ГУАП

КАФЕДРА № 44

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

доц., канд. техн. наук, доц.		О.О. Жаринов
должность, уч. степень, звание	подпись, дата	инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №5

РАЗРАБОТКА ФОРМИРОВАТЕЛЯ ИМПУЛЬСОВ, УПРАВЛЯЕМОГО ЦИФРОВЫМ КОДОМ С ИСПОЛЬЗОВАНИЕМ ЯЗЫКОВ ОПИСАНИЯ АППАРАТУРЫ

по курсу: СХЕМОТЕХНИКА

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ гр. № 4143		Е.Д. Тегай
	подпись, дата	инициалы, фамилия

Санкт-Петербург 2024

Цель работы

Разработать проект формирователя импульсов, параметры которых задаются внешним двоичным параллельным кодом в среде программирования Quartus, используя языки описания аппаратуры.

Индивидуальное задание

Задание заключается в формировании импульсов, параметры которых однозначно определяются цифровым управляющим 6-разрядным двоичным кодом. Проект устройства, который нужно реализовать на языке Verilog, должен иметь 1 выход и 8 входов:

1) один вход для подачи тактовых импульсов,

2) один вход для подачи импульса загрузки управляющего кода,

3) 6 входов для подачи внешнего управляющего 6-разрядного двоичного кода.

Номер варианта, а также его содержимое продемонстрировано в таблице 1. Для удобства, оно выделено жёлтым цветом. Следует пояснить, что напротив ячейки «Вар.» находится номер варианта, напротив «$К_1$» - значение, равное тактовой длительности вывода единиц, а «$К_0$» - значение, равное тактовой длительности вывода нулей. В данном случае оно равно N.

Значение управляющего кода может изменяться в произвольное время. Во время активного состояния входа загрузки состояние выхода не регламентируется. По окончании загрузочного импульса должна автоматически инициироваться новая фаза работы устройства, под управлением полученного значения кода (число N). Начало новой фазы работы соответствует первому переднему фронту тактового импульса после снятия импульса загрузки.

Таблица 1

Вар.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$К_1$	N	1	N	N	2	N	3	N	4	N	5	N	6	N	7
$К_0$	N	N	1	2	N	3	N	4	N	5	N	6	N	7	N
Вар.	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
$К_1$	8	N	N	9	N	10	N	11	N	12	N	13	N	14	N
$К_0$	N	8	9	N	10	N	11	N	12	N	13	N	14	N	15

Ход работы

В качестве языка описания аппаратуры был выбран Verilog.

Предыдущая лабораторная работа №4 послужила источником вдохновения для выполнения этой, так как там использовался автомат.

Искомый автомат будет содержать в себе 4 состояния: начальное, состояние, отвечающее за вывод единиц, еще одно состояние, отвечающее за вывод нулей и, наконец, проверочное. Пусть они будут пронумерованы от 0 до 3 включительно, в коем порядке и были оглашены.

- 0 (начальное). С данного состояния и начинается работа автомата. Причём внутренняя логика сразу же начинается с проверочного условия относительно сигнала загрузки. Если же, изначально пребывая в начальном состоянии, автомат не зайдет в это условие, то будет выводиться 0, что логично, ведь пока на сигнале загрузки не будет обнаружена 1, то автомату нечего будет выводить. Внутри условного блока ранее объявленному регистру loaded задаётся значение N. Это то значение, во время существования которого и была обнаружена единица на загрузке. Затем идёт переход в состояние 1, отвечающего за вывод единиц.

- 1 (вывод единиц). Это состояние является частью циклической составляющей автомата. Прежде всего внутри этого состояния выводится единица на выходном сигнале, а затем идёт небольшой проверочный блок. Условие идёт относительно ранее объявленного счётчика для вывода единиц. Это сделано для корректного вывода по количеству единиц. При удачном вхождении в это условие затем идёт ещё одно условие.

Внутри него идёт проверка – равен ли сигнал загрузки 0? Если нет, значит можно спокойно продолжать вывод единиц с теми параметрами, которые были изначально. Это реализуется с помощью инкрементирования счётчика и последующего перехода в это же состояние. Если же сигнал загрузки был равен 1, то идёт переход в состояние проверки, о чём далее.

Если же автомат не зашёл в условие относительно счётчика, значит все единицы согласно параметрам вывелись, поэтому затем нужен переход в состояние, отвечающее за вывод нулей (предварительно обнулив все счётчики).

