-bit down counter, which will decrement by one minute every time the SNOOZE button is pressed. Once the counter reaches zero, the SNOOZE button becomes disabled and the audible alarm will not stop until the user gets his/her butt out of bed and switches the alarm off. This design approach is primitive in the sense that we could have designed (programmed) the whole device on one or two PROM ICs. It would be more cost-efficient, time-efficient, and spatial-efficient to use the latter, as opposed to many ICs and wires.

39

Dice slot machine

Francisco Faria, Rasiel Mejia, Victor Montero. Dept of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

Our project is a simulation of a slot machine that will use dice as the display to determine if you win or lose. If the numbers on all three dice match, you win. If you get two outer die to match, whatever is in the middle, will determine how much you win (the higher the number the more you win). If you get any other combinations you lose. The main objective is to try and get all three die to match. The game is controlled by a de-bounced switch. To activate the game, you hit the switch once. When you want the die to stop scrolling, you just hit the switch again. This is achieved by using an XOR and a D-FF to control the state. The output of the D-FF is ANDed with a clock of a different frequency for each of the three counters, to generate some randomness. To keep the counters counting only from one to six, we use simple combinational logic. The outputs of the counters then go to the led decoders. The led decoders determine which LED's on each die should light up. This is also done with simple combinational logic. Finally, we have three LED's that will determine if you "WIN", "LOSE", or "TRY AGAIN". This will also be done using combinational logic.

40

The Lights at Butterfield

Brian Cookson, David Gibb, Patrick Martin. Dept. of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

If you are a part of the URI Community, you love Butterfield food. But how do they heat the food they serve? For most stations, they use hot lights to warm the food. Our project is the timing unit of when the hot lights go on and off. The circuit requires 11 chips, two seven-segment displays, LEDs, a DIP Switch and an SPDT Switch. The circuit starts out counting down from 60 with the hot-lights, (LEDs), on, and the two seven-segment displays counting down. When it reaches 0, the hot-lights switch off, and it counts down from an inputted variable time, (inputted by the DIP Switch). Two Multiplexers feed the count and therefore manage the two times. When this count reaches 0, the cycle starts again. The reason why there is an off time is so that one does not burn the food under the lights. There is also a kill switch, which stops the cycle automatically, shuts the hot-lights off, and clears the clock, resetting it to 60. Once the kill switch is off, the circuit starts again. A significant finding was that this project could have been tremendously simpler, if there was a double seven-segment display chip available. Because there wasn't one, this project's most complicated aspect was the design of the 2-digit clock. For reference, our group number is #40.

41

Number Guessing Game

Michael Kuhn, Gregory Herring. Dept. of Electrical and Computer Engineering, University of Rhode Island, Kingston, RI

Our project will play like a number guessing game, with choices of 0 through 3. We will have an incrementing counter cycling through 0 through 3 with each positive turn of a clock function, and a switch attached that when pressed stores the most recent value to be the one that will be guessed. After the chooser button is pressed, the game-player has a guessing button that they can press 0, 1, 2, or 3 times for their respective guess. We can then have a web of combinational logic to check to see if their answer is identical to the sotred value, and if it is it, turns on a "right" light, and if wrong, a "wrong" light. We have already analyzed this challenge and come up with a plan and a bare sketch for action. For the chooser system, all we need are 2 j/k flip flop gates and a square function frequency (clock) to increment 0 through 3 and over again, until the logic "1" input to the first j/k is disconnected, which will act as the chooser button. To store the most recent value, we only need 2 D flip flops, one for each of the j/k, and an inverted connection from the chooser button to the clock terminal of both flip flops under the guessing button, where the guessing button will act as the clock, changing the incrementing value of the guess with each push, 0 through 3 and back again. When the player has hit a value they want to guess, they can press the finish button, which allows their value to be compared to the "random" number, and their succes documented.