

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное автономное образовательное учреждение высшего образования  
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА 44

КУРСОВОЙ ПРОЕКТ  
ЗАЩИЩЕНА С ОЦЕНКОЙ  
РУКОВОДИТЕЛЬ

старший преподаватель  
должность, уч. степень, звание

\_\_\_\_\_  
подпись, дата

А.Н. Долидзе  
инициалы, фамилия

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА  
К КУРСОВОМУ ПРОЕКТУ

по дисциплине: СХЕМОТЕХНИКА

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ гр. № \_\_\_\_\_

\_\_\_\_\_  
подпись, дата

\_\_\_\_\_  
инициалы, фамилия

Санкт-Петербург 2024

## Проектное задание

Разработать электронное устройство в соответствии с предложенной схемой и исходными данными, которое обеспечило бы заданную точность и качество работы. Работоспособность устройства обязательно подтверждается моделированием с применением САПР Quartus.

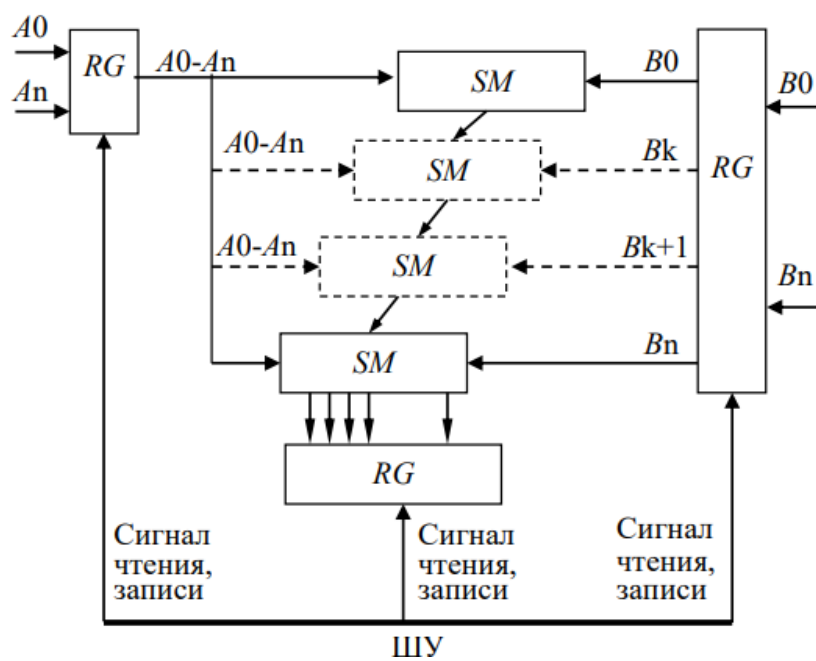
Разработка устройства включает в себя следующие этапы:

- составление и обоснование выбора функциональной схемы;
- разработку проектируемой схемы в САПР Quartus с использованием стандартных блоков, предоставляемых САПР (И, НЕ, дешифратор, триггер и т.д.) и описание ее работы с использованием временных диаграмм;
- разработку проектируемой схемы в САПР Quartus с использованием языка описания аппаратуры SystemVerilog и описание ее работы с использованием временных диаграмм;
- анализ двух подходов к разработке проектируемой схемы;
- обоснование и выбор FPGA на базе которого будет синтезироваться разрабатываемая схема;
- моделированием с применением САПР Quartus.
- назначение входов и выходов проектируемой схемы на выбранном FPGA.

Согласно варианту (схема 6, вариант 2), устройство является комбинационным блоком умножения разрядностью 16, принимающего на вход двоичные числа. Предложенная схема устройства представлена на рисунке 1. Необходимо так выбрать разрядность сумматора и следующего за ним регистра, чтобы не было переполнения.

## Схема 6

### Комбинационный блок умножения



## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	5
1. РАЗРАБОТКА ФУНКЦИОНАЛЬНОЙ СХЕМЫ .....	6
1.1. ВЫБОР ФУНКЦИОНАЛЬНОЙ СХЕМЫ.....	6
1.2. СОСТАВЛЕНИЕ СХЕМЫ УСТРОЙСТВА С ИСПОЛЬЗОВАНИЕМ СТАНДАРТНЫХ БЛОКОВ.....	6
1.3. СОСТАВЛЕНИЕ СХЕМЫ УСТРОЙСТВА С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА ОПИСАНИЯ АППАРАТУРЫ SYSTEM VERILOG.....	14
2. ВЫБОР FPGA .....	79
ЗАКЛЮЧЕНИЕ.....	83
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	84

## **ВВЕДЕНИЕ**

Целью курсового проектирования является освоение методов расчета, схемотехнического проектирования и конструирования элементов и блоков ЦВМ на базе программируемых логических интегральных схема (ПЛИС, далее FPGA).