- 2 (вывод нулей). Это ещё одно состояние, являющееся частью цикличности автомата. Здесь логика в принципе аналогична предыдущему состоянию. Отличие состоит лишь в ином значении в проверочном условии относительно счётчика (проверка звучит как «пока не вывелось N нулей»).

- 3 (проверочное состояние). Собственно, замыкающей частью цикличности автомата является данное состояние. В него автомат переходит только при прерывании. Например, когда шёл вывод тех же единиц при одних параметрах, но обнаружилась ещё одна единица на сигнале загрузки, поэтому необходимо начать вывод заново уже с другими параметрами. Внутри этого состояния идёт обнуление счётчиков и переход в состояние 1.

На рисунке 1 изображён внешний вид искомого автомата.

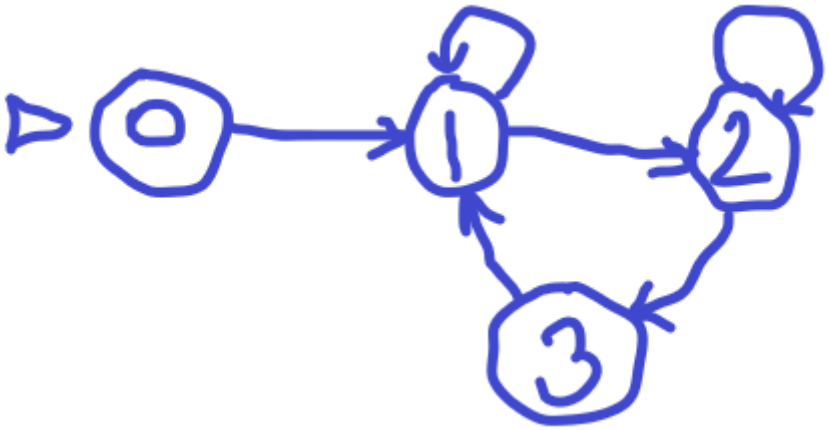

Рисунок 1 – Автомат

Теперь несколько слов про содержимое кода. Модуль состоит из нескольких входов и 1 выхода. В качестве входных сигналов выступают обычные тактовые импульсы, сигнал загрузки (его параметры задаются пользователем), 6-разрядный сигнал N (его значения также задаются

пользователем).

Так как использовался конечный автомат, было принято решение через параметры задать числовое значение каждому из состояний. Поэтому получилось:

IDLE (СОСТОЯНИЕ 0) – 00

COUNT_K1 (СОСТОЯНИЕ 1) – 01

COUNT_K2 (СОСТОЯНИЕ 2) – 10

CHECK (СОСТОЯНИЕ 3) – 11

Затем идёт блок с объявлением регистровых переменных. Раскроем каждое из них:

- State – переменная, отвечающее за статус автомата. То есть именно этой переменной и передаются названия состояний автомата.
- Counter_K0 – счётчик вывода нулей
- Counter_K1 – счётчик вывода единиц
- Loaded – переменная, которой передаётся значение N.

После этого происходит инициализация автомата. То есть задаются начальные значения. Так, изначально автомат пребывает в состоянии IDLE, а счётчики обнулены.

Потом идёт блок, описывающий логику поведения автомата. Следует отметить, что этот блок реализован с помощью case, который лежит внутри always, который «чувствителен» к каждому положительному фронту тактовых импульсов. Это сделано специально, чтобы на каждом положительном фронте автомат проверял, продолжать ли ему вывод цикла или же обновлять параметры и начинать цикл заново.

Листинг программы

```
module laba5 (
    input clk,
    input load,
    input [5:0] N,
    output reg out
);

// Параметры для представления состояний конечного автомата
parameter IDLE = 2'b00;
parameter COUNT_K1 = 2'b01;
parameter COUNT_K0 = 2'b10;
parameter CHECK = 2'b11;

reg [1:0] state;
reg [5:0] counter_K0;
reg [5:0] counter_K1;
reg [5:0] loaded;

// Инициализация автомата
initial begin
    state <= IDLE;
    counter_K1 <= 0;
    counter_K0 <= 0;
end

// Логика конечного автомата
always @(posedge clk) begin
    case (state)
        IDLE: begin
            if (load) begin
                                    loaded <= N;
                state <= COUNT_K1;
            end
        end
        COUNT_K1: begin
          out <= 1;
            if (counter_K1 < 13) begin
```

```
            if (!load) begin
            counter_K1 <= counter_K1 + 1;
            state <= COUNT_K1;
            end
            else state <= CHECK;
          end
          else begin
                              counter_K1 <= 0;
                              counter_K0 <= 0;
            state <= COUNT_K0;
          end
        end
        COUNT_K0: begin
                        out <= 0;
          if (counter_K0 < loaded - 1) begin
            if (!load) begin
            state <= COUNT_K0;
             counter_K0 <= counter_K0 + 1;
            end
            else state <= CHECK;
          end
          else begin
                              counter_K1 <= 0;
                              counter_K0 <= 0;
                              state <= COUNT_K1;
          end
        end
        CHECK: begin
                              counter_K1 <= 0;
                              counter_K0 <= 0;
                              loaded <= N;
            state <= COUNT_K1;
        end
      endcase
end
endmodule
```

ПЛИС

Результат назначения выводов ПЛИС показан на рисунке 2, а на рисунке 3 показана сама ПЛИС.

