

Lab3_z2

- В папке C:\Xilinx_trn\HLS2022\lab3_z2 создайте папку source (для исходных файлов и теста)
- Создайте на языке Си (и сохраните в папку C:\Xilinx_trn\HLS2022\lab3_z2\source)
 - Последовательно считывающую элементы массива, заполненного случайными целыми числами (тип short), имеющими значения от **0 до M-1**. Размер массива - **N** элементов.
 - Функция подсчитывает количество появлений чисел от 0 до **M-1** во входном массиве и записывает полученные значения в соответствующие элементы (с 0 до **M-1**) выходного массива, имеющего **M** элементов типа short.
Т.е. по входному массиву размером N, заполненного случайными целыми числами от 0 до M-1 строится гистограмма в выходном массиве размером M элементов.
 - В файле lab3_z2.h должны быть определены **M** и **N** и тип данных data_sc, имеющий тип short.

Например, можно примерно так (надо поменять название функции на lab3_z2 и – на ваше усмотрение):

```
void lab (data_sc inA_ar[N], data_sc out_ar[M])
{
    int i;
    data_sc temp;
    for (i = 0; i < N; i++) {
        temp = inA_ar[i];
        out_ar[temp] = out_ar[temp]+1;
    }
}
```

- Создать на языке Си тест для проверки работы функции. Тест должен обеспечивать
 - Запуск функции 3 раза
 - каждый раз с новым начальным значением от генератора случайных чисел (от 0 до 8191).
 - Перед следующим запуском функции не забудьте обнулить формируемый функцией выходной массив.
 - Формирование исходного входного массива, заполненного случайными числами.
 - N=8192, M=8192 (заданы в h файле)
 - проверку правильности полученного функцией результата (сравнение полученного функцией массива и ожидаемого массива, вычисленного в тесте с использованием другого, не такого как в функции, алгоритма построения гистограммы) и формирование признака успешного/неуспешного выполнения для каждого запуска функции.
- Отладить функцию и тест (при неправильном результате в любом из запусков функции должен сообщать об ошибке).
- Создать скрипт автоматизирующий процесс:
 - Создания проекта lab3_z2,

- Микросхема: xa7a12tcs325-1Q
 - Период тактового сигнала: 8нс, uncertainty 1нс.
-
- Подключения файла lab3_z2.c (папка source),
- Подключения файла lab3_z2_test.c (папка source),
- Си моделирование
- Создания решений
 - sol1, для которого
 - Block Level I/O и Port Level I/O interface по умолчанию
 - Синтез
 - C/RTL cosimulation (с опцией полной трассировки)
 - sol2, для которого
 - Block Level I/O по умолчанию
 - Port Level I/O interface **ap_fifo** ТОЛЬКО для входного массива
 - Синтез
 - C/RTL cosimulation (с опцией полной трассировки)
 - sol3, для которого
 - Block Level I/O по умолчанию
 - Port Level I/O interface **ap_fifo** ТОЛЬКО для выходного массива
 - Синтез
 - C/RTL cosimulation (с опцией полной трассировки)
- Отладить и проверить работу созданного скрипта.
- После выполнения скрипта открыть GUI
- Убедиться, что созданы все решения
- Используя средства HLS сравните полученные решения
 - использованные интерфейсы
 - аппаратные затраты
 - результаты планирования (Schedule viewer)
 - C/RTL cosimulation
 - Заполненную xls таблицу и полученный график
 - объясните полученные результаты.
- Оформить отчет, который должен включать
 - Задание
 - Раздел с описанием исходного кода функции
 - Раздел с описанием теста
 - Раздел с описанием созданного командного файла
 - Раздел с описанием результатов сравнения решений (со снимками экрана)
 - Раздел с анализом результатов
 - Анализ и выбор оптимального (критерий максимальная производительность) решения
 - Выводы

Архив должен включать всю рабочую папку проекта, отчет