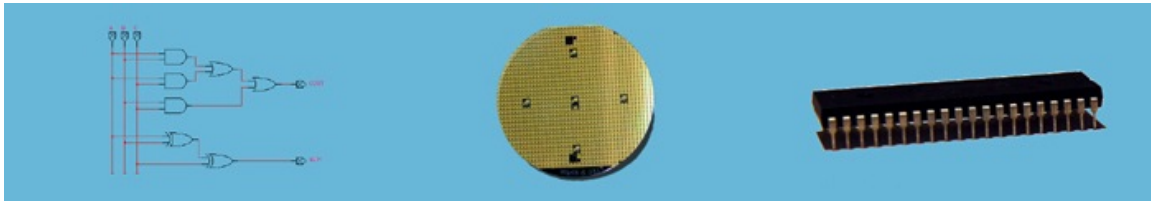




CE1900 WEEK 10 LABORATORY EXERCISES



LABORATORY SUMMARY

The quarter is rapidly coming to a close. Students have learned a significant amount about combinational digital logic design. A knowledge base of digital signals, canonical equations, basic Boolean algebra, logic gates, K-map reduction techniques, binary arithmetic on signed and unsigned binary numbers, adders, subtractors, ALUs, encoders, decoders, and multiplexers has been studied this quarter. Additionally, basic VHDL dataflow and structural modeling has been studied.

This week, students continue to practice design techniques by completing the design of a car alarm.

Students will complete multiple design tasks:

- **Design** and **simulate** a VHDL dataflow description of the alarm door priority encoder.
- **Design** and **simulate** a VHDL dataflow description of the alarm controller.
- **Design** and **simulate** a VHDL dataflow description of the alarm 7-segment decoder.
- **Design** and **simulate** a schematic blueprint of the alarm system top-level entity.
- **Implement** the **car alarm system** using the DE1 laboratory board and a sonic-alert.
- **Test** the **car alarm system** in the laboratory.

The laboratory exercises reinforce these CE1900 learning objectives:

- **Use** VHDL entry to design and simulate arithmetic circuits.
- **Implement** completed designs in programmable logic devices.

Print this document and **bring** it to lab with the preliminary activities completed in the spaces provided.

Hint: save your color print cartridge by selecting “black-and-white” printing from your print options.

REMEMBER

Complete all homework and preliminary lab exercises *before you come to lab*.

Review the in-class laboratory exercises to preview the work you will do in the lab.



CE1900 WEEK 10 LABORATORY EXERCISES

PRELIMINARY LAB EXERCISES

A simple car alarm system uses a set of contact switches mounted on the hinged surfaces of the automobile. The contact switch produces 0V (logic 0) when the hinged surface is closed and 5V (logic 1) when the hinged surface is open. Table 1 maps each contact switch to a hinged surface.

TABLE 1: CONTACT SWITCHES BY ID NUMBER

CONTACT SWITCH	HINGED SURFACE
1	trunk
2	gas tank door
3	right rear door
4	right front door
5	engine compartment hood
6	left front door
7	left rear door

The car alarm system uses two 7-segment displays to provide user information. When disabled, the alarm system turns all the 7-segments off resulting in two blank digits. When enabled, the alarm system displays AA – for alarm armed – when no hinged surfaces are open. If a hinged surface is opened when the alarm is enabled, the display shows the characters given in Table 2. Priority encoding is used with switch 7 being the highest priority.

TABLE 2: DISPLAY WHEN ALARM ENABLED

CONTACT SWITCH OPEN (1)	LEFT DIGIT	RIGHT DIGIT
1	t	r
2	g	t
3	r	r
4	r	F
5	e	c
6	L	F
7	L	r

The car alarm system uses a sonic-alert actuator to produce a high-decibel sound into the environment if the alarm is enabled and any contact switch is open. The hope, of course, is that a burglar will flee because they risk discovery by nearby people reacting to the loud alarm. (Student thought question: This premise worked well two decades ago. Today, do most people react to car alarms? If not, can you think of better ways for car alarms to help prevent theft?)

The car alarm can be designed in many ways. Design is flexible. Engineers learn basic knowledge and design skills that they then apply creatively to solve the problem. Creativity will lead to many different solutions as different engineers apply the art of engineering with the science of engineering. One way to design the alarm requires three components: an 8-to-3 priority encoder, an alarm control unit, and a decoder for characters on the 7-segment display. The priority encoder condenses the eight contact switch values into a three-bit binary number reporting the highest numbered switch that is open, the alarm controller makes decisions about what to display on the 7-segment displays because of the three bit binary number created by the encoder, and the 7-segment decoder draws the characters on the display. This type of design – dividing a complex task into subparts – is a common approach in engineering.

