CS2100 Computer Organisation

Lab 09: Gate-Level and Block-Level Design (Week 4th November)

Instruction

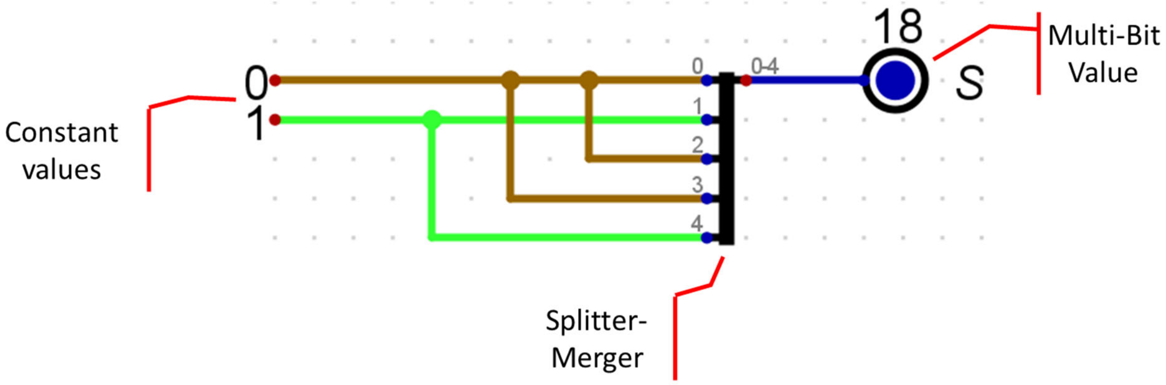
Short and clean
We have separated the lab information into i) **instruction** and ii) **report**. Whenever there is a question in the instruction (easily identified as they have **[X pts]** tagged to the end), write / type your answer in the corresponding location in the **report** document. Please take note of the submission specification at the end of this document.

Objective

1. To learn the difference between gate-level design and block-level design.
2. To learn the behavior of parallel adder.

New components in Digital

Fire up **Digital** and use “File→Open” to open the provided “Lab 09 – Circuit Template.dig”. The template circuit contains the inputs, outputs and the two required components for this lab (NAND gate and a 4-bit Cascade Adder / Parallel Adder). There are three new **Digital** components, captured below in one example to illustrate their functionality:



- **[Constant Values]** Provide the constant 0 (false) or 1 (true) for your circuit. Use these if you find that you need to tie the input of a component to a 0 or 1 permanently.
- **[Multi-Bit Value]** The “input” and “output” component can be set to multiple bits. The example here is a 5-bit output. Benefits includes meaningful display of the 5 bits as a single value, e.g. the above show the 5-bit 10010 as 18_{10}.
- **[Splitter-Merger]** This component allows us to merger or split input / output. The example shown above merge 5 1-bit input into a single 5-bit output.

Procedure:

1. You are to design a **multiply-by-6** circuit given a 3-bit binary unsigned value *ABC* as its input. The circuit generates a 5-bit binary number $S_4\ S_3\ S_2\ S_1\ S_0$, and an output ***V***.

   Since the circuit may not accommodate certain input value owing to the limited number of output bits, **such an input value is deemed invalid**, and the corresponding output will be **don't-care values**. The output ***V*** is used to indicate whether the input value is valid or not: **1** if the input value is valid, or **0** otherwise.

   Give the truth table in report. **[4 pts]**

2. **[Gate-Level Design]** Give the K-maps for S_4 , S_3 , S_2 , S_1 and S_0. Write out the simplified SOP (sum-of-products) expression for each of them. **[5 pts]**

3. **[Block Level Design]** **Instead of using the results from (2)**, let's implement the multiply-by-6 circuit using a 4-bit cascade adder (parallel adder). The 4-bit adder is used to generate the 5-bit output **$S_4S_3S_2S_1S_0$**. [Hint: Think in binary ☺]

   For the output **V**, you are to use a NAND gate to implement it.

   Design your circuit in **Digital** and paste a screenshot of the complete circuit into the report. **You are not allowed to add any additional digital logic component** (i.e. you can solve the lab just by connecting the components). Note that the output "S" is a 5-bit value, connect the **$S_4S_3S_2S_1S_0$** accordingly. You can add more constants (1 / 0) components to reduce wiring mess. **[7 pts]**

from truth table s0
is always 0, so it can
be wired to a 0 input

idea is to change
(a * 6) into
(a + a * 2) * 2
where mult by 2 is
just a shift left operation

notice the mapping
in decimal
0 -> 0
1 -> 3
2 -> 6
3 -> 9
4 -> 12
5 -> 15
6 -> X
7 -> X
just need to
get this value
and then shift
left by 1 to get
'multiply by 6'

4. Implement the circuit in (3). Show your circuit to your lab TA. **[4 pts]**

Common Errors:

- Remember to connect **all** inputs to some values (remember the **constant values**).
- Misread the significance of the multi-bit values. The "0th index" is the **least significant** bit.

You need to submit into Luminus Folder no longer than **2359 on the same day** you have the online lab. Please rename the **report document** with your student id: **[student_id]_lab08.docx** or **[student_id]_lab08.pdf**, e.g. **A1234567X_lab08.pdf**. Submit into the correct folder on Luminus before 2359 of your lab day.

Marking Scheme: Report – 16 marks; Demonstration – 4 marks; Total: 20 marks.