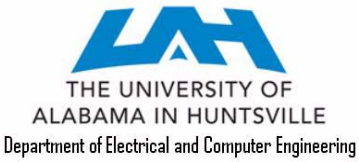


CPE 324 Advanced Logic Design Laboratory

Laboratory Assignment #8

Temperature Sensor

(7.5 % of Final Grade)

Purpose

The purpose of this laboratory project is to provide each student the opportunity to develop a component module and integrate it into a larger system-level design. The laboratory also gives the student the opportunity to create a logic circuit in Verilog HDL that will implement a specified multi-signal waveform. The project is to develop a temperature sensor interface that will continuously display the temperature in both the Fahrenheit and Celsius temperature scales using the seven segment LED displays that are present on the Altera DE2-115 Educational Board.

Background

Each student is to assume that he/she is a digital design engineer that is employed at a leading embedded systems corporation. The task at hand is to develop the interface electronics to implement a temperature sensor that is to continuously display the current temperature in Fahrenheit and Celsius. The interface electronics is to be designed using dedicated application specific logic as opposed to a microprocessor, and is to interface to two seven segment LEDs or LCDs. Because of mass procurements of parts by the previous administration (before the Chapter 11 reorganization effort began), the design must utilize a Dallas Semiconductor DS1620 integrated circuit. All temperature readings are to be accurate to plus or minus 0.5 degrees Celsius and the targeted rapid prototyping platform is to be the Altera DE2-115 Educational Board. The unit should reset the temperature acquisition process when one presses and releases the specified clear (CLR) button and should be able to be switched between Fahrenheit and Celsius by moving the specified toggle switch from one position to the other position.

Before his employment contract was terminated during the last round of lay-offs at the company, the person who was responsible for the temperature sensor design had created the high-level design that is shown in Figure 1. In this design, he incorporated the *BINTOHEX*, and the *TEMP2BCD* components that were present in the corporation's Intellectual Property Core (IPC) library. He also developed the clock division module, *CLOCK_DIVIDE*, to slow down the system clock so that it could be used in an indirect manner to clock the DS1620 IC. In this same high-level design this engineer had also identified the ALTERA Cyclone IV E inputs and output pins that are to be connected to the DS1620 pins, internal 50 MHZ clock, and seven segment display LEDs. It turns out that the only module that needs to be designed is the one that is labeled *DS1620_INTERFACE*. This module is clocked by the divided clock signal and drives the DS 1620's DQ bidirectional signal (using the built-in Altera tristate pin buffer), RST input, and the DS 1620 CLK input. The VHDL models for these IP Cores can be found on the CPE 324 Course Canvas Page.