Node Name	Direction	Location	I/O Bank	VREF Group	I/O Standard
N[5]	Input	PIN_A1	2		3.3-V LVTTL (default)
N[4]	Input	PIN_A2	2		3.3-V LVTTL (default)
N[3]	Input	PIN_A3	2		3.3-V LVTTL (default)
N[2]	Input	PIN_A4	2		3.3-V LVTTL (default)
N[1]	Input	PIN_A5	2		3.3-V LVTTL (default)
N[0]	Input	PIN_A6	2		3.3-V LVTTL (default)
clk	Input	PIN_A7	2		3.3-V LVTTL (default)
load	Input	PIN_A8	2		3.3-V LVTTL (default)
out	Output	PIN_A9	2		3.3-V LVTTL (default)
<<new node>>					

Рисунок 2 – Назначение выводов ПЛИС

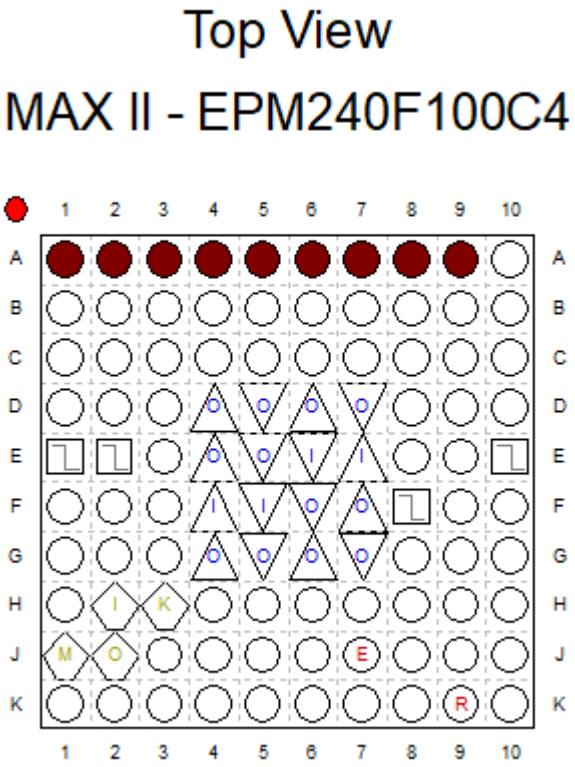

Рисунок 3 - ПЛИС

Временная диаграмма

Результаты функциональной и временной симуляций при малых значениях N (0 – 19) продемонстрированы на рисунках 4 – 9.

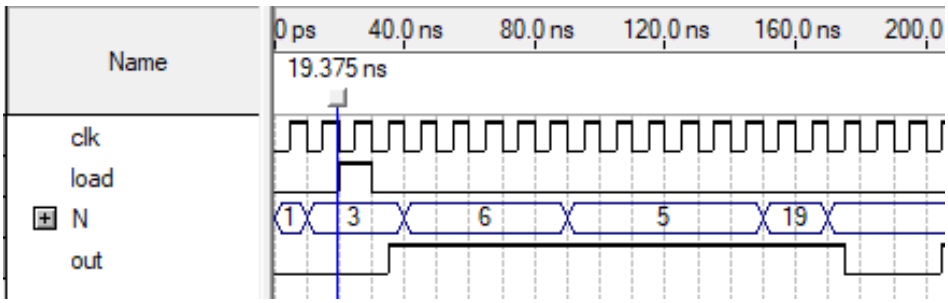

Рисунок 4 – Функциональная симуляция при малых значениях (начало)

