EDA322/ DIT797: Digital Design
Exam - March 2023

Date: March 15, 2023

Time: **14:00-18:00**

Examiner: Ioannis Sourdis

Department: Computer Science and Engineering

Inquiries: contact through phone, phone extension 1744

Duration: 4 hours

Grading scale: 100 points in total

Chalmers:
0: 0%-49%, 3: 50%-64%, 4: 65%-84%, 5: 85%-100%
GU:
Fail (U): 0%-49%, Pass (G): 50%-79%, Pass with Distinction (VG): 80%-100%

Available references: an A4 paper sheet (2 pages) with student notes, a calculator, are allowed.

General: Submit your solutions, in English, on blank paper sheets. Write legibly; feel free to use figures to get your point across.

The order of answering the questions does not matter (start with the easiest ones).

Please start the solutions for each problem on a new sheet. Please number the sheets so that the solutions are in numerical order.

Note that it is possible to receive partial credit for an answer even if it is not 100% correct.

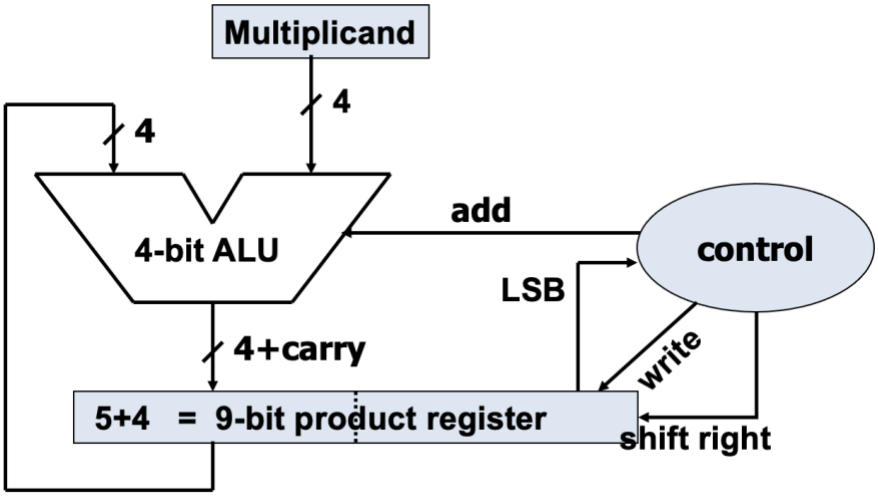
Your personal identity code is required on each submitted sheet!

Good luck!

Note: *These are example answers to the exam questions for one possible set of randomized variables.*

Question 1 Arithmetic: (10 points)

- Show the contents of the registers in a following serial 4-bit multiplier at every step when performing the following operation 5*3.
- How would the multiplier support the operation -5*3?



Answer

Similar to lecture on Arithmetic, slides 11-12.

Question 2 Interfaces: (10 points)

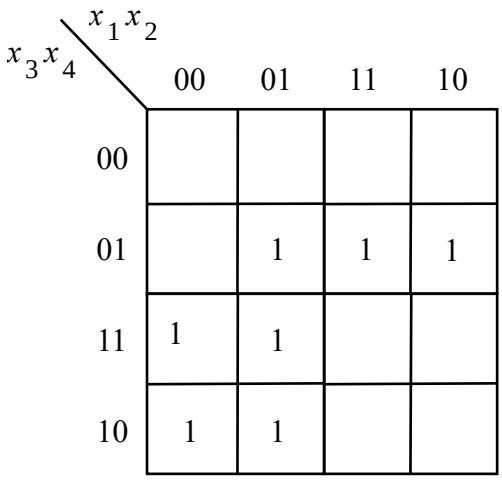
State the differences between the push and a pull flow control interface. Why would you use one of them instead of the full flow control interface.

Answer

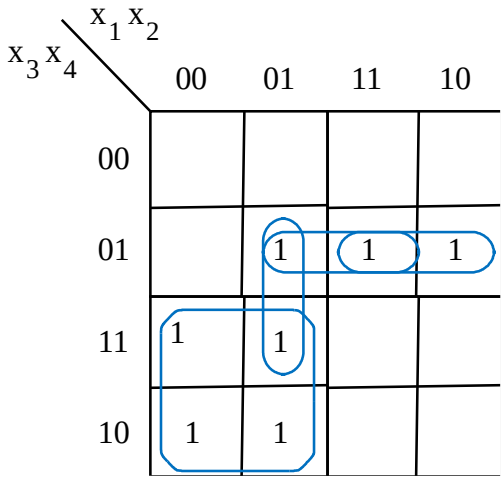
Lecture in Interfaces slide 8. We'd use one of them if possible because it is simpler and requires less logic.

Question 3 Hazards: (10 points)

- Find and remove all hazards in the K-map:



Answer



Show the minimal function first (e.g. f= x3x1' + x4x1'x2 + x3'x4x1)
And then explain where the hazard is and how it can be avoided (e.g. adding term x3'x4x2) [OPTIONAL]

Hazard free version: f= x3x1' + x4x1'x2 + x3'x4x1 + x3'x4x2

Question 4 Number representation: (10 points)
Suppose you need to represent distance from 0 to 100 meters (m) with a maximum absolute error of 1 mm. Find a fixed point representation with minimum number of bits that fits the above specification.

