Computation II: embedded system design (5EIB0)

Description | Submission | Edit | Submission view

Available from: Friday, 5 April 2024, 2:00 PM
Requested files: parametrized_counter.v (Download)
Type of work: Individual work

Assignment

In this assignment, you have to create a Verilog implementation of a parametrized counter (**parametrized_counter**). When the counter is triggered, it will run for a number of cycles, and then stop counting until it is triggered again.

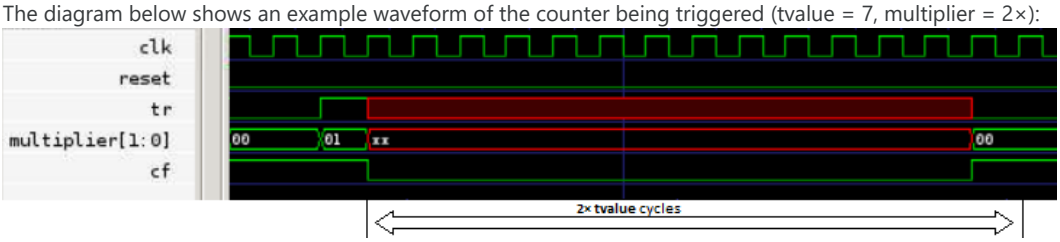
Compared to a normal counter, this parametrized counter has a special feature: it can count **1n**, **2n**, **4n** and **8n** cycles.

- The value of '**n**' can be customized using a Verilog parameter.
- The multiplier (1, 2, 4 or 8) can be selected using an input.

It has the following ports:

- **clk (1 bit input, positive edge):** Clock for the module
- **reset (1 bit input, active high, synchronous):** Reset for the module.
- **tr (1 bit input, active high):** Trigger (i.e. start) the counter. *If the counter was already running, nothing should happen.*
- **cf (1 bit output, active low):** Indicates the counter is running. In other words, it is high only when the counter is **not** running.
- **tvalue (parameter):** Value of '**n**'. For example, when the multiplier is **1**, **cf** should be low for ***tvalue*** - *1* cycles. *Your module must support **tvalue**s of at least 32 cycles.*
- **multiplier (2 bit input, little-endian):** Multiplier. **00** = 1×, **01** = 2×, **10** = 4×, **11** = 8×. For example, when the multiplier is **10**, **cf** should be low for 4 × **tvalue** - 1 cycles. *If this value is changed while the counter is already running, nothing should happen.*

The diagram below shows an example waveform of the counter being triggered (tvalue = 7, multiplier = 2×):



Debug

Click on the symbol marked below to see the waveforms produced by your design. Please note that no waveforms will be produced if your code has an error that prevents it from being simulated.

Requested files

parametrized_counter.v

```
1  module parametrized_counter (
2          input clk,
3          input reset,
4          input tr,
5          input [1:0] multiplier,
6          output cf
7      );
8
9      parameter tvalue = ...; // Replace the ... with a reasonable default value
10
11     // Add your implementation here
12
13 endmodule
```

VPL

Data retention summary