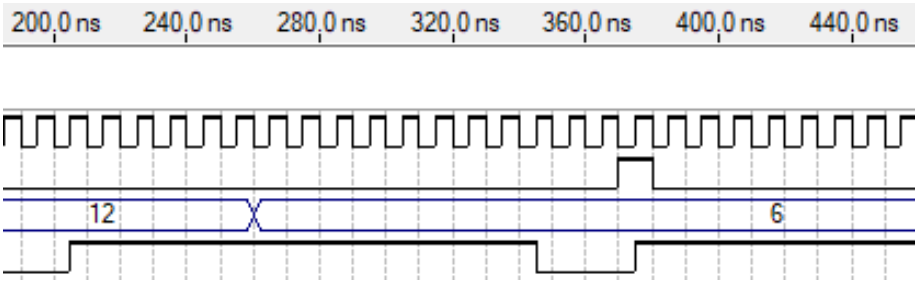

Рисунок 5 – Функциональная симуляция при малых значениях (середина)

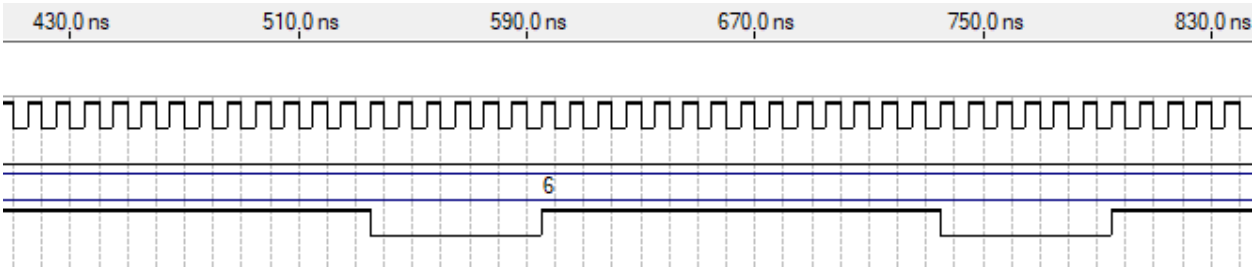

Рисунок 6 – Функциональная симуляция при малых значениях (конец)

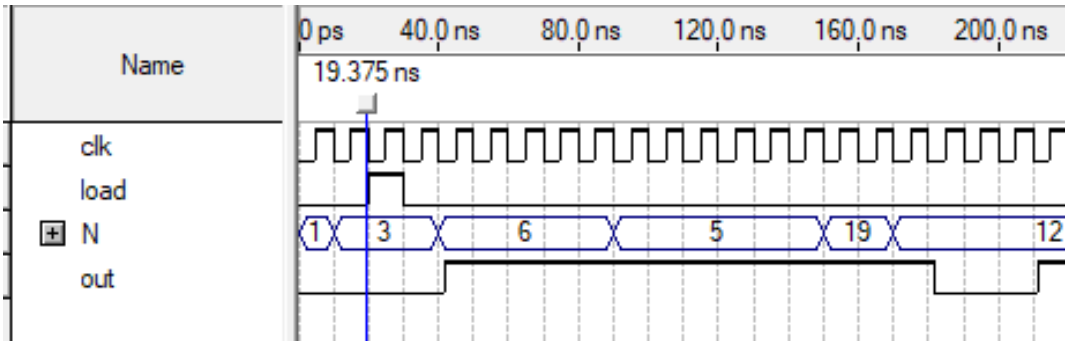

Рисунок 7 – Временная симуляция при малых значениях (начало)

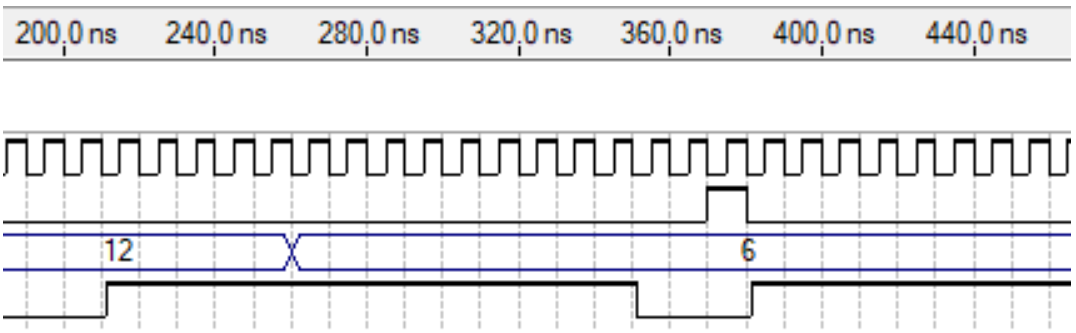

Рисунок 8 – Временная симуляция при малых значениях (середина)

