HLS ASSIGNMENT

Total Points: 50 | Questions: 10 | Date: March 04, 2026

AI-generated graduate-level electrical assignment. Contains 10 questions covering key concepts.

Question 1

5.0 points

Explain the role of Control and Data Flow Graphs (CDFGs) in the high-level synthesis of FPGAs. How do CDFGs assist in optimizing hardware design?

Question 2

5.0 points

Given the task graph in the diagram below with three tasks T1, T2, and T3 executed on an FPGA, where T1 takes 5 ns, T2 takes 10 ns, and T3 takes 7 ns, and considering that T1 must finish before T2 and T3 can't start until T2 is done, what is the minimum execution time for all tasks on the FPGA?

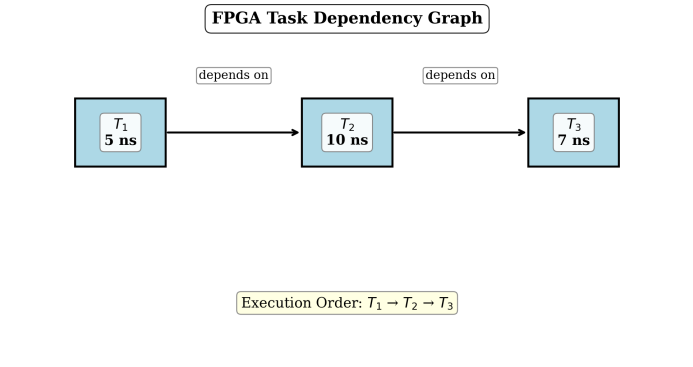

Fig. 2. Circuit diagram for Question 2

Question 3

6.0 points

Analyze the task graph in the diagram below and determine the critical path. Assume each task's execution time is shown in the diagram. What is the length of the critical path?

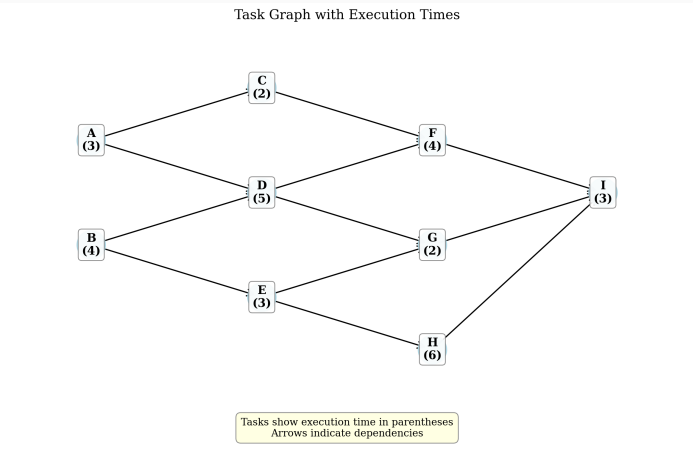

Fig. 3. Circuit diagram for Question 3

Question 4

6.0 points

Write a C++ function that represents a simple task scheduling algorithm for the task dependencies illustrated in the diagram below. Tasks: Task A must finish before Task B and Task C. Task B and C are independent after A finishes. Provide a function template.

```
void scheduleTasks() {
    // Define logic here
}
```