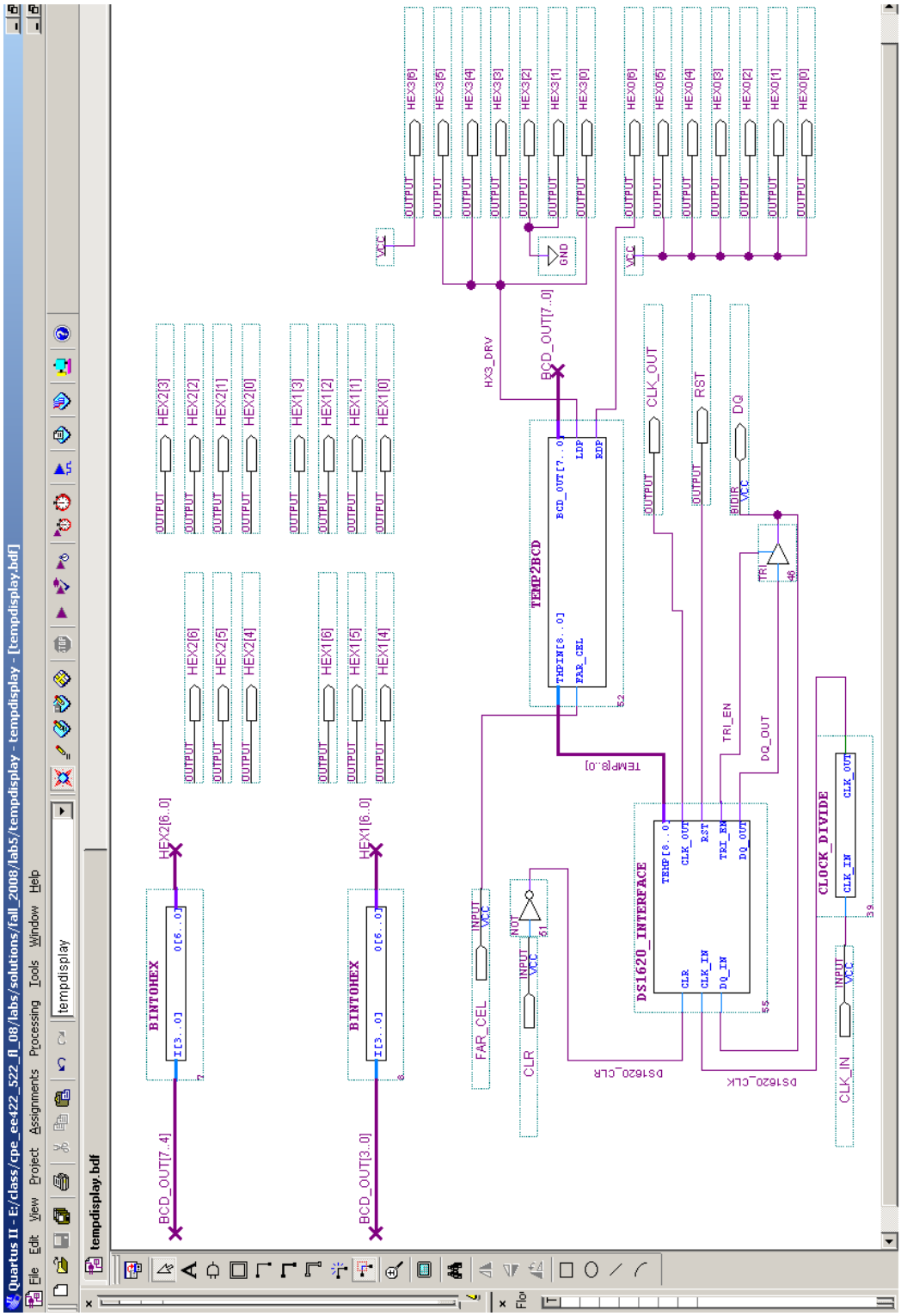


Figure 1. Proposed High-level Structural Representation of Temperature Sensor Design