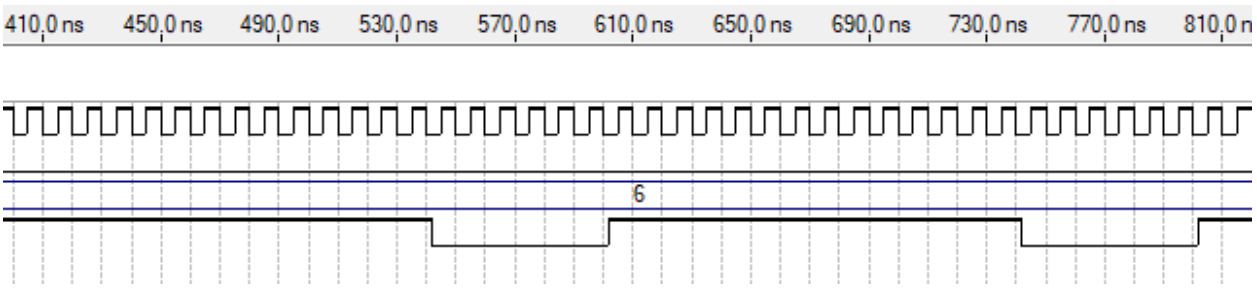

Рисунок 9 – Временная симуляция при малых значениях (конец)

Результаты функциональной и временной симуляций при средних значениях N (20 – 30) продемонстрированы на рисунках 10 – 15.

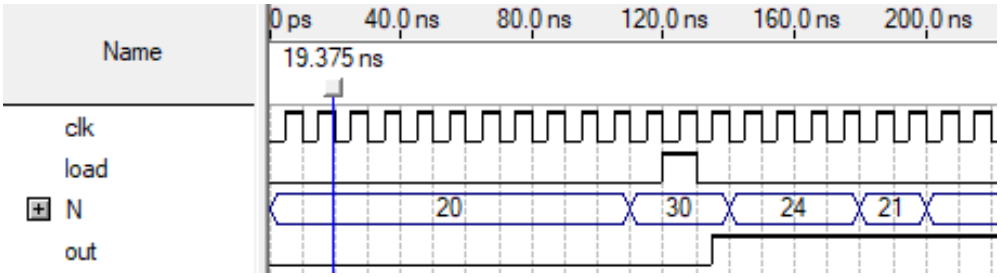

Рисунок 10 – Функциональная симуляция при средних значениях (начало)

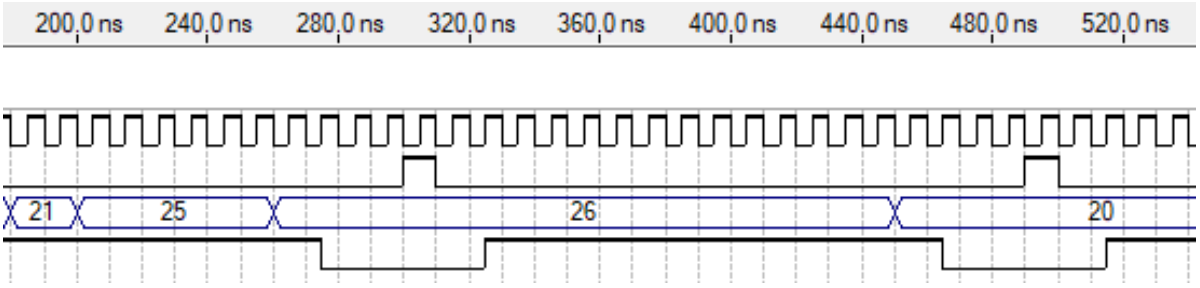

Рисунок 11 – Функциональная симуляция при средних значениях (середина)

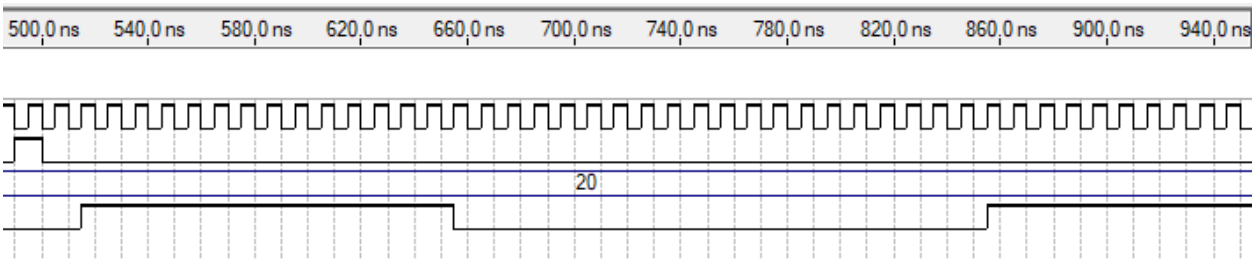

Рисунок 12 – Функциональная симуляция при средних значениях (конец)

