

Домашнее задание №2 по курсу «Введение в архитектуру вычислительных систем»

ФИО: Новикова Полина

Номер группы: Б01-304

Вариант: 5

Email: novikova.pp@phystech.edu

Пункт 1. Документация

Функциональное описание

Модуль `cnt_en_clr` представляет собой счетчик с сигналами разрешения счета и очистки. Основные функции:

- Счетчик увеличивает свое значение на 1 каждый такт, когда активен сигнал `cnt_en_i`
- При активации сигнала `cnt_clr_i` счетчик сбрасывается в 0
- При достижении максимального значения счетчик переполняется (сигнал `ovf_o`), и значение сбрасывается в 0
- Асинхронный сброс по отрицательному фронту `arstn_i`

Структурная схема

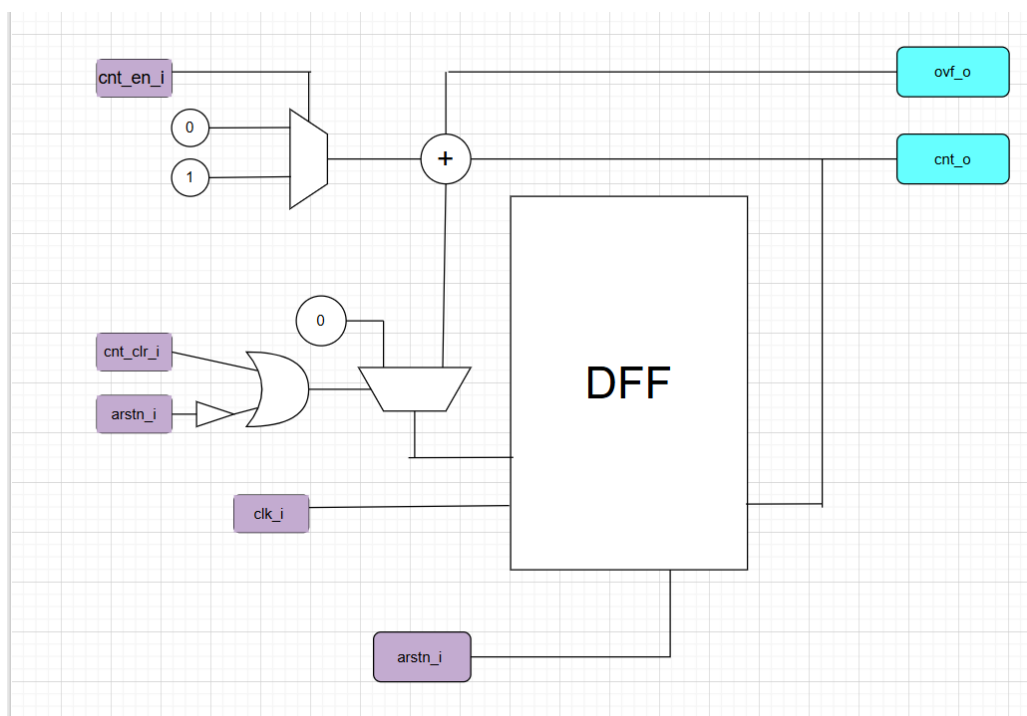


Рис. 1: Структурная схема модуля `cnt_en_clr`

Параметры

- CNT_WIDTH - ширина счетчика в битах (положительное целое число)

Порты

Название	Ширина	Направление	Описание
clk_i	1	Input	Тактовый сигнал
arstn_i	1	Input	Асинхронный сброс (активный 0)
cnt_en_i	1	Input	Разрешение счета
cnt_clr_i	1	Input	Сигнал очистки
cnt_o	CNT_WIDTH	Output	Значение счетчика
ovf_o	1	Output	Сигнал переполнения

Таблица 1: Описание портов модуля

Тактирование и сброс

- Тактирование: по положительному фронту clk_i
- Сброс: асинхронный, по отрицательному фронту arstn_i
- При сбросе: cnt_o = 0, ovf_o = 0

Тестирование

Тестбенч проверяет:

1. Асинхронный сброс
2. Очистку счетчика
3. Нормальный режим счета
4. Переполнение счетчика
5. Комбинации сигналов разрешения и очистки

Пункт 2. Имплементация

Код модуля с учетом всех необходимых требований:

```
1
2 module cnt_en_clr #(
3     parameter CNT_WIDTH = 8
4 ) (
5     input clk_i,
6     input arstn_i,
7     input cnt_en_i,
8     input cnt_clr_i,
9     output reg [CNT_WIDTH-1:0] cnt_o,
10    output reg ovf_o
11 );
12
13 always @(posedge clk_i or negedge arstn_i) begin
14     if (!arstn_i) begin
15         cnt_o <= {CNT_WIDTH{1'b0}};
```

```

16     ovf_o <= 1'b0;
17 end
18 else if (cnt_clr_i) begin
19     cnt_o <= {CNT_WIDTH{1'b0}};
20     ovf_o <= 1'b0;
21 end
22 else if (cnt_en_i) begin
23     if (cnt_o == {CNT_WIDTH{1'b1}}) begin
24         cnt_o <= {CNT_WIDTH{1'b0}};
25         ovf_o <= 1'b1;
26     end
27     else begin
28         cnt_o <= cnt_o + 1;
29     end
30 end
31 else begin
32     ovf_o <= ovf_o;
33 end
34 end
35
36 endmodule

```

Листинг 1: Модуль счетчика

Пункт 3. Тестирование

Напишем testbench, который будет проверять возможные сценарии работы модуля и посмотрим на временных диаграммах результаты. Сначала проследим за тем, в какой момент начинает меняться выходной сигнал. Затем протестируем работу модуля с активным `en-i`, то есть с разрешенным счетом и дойдем до переполнения. После этого посмотрим, как реагирует модуль на неактивный сигнал управления счетом. Затем очистим счетчик с помощью `cnt_clr-i`. И проверим, как работает асинхронный сброс.

Код для тестирования:

```

1  'timescale 1ns/1ps
2
3  module cnt_en_clr_tb();
4
5  localparam CNT_WIDTH = 4;
6
7  reg clk_i;
8  reg arstn_i;
9  reg cnt_en_i;
10 reg cnt_clr_i;
11 wire [CNT_WIDTH-1:0] cnt_o;
12 wire ovf_o;
13
14
15 cnt_en_clr #(
16     .CNT_WIDTH(CNT_WIDTH)
17 ) dut (
18     .clk_i(clk_i),
19     .arstn_i(arstn_i),
20     .cnt_en_i(cnt_en_i),
21     .cnt_clr_i(cnt_clr_i),
22     .cnt_o(cnt_o),
23     .ovf_o(ovf_o)
24 );

```

```

25
26
27 initial begin
28     clk_i = 0;
29     forever #5 clk_i = ~clk_i;
30 end
31
32 initial begin
33     $dumpvars;
34     arstn_i = 1;
35     cnt_en_i = 0;
36     cnt_clr_i = 0;
37
38     #10 arstn_i = 0;
39     #10 arstn_i = 1;
40
41     cnt_en_i = 1;
42     #100;
43
44     cnt_clr_i = 0;
45
46     cnt_en_i = 1;
47     #100;
48
49     cnt_en_i = 0;
50     #20;
51     cnt_en_i = 1;
52     #30;
53     cnt_clr_i = 1;
54     #10
55     cnt_clr_i = 0;
56     #35
57     arstn_i = 0;
58     #20
59     arstn_i = 1;
60     #30
61
62     $finish;
63 end
64
65 endmodule

```

Листинг 2: Testbench

Временные диаграммы

На временных диаграммах хорошо видны изменения выходных сигналов в зависимости от входных. Работа модуля соответствует функциональному описанию



Рис. 2: Временные диаграммы работы модуля

Отметим особые места на схеме:

1. Начало счета
2. Переполнение
3. Неактивный сигнал управления счетом - счет приостанавливается.
4. Очистка счетчика
5. Асинхронный сброс, по отрицательному фронту сигнала arstn-i