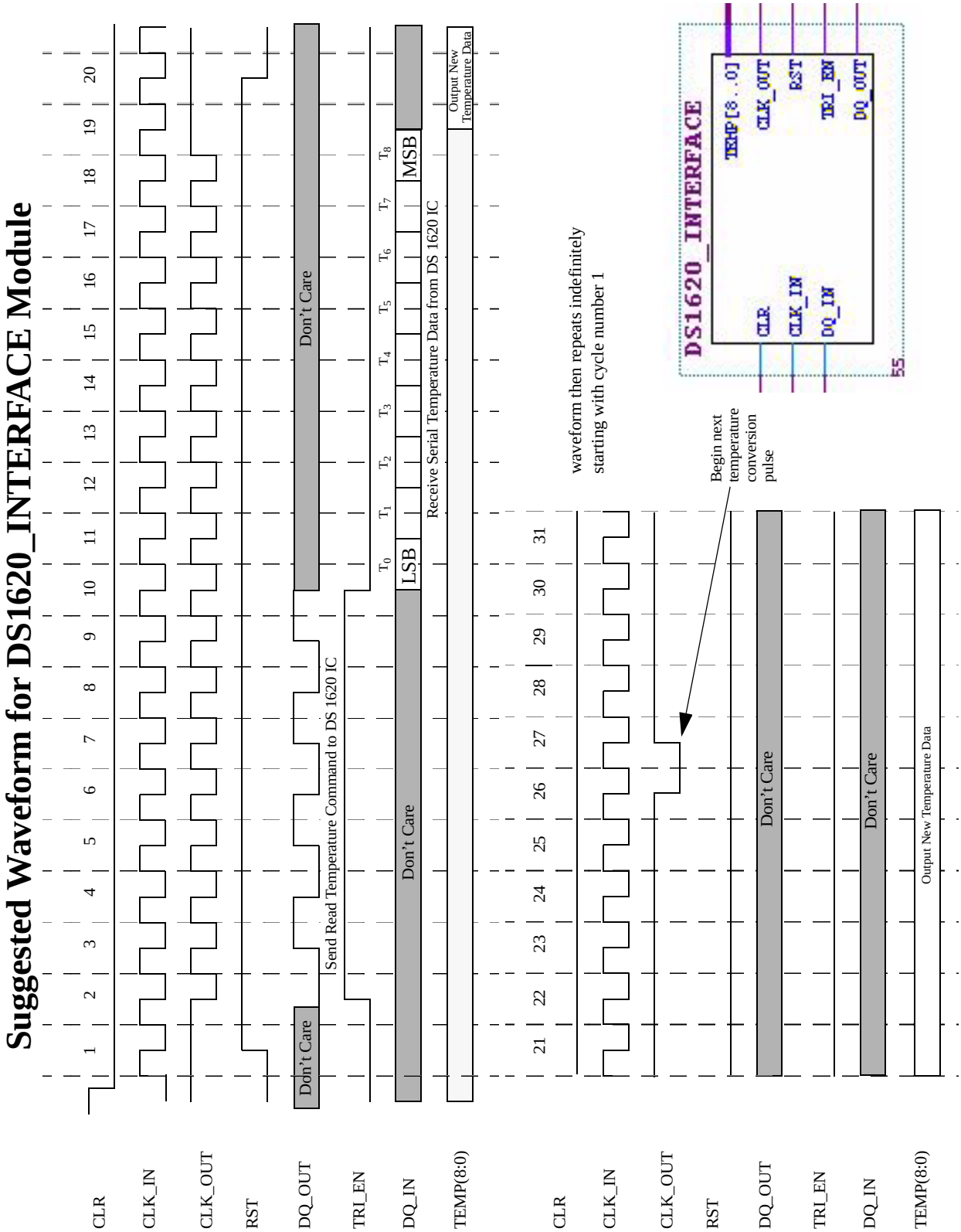
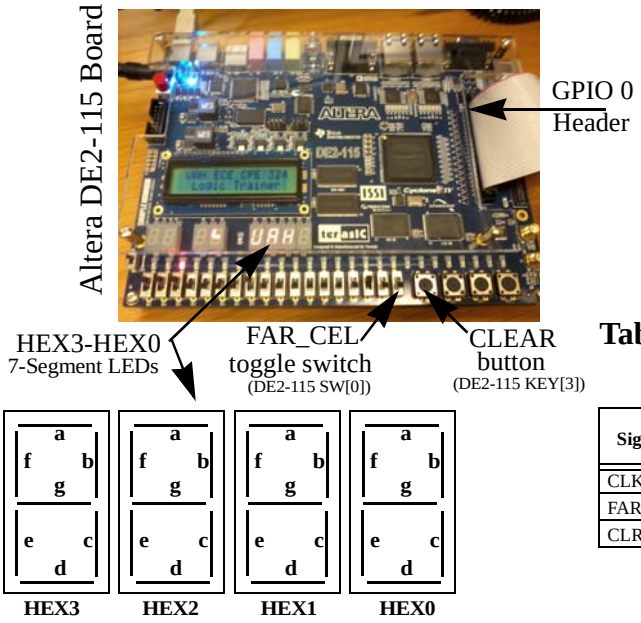


Figure 2. Suggest Waveform for DS1620_INTERFACE Module

The prototype implementation of the temperature sensor design is to be implemented on the Altera DE2-115 Educational board. It is to utilize the GPIO 0 interface port which is connected to the DS 1620 IC in the manner shown below in Table 1:.