Answer

a) 1 mm absolute error needs a resolution of 2 mm = 1/500 m.

If meters is the integer part we need, then we need 7 bits for integer (to count between 0-100 meters) because 2^7= 128. Then, the fraction needs 9 bits to count $1/2^9$ = 1/512 meters (which is a bit shorter than the 1/500 meters resolution). So the fixed-point representation would require 16 bits (7.9).

Alternatively, we can count multiples of 2 mm which is the required resolution. 1 meter = 50000 * 2mm for which we need again 16 bits (2^{16}=64K).

Question 5 Pipelining, Timing and FPGAs: (10 points)
Consider an unpipelined 8-bit Ripple Carry Adder implemented in an FPGA with 3-bit input LUTs. Consider also that the delay of an LUT is 1 ns, the setup time of a flipflop is 100 ps and the propagation time is 100ps.

a) What is the latency, maximum operating frequency and throughput of the unpipelined version? Consider that inputs and outputs of the unpipelined version are registered.
b) Pipeline the adder to increase its throughput by 2 times. What is the latency, maximum operating frequency and throughput of the pipelined adder?

Answer
The implementation of the Ripple carry adder in the FPGA would need 2 LUTs (parallel to each other) for each full adder, one for the sum bit and another for the carry out (both LUTs would have the same inputs: the two bits of the operands and the carry in).

a) latency of unpipelined RCA: 8*1 +0.1 +0.1= 8.2 ns

max operating frequency: 1/8.2 GHz ~= 0.122 GHz = 122 MHz

Throughput is one addition per 8.2 ns = 122 Mops/sec

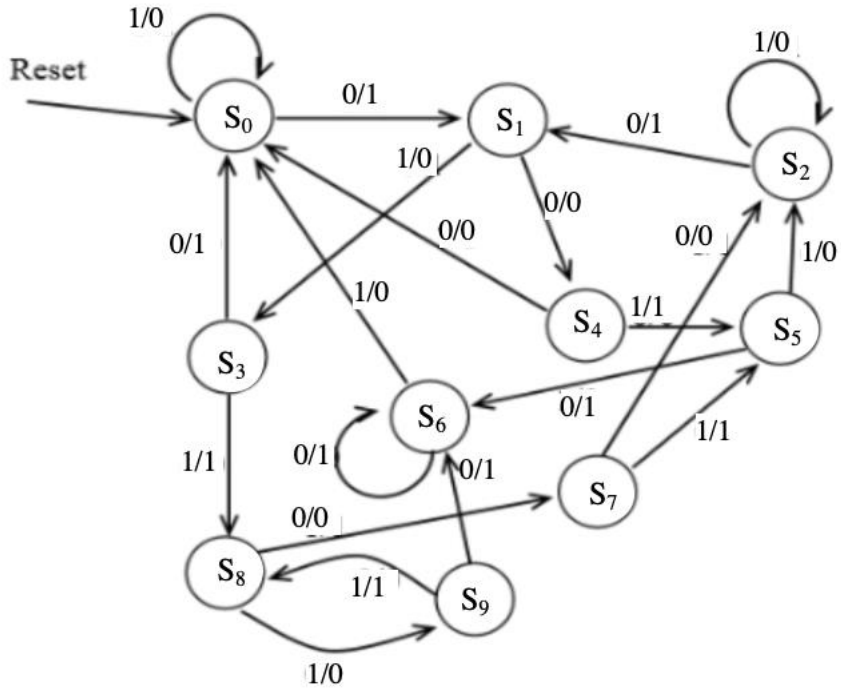
b) In order to double the throughput we need one results at least every 8.2/2 ns = 4.1 ns
from the 4.1 ns we need to take away the 0.1+0.1 ns needed for the propagation time of the starting register and the setup time of the receiving register.
So this leaves 3.9 ns for the combinational logic. In 3.9 ns we can fit only 3 LUT delays, so 3 full adders.

So in order to double the throughput we need to split the adder in 2 stages of 3+3+2 FAs in each stage

The latency of the pipelined design would be 3 times the latency of the longest stage (the one with the 3 FAs) so 3*(3+0.1+0.1) ns= 9.6 ns

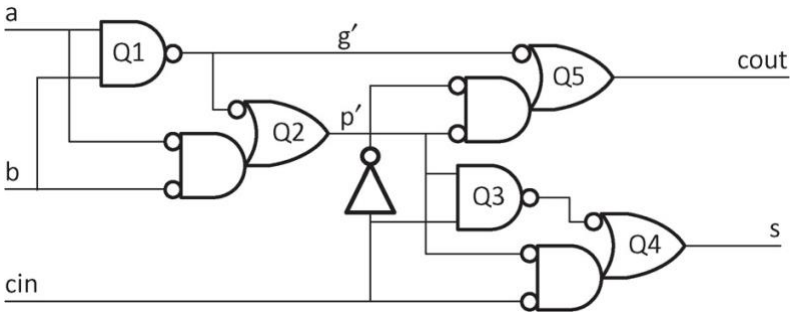
Its minimum clock period is the delay of the 3 FAs plus the setup and propagation times: 3+0.1+0.1 ns = 3.2 ns
The maximum operating frequency would be 1/(3+0.1+0.1) GHz = 312.5 MHz
And its throughput would be 1 operation every 3.2 ns = 312.5 Mops/sec

