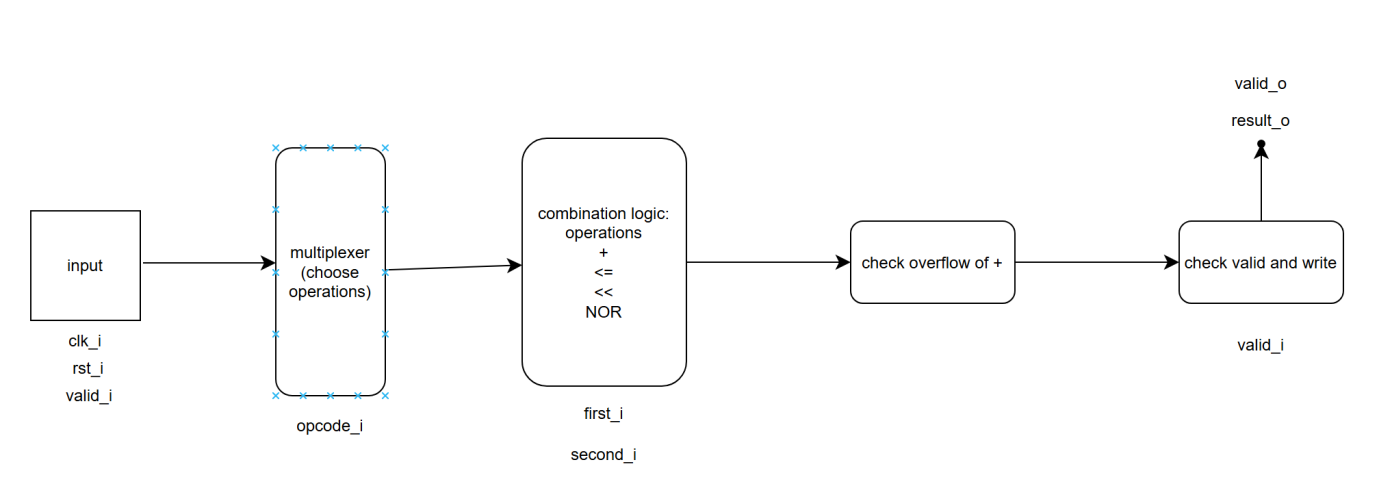
Титов Артём Александрович

Б01-306

Вариант № 3

[titov.artem@phystech.ru](mailto:titov.artem@phystech.ru)

Alu-register выполняет 4 арифметические операции (знаковое сложение, сравнение меньше или равно второго числа с первым, логический сдвиг второго числа влево на количество битов равное первому числу, побитовый NOR) с двумя числами заданной длины, сохраняя результат и проверяя результат на валидность.



**Параметры:** WIDTH длина операндов, должна быть больше 0.

**Порты:**

valid\_i сигнал валидности операции, сигнализирует о записи результата и о валидности выполненной операции. Принимает значения 0 или 1.

first\_i шина первого операнда, содержит значение первого операнда. Диапазон значений зависит от WIDTH для знакового сложения принимает значения от -2^(WIDTH – 1) до 2^(WIDTH – 1) – 1. Для остальных операций принимает значения от 0 до 2^N.

second\_i шина второго операнда, содержит значение второго операнда. Диапазон значений зависит от WIDTH для знакового сложения принимает значения от -2^(WIDTH – 1) до 2^(WIDTH – 1) – 1. Для остальных операций принимает значения от 0 до 2^N – 1.

opcode\_i шина кодирующая операции. Принимает значения от 0 до 3.

valid\_o сигнал валидности результата, сигнализирует последующим операция о корректном срабатывании. Принимает значения 0 или 1.

result\_o шина результата операции, сохраняет результат операции. Диапазон значений зависит от WIDTH для знакового сложения принимает значения от -2^(WIDTH – 1) до 2^(WIDTH – 1) – 1. Для остальных операций принимает значения от 0 до 2^N – 1.

**Тактирование и сброс:**

clk\_i тактирующий сигнал, при положительном фронте сигнала происходит проверка валидности и запись результата и валидности.

Вычисления происходят независимо от clk\_i, при любом изменении сигналов.

rst\_i сигнал сброса при значении 1 не происходит запись, и на выход подаётся сигнал о невалидности результата.

**Тестирование:**

WIDTH = 3

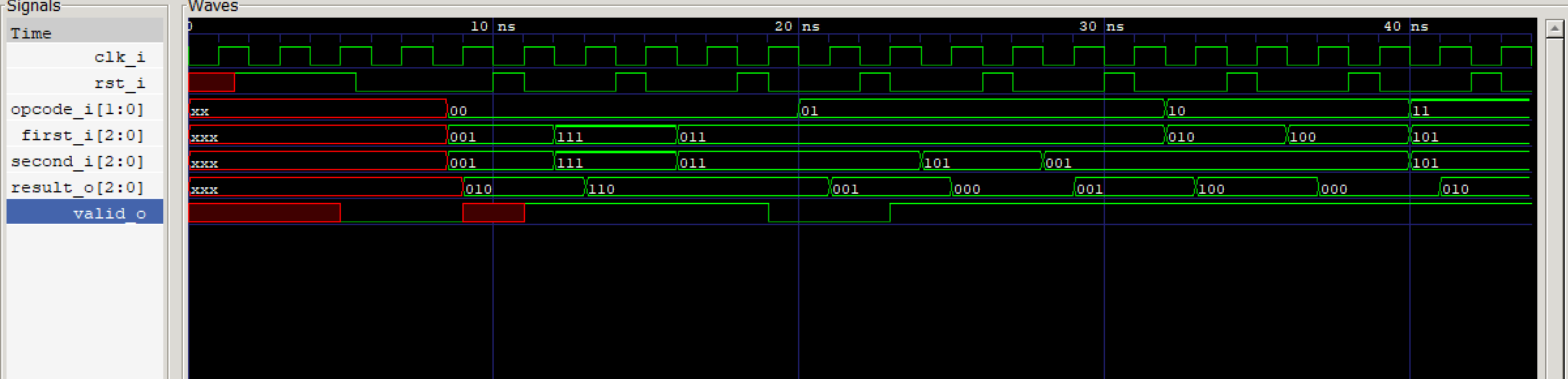
Для сложения рассмотрим сумму 001 + 001 = 010 (1 + 1 = 2); 111 + 111 = 110 ( - 1 – 1 = -2); а также переполнение 011 + 011 = 111 valid\_o должно стать 0.

Для <= рассмотрим равенство 011 <= 011 result\_o должен быть 1; для false возьмем second\_i = 101 и first\_i = 011 result\_o должен быть 0; И для true возьмём second\_i = 001 и first\_i = 011 result\_o должен быть 1.

Для << возьмём second\_i = 001 и first\_i = 010 result\_o должен быть 100; возьмём second\_i = 001 и first\_i = 100 result\_o должен быть 000.

Для NOR возьмём second\_i = 101 и first\_i = 101 result\_o должен быть 010.

**Скриншот диаграммы:**

****