Complex Engineering Activity

Integrating UART to Pipelined Processor

Course Code and Title: EE-475L Digital Systems
Semester: Spring 2024
Instructors: Miss Shehzeen Malik
Total Marks: 30
Deadline: Last week of semester

CLOs and PLOs for Complex Engineering Problem

CLOs		**Description**	**Domains & Levels**	**PLOs, Levels**
CLO1	Lab	Understand and apply the knowledge of Instruction Set Architecture and digital systems to build different elements of a processor.	Cognitive	PLO1 Apply
CLO2	Lab	Design, implement and verify a pipelined processor with given specifications.	Psychomotor	PLO3 Articulation
CLO3	Lab	Demonstrate teamwork and project management.	Affective	PLO11 Organization

Problem Statement

UART is one of the serial communication protocols used in microcontrollers, embedded systems, and other electronic devices. This complex engineering activity requires design, implementation, demonstration, and documentation of a UART transmitter and receiver module integrated with the pipelined processor with interrupts generated for transmission and receiving completion of a byte and CSRs accomodating UART interrupt.

Objectives

The objective of this complex engineering problem is to familiarize students integration of peripherals with the processor and dealing with external interrupts. In the process, enhancing

Copyright © 2024 Department of Electrical Engineering, University of Engineering and Technology Lahore, Pakistan. Permission is granted to copy and distribute for educational purpose. However, any commercial use of the material, in any form, is not allowed.

their understanding of the design process of a serial communication protocol – UART, following datapath and controller paradigm.

Tasks

Design and implement Datapath and Controller of a UART transmitter and receiver module with the following specifications and integrate it with the pipelined processor.

1. UART Data frame constituting one start-bit, 8-data bits and one or two stop-bits.
2. Number of stop bits will be decided by writing 1 (two stop bits) or zero (one stop bit) in bit-1 of control register.
3. Receiver baud works at 16 times the transmitter. It samples start bit after 8 cycles and remaining bits after 16 cycles of sampling the start bit (i.e. every bit is sampled at mid-point of baud counter).
4. UART Rx-Data register has 31st bit to show it is busy receiving and bits 0-7 for storing the byte received.
5. Interrupt enable and pending registers for UART.
   - Bit-0 will be for transmission completion of a byte. Setting it 1 in enable register will enable this interrupt and will signal completion of transmission in pending register.
   - Bit-1 will be for single byte receiving completion interrupt.
6. Designate bit-16 of MIP and MIE CSR registers for UART interrupt and exception code 16 for UART Interrupt in MCAUSE register.
7. Write a C-language or assembly code to take the 8-bits data from memory into processor registers for transmitting through UART. The UART trap handler should read data from UART Rx-Data register and store it back to memory. Remember to populate CSR registers as required.

Bonus Task

Implement FIFOs of depth 8 for UART Tx- and Rx- data registers. In that case, Tx-full bit will be 1 when Tx-FIFO is full.

Deliverables

1. A report containing the following items:
   (a) Top Level Diagram of the design.
   (b) The designed datapath circuit along-with control signals. The design should also show the labels of wires, along with the label for the width of the line for multi-bit wires.

Copyright © 2024 Department of Electrical Engineering, University of Engineering and Technology Lahore, Pakistan. Permission is granted to copy and distribute for educational purpose. However, any commercial use of the material, in any form, is not allowed.

(c) State Machines for the controller of UART transmitter and receiver circuit.

(d) Contribution of each member in the entire process.

2. Simulation of your designed module on QuestaSim or any simulator. Show the simulation to the instructor. (No CEA will be considered without proper simulation.)

3. FPGA Implementation: Synthesize the circuit for Nexys A7 FPGA available in the lab. Tie the in/outs of UART to Pmod Header connector. The data can be observed on the computer terminal putty using the UART to USB converter.

Copyright © 2024 Department of Electrical Engineering, University of Engineering and Technology Lahore, Pakistan. Permission is granted to copy and distribute for educational purpose. However, any commercial use of the material, in any form, is not allowed.