Question 6 FSMs: (10 points)
The state diagram defines an FSM with an Input X and an output Z. The values on each arrow in the diagram have the format of X/Z.
Minimize the number of states of this FSM, list the equivalent states and draw the minimized state diagram.



Answer

Question 7 Testing: (10 points)
Activate, propagate and justify the stuck-at-0 fault at the g' (note wire p' is not connected with the output if the inverter).

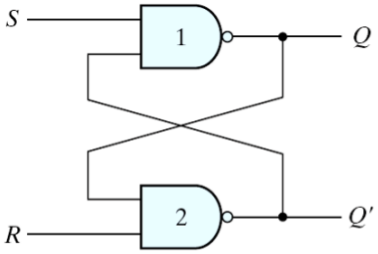


Answer
a=0 or b=0 to activate the fault

to propagate through Q5, the lower input of Q5 should be 0, which needs any input of the AND gate (after the invention) to be 0, e.g., cin=0.

Question 8 Asynchronous Sequential circuits: (10 points)
Analyze the NAND-based S'R'-latch as an asynchronous circuit. Explain what is the problem when SR changes from 00 to 11.

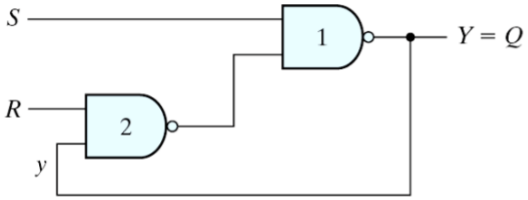
Answer



(a) Cross-coupled circuit

S	R	Q	Q'	
1	0	0	1	
1	1	0	1	(After $SR = 10$)
0	1	1	0	
1	1	1	0	(After $SR = 01$)
0	0	1	1	

(b) Truth table



(c) Circuit showing feedback

y \ SR	00	01	11	10
0	1	1	0	0
1	1	1	1	0

(d) Transition table

SR latch with NAND gates

Y= [S(Ry)']' = S' +Ry

Depending on the various delays and assuming SR=00 changes to SR=11...
If SR=00 -> SR=10 -> SR=11, we get stable state with output of 0.
If SR=00 -> SR=01 -> SR=11, we get stable state with output of 1.

Question 9 Memory and sequential circuits: (10 points)

- Draw the gate level block diagram of the memory with 4 entries, each entry having 2-bits using a D-latch for each memory cell (show the gatelevel of one of the D-latches, too).
- Now, replace the D-latches with D-Flip-flops (show the gatelevel of one D-flipflop), make any additional changes to the design in order to ensure that the memory works correctly, and justify your choice (even if you decide not to make any changes).
- What changed in the functionality of the memory?

Answer

- Similar to Lecture on memory, slide 25.
- Additional changes: the decoded address needs to be registered before going to the output AND gate of each D-flipflop. That is because the D-flipflop will provide its contents a cycle after a read request is made.
- The change is that the memory now answers a cycle after the read/write request.

Question 10 Power and Arithmetic: (10 points)
Consider an 8-bit ripple carry adder and an 8-bit carry select adder, which is split in two blocks of 4-bits each. Assume also that the area and delay of the multiplexers in the carry select adder are zero. If each design operates at their maximum frequency, how do they compare in terms of performance per watt. Which of the two designs is more energy efficient?

Note: assume Power consumption is only due to dynamic power.

Answer
If multiplexers have zero delay and area, then the carry select adder (CSA) has half the delay (d) compared to the ripple carry adder (RCA) (2*d). This means that CSA can have double the frequency (2*f) compared to RCA (f). It also has 1.5 times higher area (1.5*A) compared to the ripple carry (A).

The dynamic power depends is proportional to the frequency of the design and to the area (number of gates). Considering everything else (activity, capacitance, voltage) are the same:

$P_{RCA} = f*A*K$
And $P_{CSA} = 2*f*1.5A*K$ so
$P_{CSA} = 3*P_{RCA}$, the power consumption of CSA is 3 times higher than the power of RCA.

Performance depends on the frequency, so CSA has double performance than RCA.

So if Performance per Watt of the RCA is Perf/Power then CSA performance per Watt is 2*Perf/3*Power, i.e. 66% of RCA performance per watt.

Performance per Watt is an energy efficiency metric. It is actually the inverse of energy per operation, which is 1/(Perf/Watt). So, the higher the performance per Watt the higher the energy efficiency. That means that the RCA design is more energy efficient.

END of EXAM