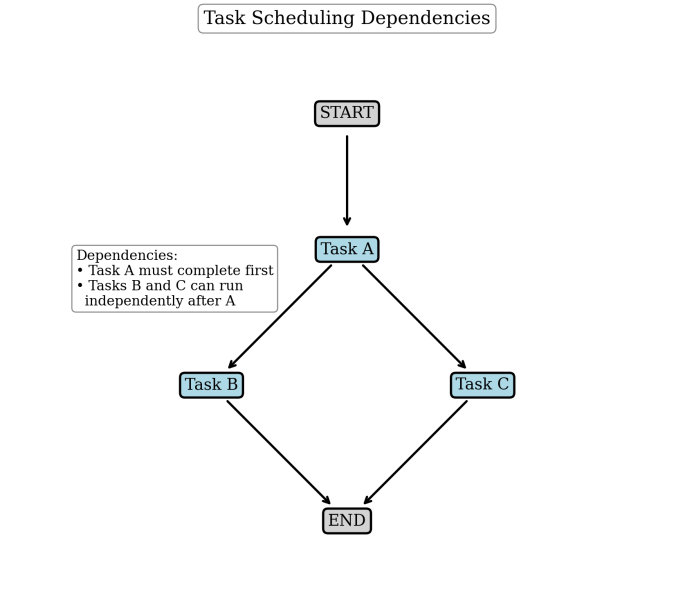

Fig. 4. Circuit diagram for Question 4

Question 5

6.0 points

Convert the CDFG expressed in pseudo code from the diagram below to a set of RTL operations suitable for FPGA synthesis. Assume simple arithmetic operations take 1 cycle each. Pseudo Code: X = A + B; Y = X - C; Z = Y * D.

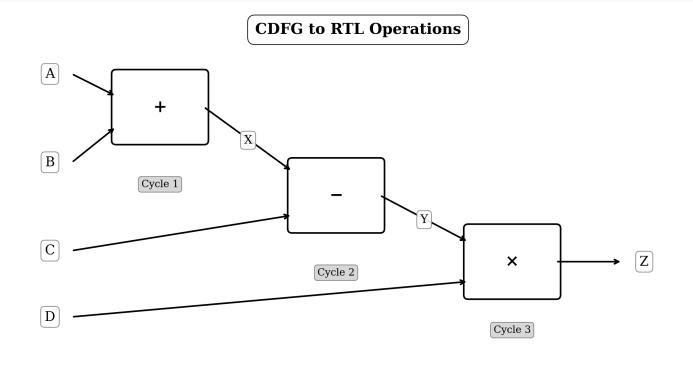

Fig. 5. Circuit diagram for Question 5

Question 6

5.0 points

Discuss the significance of task parallelism in FPGA-based systems compared to CPU-based systems. Focus on how high-level synthesis facilitates this parallelism.

Question 7 7.0 points

Given a C++ code snippet that calculates the greatest common divisor (GCD) of two numbers, convert this code into an equivalent Verilog module. Provide a Verilog code template for the GCD calculation.

```
module gcd(input wire [7:0] a, input wire [7:0] b, output wire [7:0]
gcd_out);
    // Declare internal signals
    // GCD logic
endmodule
```

Question 8 4.0 points

Examine the sample CDFG provided in the diagram below. Identify and list all dependencies between operations and categorize each as data or control dependencies.

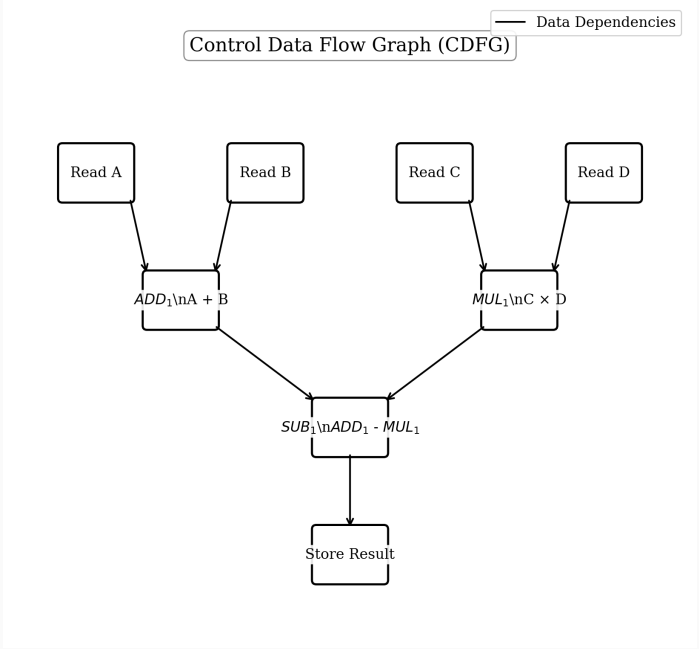

Fig. 8. Circuit diagram for Question 8

Question 9 3.0 points

Describe the process of converting a task graph into a hardware schedule suitable for FPGA implementation. Mention the challenges involved during this conversion.

Question 10 3.0 points

Assume an FPGA implementation schedule must meet a throughput of 100 operations per microsecond. If each operation requires 100 ns on average, calculate the required level of parallelism (number of concurrent operations).