Table 1: Assumed wiring connections between DE2-115 board and DS 1620 IC

Signal Name	Cyclone IV E Pin Number	GPIO_0 JP1	Expansion Cable Wire	DS 1620 Pin Number	DS 1620 Pin Name
VCC		11	Red	8	V_{DD}
GND		12	Black	4	GND
DQ	PIN_AF16	14	blue/green #1	1	DQ
CLK_OUT	PIN_AD19	15	blue/green #2	2	CLK/$\overline{\text{CONV}}$
RST	PIN_AF15	16	blue/green #3	3	$\overline{\text{RST}}$



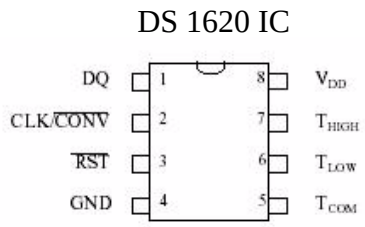


Table 2: 50 Mhz Internal Clock and Switches on the DE2-115 Board

Signal Name	Cyclone IV E Pin Number	Description
CLK_IN	PIN_Y2	50 MHz clock input
FAR_CEL	PIN_AB28	Toggle Switch SW[0]
CLR	PIN_R24	Push button Key Switch KEY[3]

Table 3: Cyclone IV E Pin Numbers for HEX3-HEX0 7-Seg LEDs on DE2-115 Board

Segment	HEX3		HEX2		HEX1		HEX0	
	Signal Name	Cyclone IV E Pin No.	Signal Name	Cyclone IV E Pin No.	Signal Name	Cyclone IV E Pin No.	Signal Name	Cyclone IV E Pin No.
a	HEX3[0]	PIN_V21	HEX2[0]	PIN_AA25	HEX1[0]	PIN_M24	HEX0[0]	PIN_G18
b	HEX3[1]	PIN_U21	HEX2[1]	PIN_AA26	HEX1[1]	PIN_Y22	HEX0[1]	PIN_F22
c	HEX3[2]	PIN_AB20	HEX2[2]	PIN_Y25	HEX1[2]	PIN_W21	HEX0[2]	PIN_E17
d	HEX3[3]	PIN_AA21	HEX2[3]	PIN_W26	HEX1[3]	PIN_W22	HEX0[3]	PIN_L26
e	HEX3[4]	PIN_AD24	HEX2[4]	PIN_Y26	HEX1[4]	PIN_W25	HEX0[4]	PIN_L25
f	HEX3[5]	PIN_AF23	HEX2[5]	PIN_W27	HEX1[5]	PIN_U23	HEX0[5]	PIN_J22
g	HEX3[6]	PIN_Y19	HEX2[6]	PIN_W28	HEX1[6]	PIN_U24	HEX0[6]	PIN_H22

During his last preliminary design review, before he was laid off, the engineer that preceded you proposed that the DS 1620 IC be run in thermostat mode with the interface logic being designed so it would periodically strobe the CLK/$\overline{\text{CONV}}$ pin of the DS 1620 to cause it to perform continuous temperature measurements and conversions. The temperature would then be read by the interface logic from the DS 1620 using the so called "three wire" mode that is discussed in the DS 1620 Data Sheet. In this mode, the DQ pin must first be driven in a serial manner by the interface circuitry that is being designed. Upon receiving the read temperature command, the direction of the DQ pin would then change and the DS 1620 would drive the DQ pin in a serial manner to output the 9 bit temperature reading. Then the process would be repeated in a continuous manner. The complete protocol for the three wire operation is described in detail in the DS 1620 data sheet but the previous design engineer was able to extract from this documentation a detailed timing diagram that related directly to the *DS1620_INTERFACE* element that was to be created. This timing diagram is shown in Figure 2. This waveform is composed of 31 cycles and begins after the first rising edge of the divided system clock after the CLR signal has gone low. It then repeats itself indefinitely.

Assignment

This project is already two months behind schedule and your supervisor has assigned you the task of completing the design. She has imposed a deadline of April 10, 2015 for you to develop a complete functioning prototype and to fully document the design. You need to complete and demonstrate this project during this time period -- the final round of lay-offs has been rumored for next month and your plan to get rich quick in the stock market is not working out as well as you had planned. To make matters worse the promised support Verilog Modules were actually written in VHDL not Verilog! This is ok though because you know that with the version of the Quartus II tool you can integrate your Verilog module into the VHDL design and will not have to convert that portion of the design the Verilog. This is being done by an intern and should be complete very soon! But you need to work concurrently.