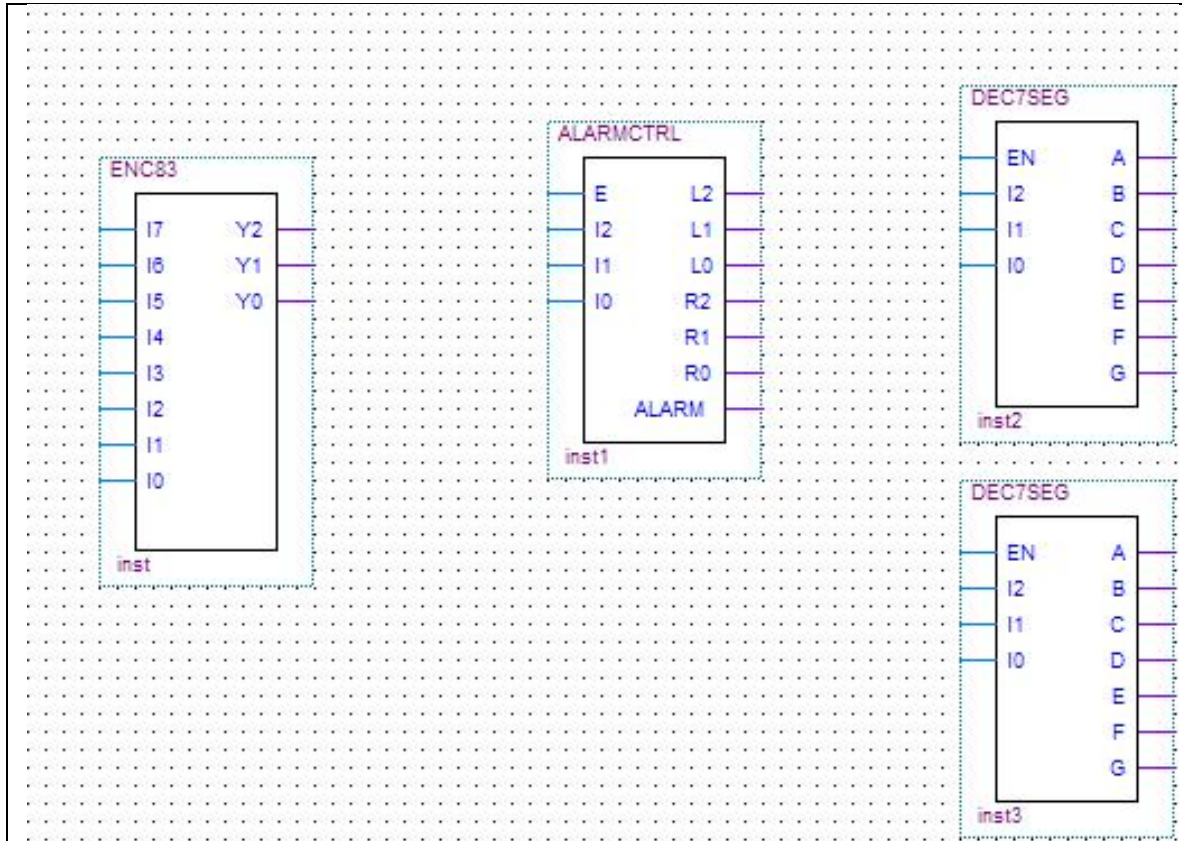


FIGURE 1: THE SUGGESTED TOP-LEVEL ENTITY STRUCTURE

Figure 1 suggests the structure of the top-level entity. The seven contact switches are inputs to the 8-to-3 priority encoder. Connect the switches in numerical order to the priority encoder inputs (i.e. contact switch 7 to input 7, contact switch 1 to input 1, etc.). **Note that the priority encoder shown in this diagram is missing the standard output that shows no energy on any input. You will likely want to add that signal.**

Design a priority encoder that reports the most significant switch that is active but also reports 000 when **no switch** is active (hint: see the highlighted line in the paragraph above). Leave the I0 input grounded in the top-level design since no contact switch 0 exists in the car. When the hinged surfaces are all closed the encoder will produce the number 000 on the outputs – representing the all-ok condition.

Design the alarm controller entity (ALARMCTRL) so that it uses the priority encoder output to make decisions about displayed characters. The enable signal should be attached to an input pin to selectively turn the alarm behavior on and off. The alarm controller outputs control the 7-segment decoders. The set labeled L2, L1, and L0 control the left digit of the alarm display while the set labeled R2, R1, and R0 control the right digit of the alarm display. **Hint:** how many different characters must be shown on the left and right displays? **Choose** your own unique binary encoding to represent the character desired on the display. The alarm controller output named ALARM should be attached to an output pin. This pin will be connected to the + signal of the sonic alert while the – signal of the sonic alert will be attached to the ground of the laboratory board during implementation of the circuit in the laboratory. Remember that the alarm should sound (logic 1) whenever any door is opened **and** the alarm is enabled.



CE1900 WEEK 10 LABORATORY EXERCISES

Design the seven-segment decoder (DEC7SEG) so that it displays the characters commanded by your ALARMCTRL module. **Turn on** the appropriate segments based on the desired character.

Complete a VHDL dataflow description for the priority encoder, the alarm controller, and the seven-segment decoder. **Complete** a schematic blueprint for the top-level design. **Don't forget** that the top-level design should have an *input* connected to the enable signals of the ALARMCTRL and DEC7SEG entities; and that the ALARMCTRL *output* called ALARM should be attached to an output pin to turn on and off the sonic alert. The DE1 and DE0 boards have an expansion header that provides general-purpose input/output pins, power, and ground. Refer to the appropriate pages of their manuals for information on this header.

LABORATORY EXERCISES

Each student must demonstrate the laboratory work by the end of the week 10 lab period.

1. **Take** the weekly CE1900 quiz given by your instructor at the start of the laboratory.
2. **Implement** the lab design in the DE1 or DE0 laboratory board.
3. **Test** the DE1 or DE0 implementation to **verify** operation.
4. **Demonstrate** to the instructor. The instructor will complete the signature block at the end of this document. The signature block serves as proof that you have completed the laboratory assignments. **Keep** this document in your binder for CE1900 in case you need to refer to it in future work or in case the instructor asks to see the signature block again at a later date.
5. **Estimate** the total amount of time you spent working on the pre-lab and lab

exercises. **Record** that time in minutes: _____

FOR INSTRUCTOR USE ONLY: DO NOT WRITE IN THIS TABLE

ITEM	COMPLETED	SCORE
Checking your progress		
Design exercises		
Demonstrates understanding of Quartus		
Lab demonstration		

Instructor Signature: _____