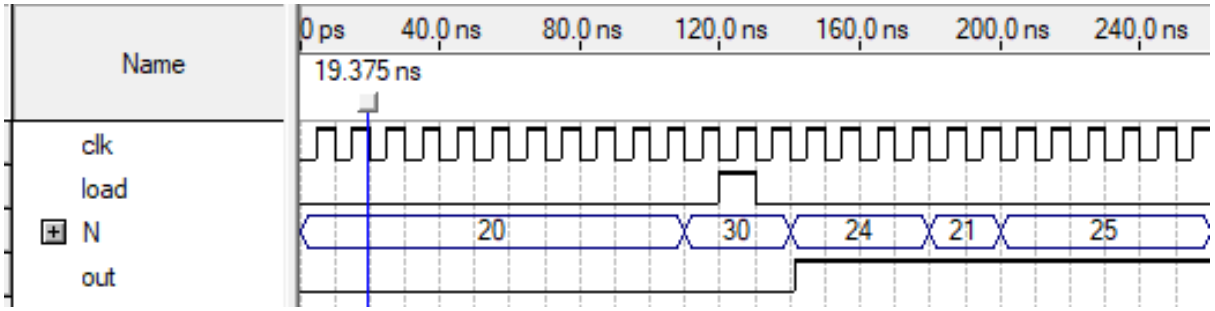

Рисунок 13 – Временная симуляция при средних значениях (начало)

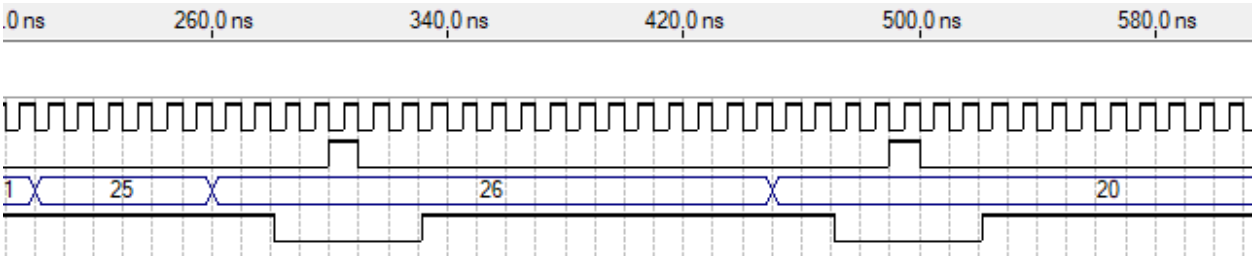

Рисунок 14 – Временная симуляция при средних значениях (середина)

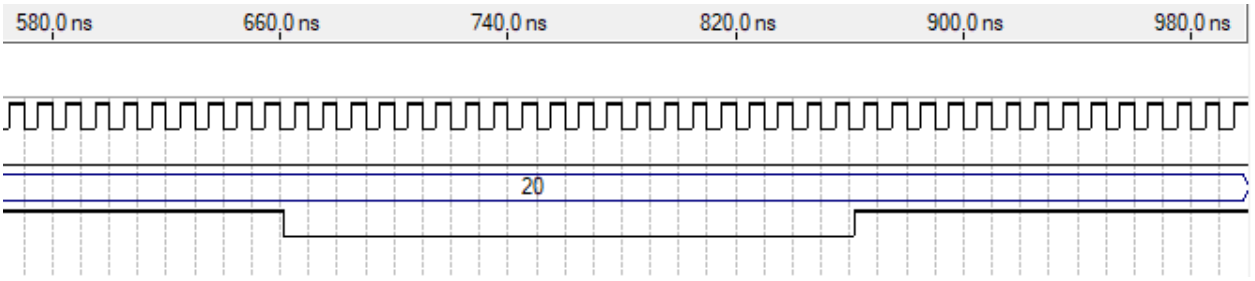

Рисунок 15 – Временная симуляция при средних значениях (конец)

Наконец, результаты функциональной и временной симуляций с максимальным значением N продемонстрированы на рисунках 16 – 21.