The assignment for this class laboratory includes

1) The design and complete documentation of the *DS1620_INTERFACE* module
2) The complete simulation of the *DS1620_INTERFACE* design from Simulation 3.
3) A complete description of the operation of the Temperature Sensor design shown in Figure 1.

Each student is to work individually and demonstrate the design to the laboratory instructor. The final report should conform to the previously released standard and include documented Verilog code for the DS1620_INTERFACE module, simulation results (repeat of Simulation 3 assignment), and operational description of temperature sensor design. Also in your report indicate whether the design worked as expected the first time the DS1620_INTERFACE module was inserted or if further debugging was required before it worked correctly. If this is the case then indicate what issue or issues were present that were not detected by simulation. It is ok to reuse some of your text from your previously submitted Simulation 3 assignment in CPE 322 where it is applicable.

Practical IP Core Notes

To facilitate your design effort, a structural VHDL model of the schematic diagram that is shown in Figure 1, is available course Canvas site. The file *lab8_ip_core_elements.vhd.txt* contains the top-level design and all the supporting IP core elements except the *DS1620_INTERFACE* module that you have been asked to design. A template file for the DS1620_INTERFACE module is also present it is named DS1620_INTERFACE.v.txt. A recommended procedure is to first, open up Quartus II and create a new project for your design that targets the Cyclone IV E board's EP4CE115F29C7 device. Name this project *lab8*, same as the top-level entity in the VHDL file. Then create a new VHDL type file. Using the windows clip-board copy the contents of the *lab8_ip_core_elements.vhd.txt* file and paste into it in this new VHDL file. Save the file giving it the name *lab8_ip_core.vhd* keeping checked the box that indicates that the file will be included in the project. In a similar manner create a new Verilog HDL file, place the contents of this module you have created into this file and name it *lab8.v* and keep checked that this file is to be included into the project. Then compile your design once. Then enter all the pin assignments for the design. These are contained in Tables 1-3 and then compile the design again. In order to function correctly with theVHDL entered components, the DS1620_INTERFACE module you create should list the port signals in the order shown below.

```
// Verilog  Module to input temperature data from the DS 1620
// temperature IC
// template by
// B. Earl Wells, ECE Department University of Alabama in Huntsville
// inputs:
//    CLR an active high clear input which will reset internal signals
//    and restart the temperature acquisition process
//    CLK_IN the input clock (must be less than 1 MHZ)
//    DQ_IN the input signal that is used to input data from the
//        external DS 1620 IC. The 1620 only produces this data in
//        response to a command issued from the external device that
//        occurred over the "three wire" communication port. When data
//        is sent from the DS 1620, it is in a bit serial manner. Such
//        data is assumed to be valid at the rising edge of the CLK_OUT
//        signal. In this design the DQ_IN signal is used to receive
//        the 9 bit Celsius temperature from the DS 1620
// outputs:
//    TEMP the temperature acquired from the DS 1620 in 9 bit two's
//        complement format. (Output Resolution is 1/2 degree C)
//    CLK_OUT the clocking signal that drives the DS 1620 during its
//        "three wire" communication and the start conversion signal
//        that causes the DS 1620 to make the next temperature
//        conversion
//    RST the reset control line of the DS 1620 which is high during
//        "three wire" communication but low in all other cases
//    TRI_EN the controling signal for the external tristate buffer.
//        This signal indicates when the DQ input of the external
//        DS 1620 will be driven by DQ_OUT (this occurs when TRI_EN='1')
//        or when the DS 1620 will drive the DQ_IN input (this occurs
//        when TRI_EN='0')
//    DQ_OUT the output signal that drives the DS 1620 during "three
//        wire" communication. Communication along this signal occurs
//        in close coordination with the CLK_OUT signal. Such
//        communication is bit serial with the valid logic value being
//        placed on the DQ_OUT line at least 50 ns before the rising
//        edge of CLK_OUT. Note in this design DQ_OUT is used to send
//        the read temperature command to the DS 1620.

module DS1620_INTERFACE(input CLR, CLK_IN, output reg [8:0] TEMP,
   output CLK_OUT,RST,TRI_EN, input DQ_IN, output reg DQ_OUT);

// Enter your model for the DS1620_INTERFACE HERE!

endmodule
```