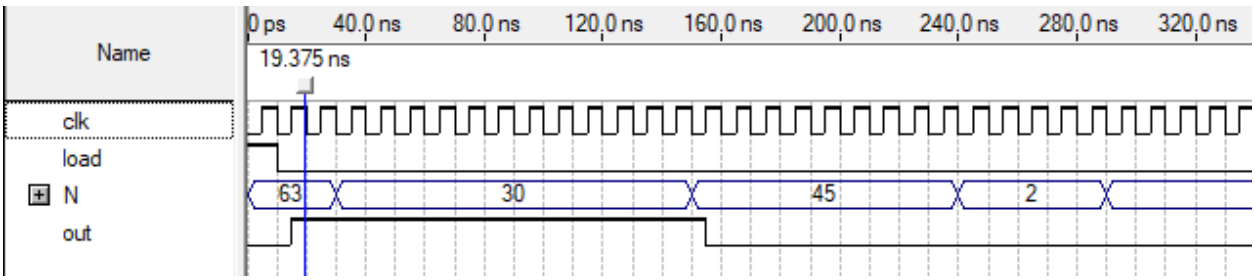

Рисунок 16 – Функциональная симуляция при максимальных значениях (начало)

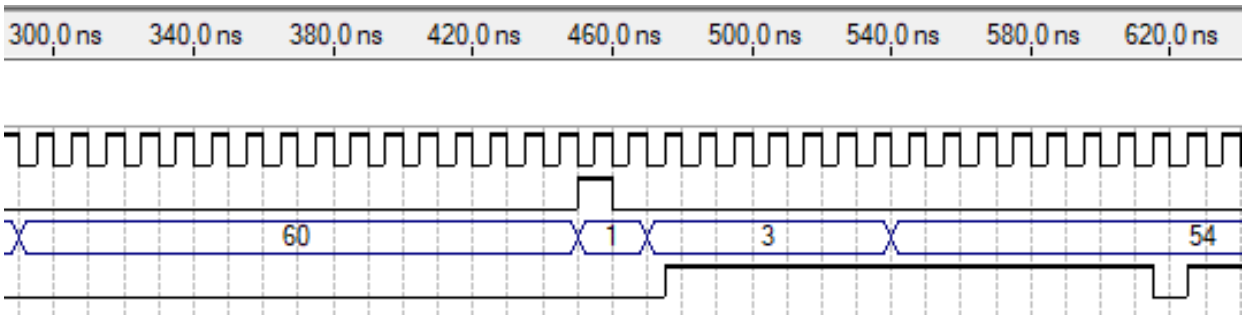

Рисунок 17 – Функциональная симуляция при максимальных значениях (середина)

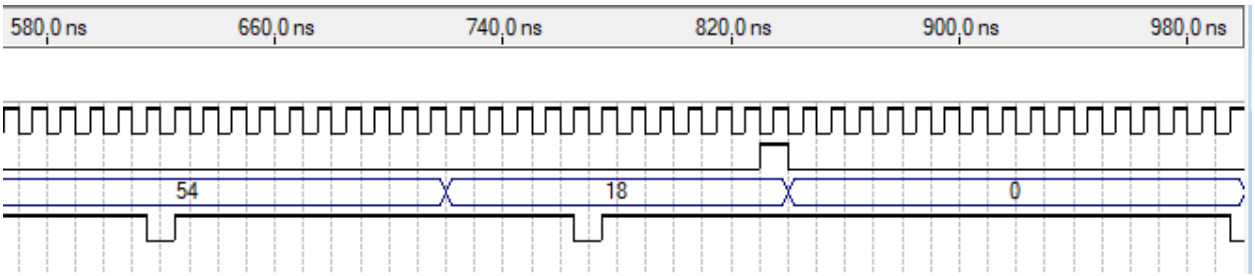

Рисунок 18 – Функциональная симуляция при максимальных значениях (конец)

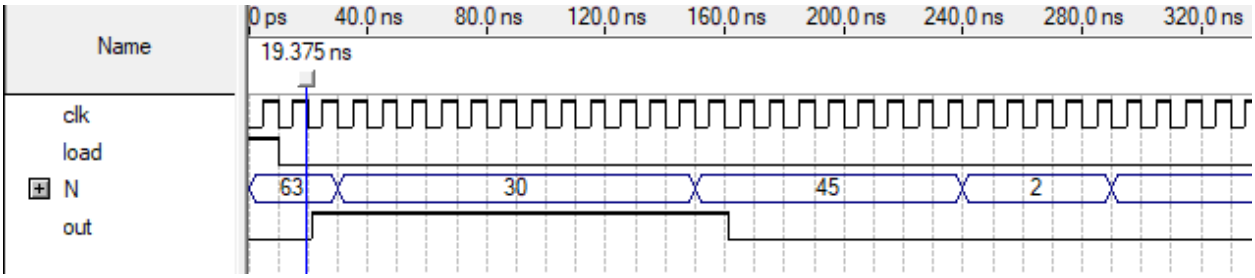

Рисунок 19 – Временная симуляция при максимальных значениях (начало)

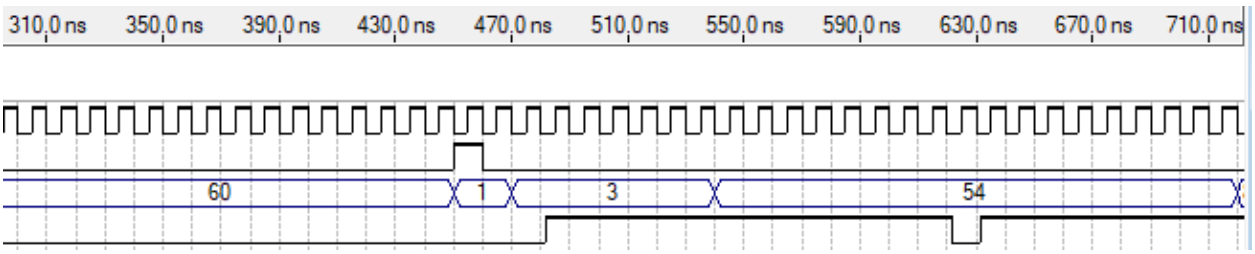

Рисунок 20 – Временная симуляция при максимальных значениях (середина)

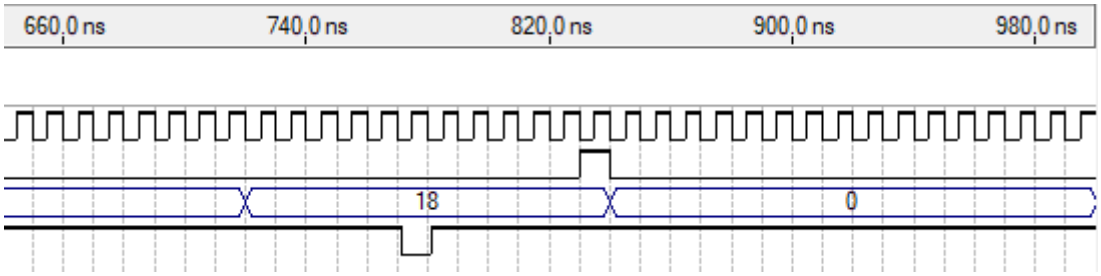

Рисунок 21 – Временная симуляция при максимальных значениях (конец)

Выводы

В данной лабораторной работе был разработан проект формирователя импульсов, параметры которых задаются внешним двоичным параллельным кодом в среде программирования Quartus, используя языки описания аппаратуры.

Список используемых источников

1. Проектирование встраиваемых систем на ПЛИС. / З.Наваби; перев. с англ.В.В. Соловьева. – М.: ДМК Пресс, 2016. - 464 с.

2. Проектирование цифровых устройств на ПЛИС: учеб. пособие / И.В. Ушенина. - СПб: Лань, 2022. - 408 с.

3. Цифровая схемотехника и архитектура компьютера / Д.М. Харрис, С.Л.Харрис; пер. с англ. ImaginationTechnologies. – М.: ДМК Пресс, 2018. - 792 с.

4. Учебно-методические материалы к выполнению лабораторной работы №5 по дисциплине «Схемотехника» (2-й семестр изучения дисциплины) // Жаринов. О.О: [Электронный ресурс] // Санкт-Петербургский государственный университет аэрокосмического приборостроения. URL.: https://pro.guap.ru/inside/student/tasks/4f645754182c1092dee745c6a09dc3fe/dow

nload. (Дата обращения: 29.03.24).

5. Лекция №3 от 11 марта 2024 года по дисциплине «Схемотехника» (2-й семестр изучения дисциплины) // Жаринов. О.О: [Электронный ресурс] // Санкт-Петербургский государственный университет аэрокосмического приборостроения. URL.: https://bbb1.guap.ru/playback/presentation/2.3/4e99f54650dea30ef1dc263d08fafdd7f0c36944-1710157682528. (Дата обращения: 29.03.24).

6. Отчёт о выполнении лабораторной работы №4 по дисциплине «Схемотехника» (2-й семестр изучения дисциплины) // Тегай. Е.Д: [Электронный ресурс] // Санкт-Петербургский государственный университет аэрокосмического приборостроения. URL.: https://pro.guap.ru/inside/student/reports/3910573/download. (Дата обращения: 29.03.24).