В ходе выполнения курсового проекта должны быть получены схема устройства и программный код на языке описания аппаратуры System Verilog, соответствующие заданному в проектном задании функционалу. Устройство представляет собой комбинационный блок умножения восьмибитных двоичных чисел.

## 1. РАЗРАБОТКА ФУНКЦИОНАЛЬНОЙ СХЕМЫ

### 1.1. ВЫБОР ФУНКЦИОНАЛЬНОЙ СХЕМЫ

Реализация данной схемы была выполнена с помощью регистров разрядностью 16 и 32, сумматоров разрядностью 8, блоков «И». Управление блоком (считывание множимых и вывод результата) осуществляется с помощью шины управления.

### 1.2. СОСТАВЛЕНИЕ СХЕМЫ УСТРОЙСТВА С ИСПОЛЬЗОВАНИЕМ СТАНДАРТНЫХ БЛОКОВ

В ходе разработки схемы устройства были использован стандартный модуль *AND*, а также модуль регистра (*lpm\_dff0* и *lpm\_dff1*) и модуль сумматора (*8fadd*). На рисунках 2-6 представлен результат моделирования схемы в графическом редакторе среды Quartus.

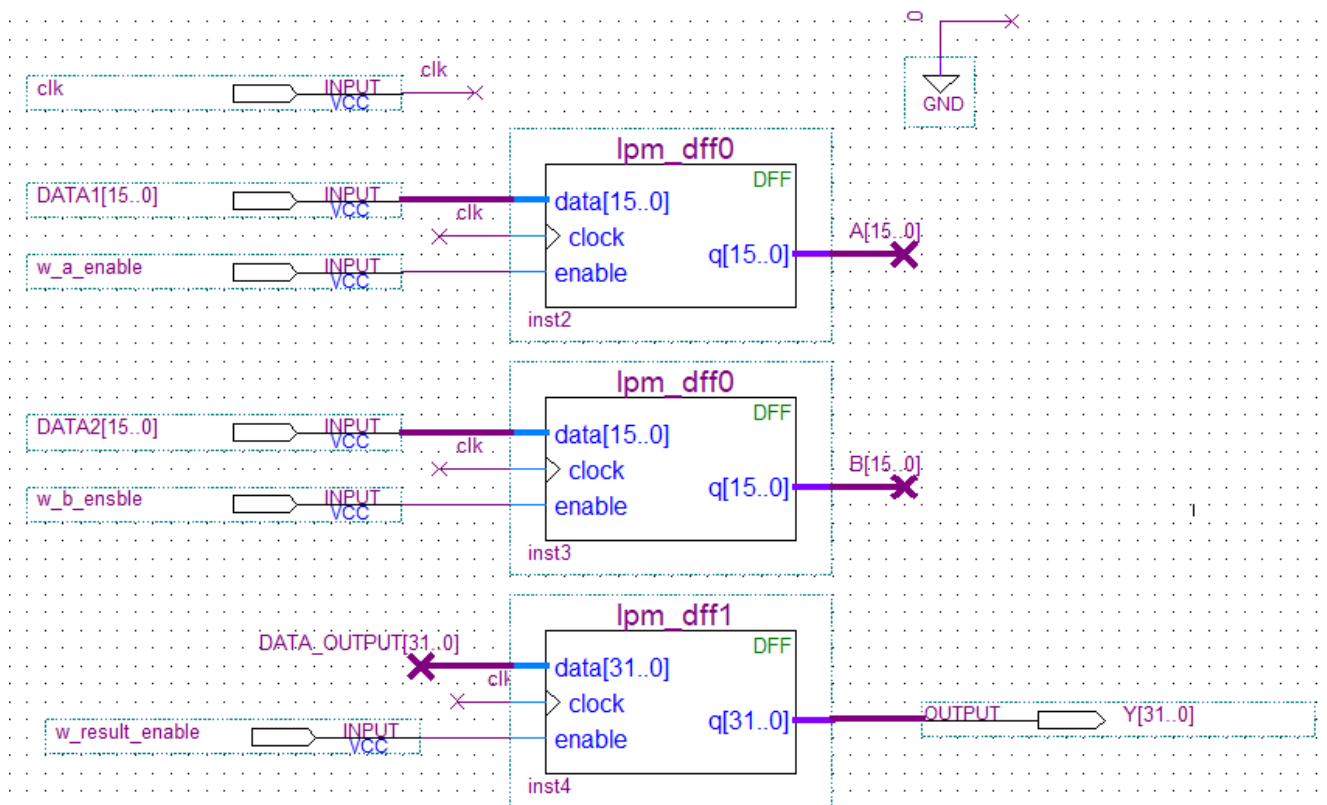


Рисунок 2 – Регистры для входных и выходных данных

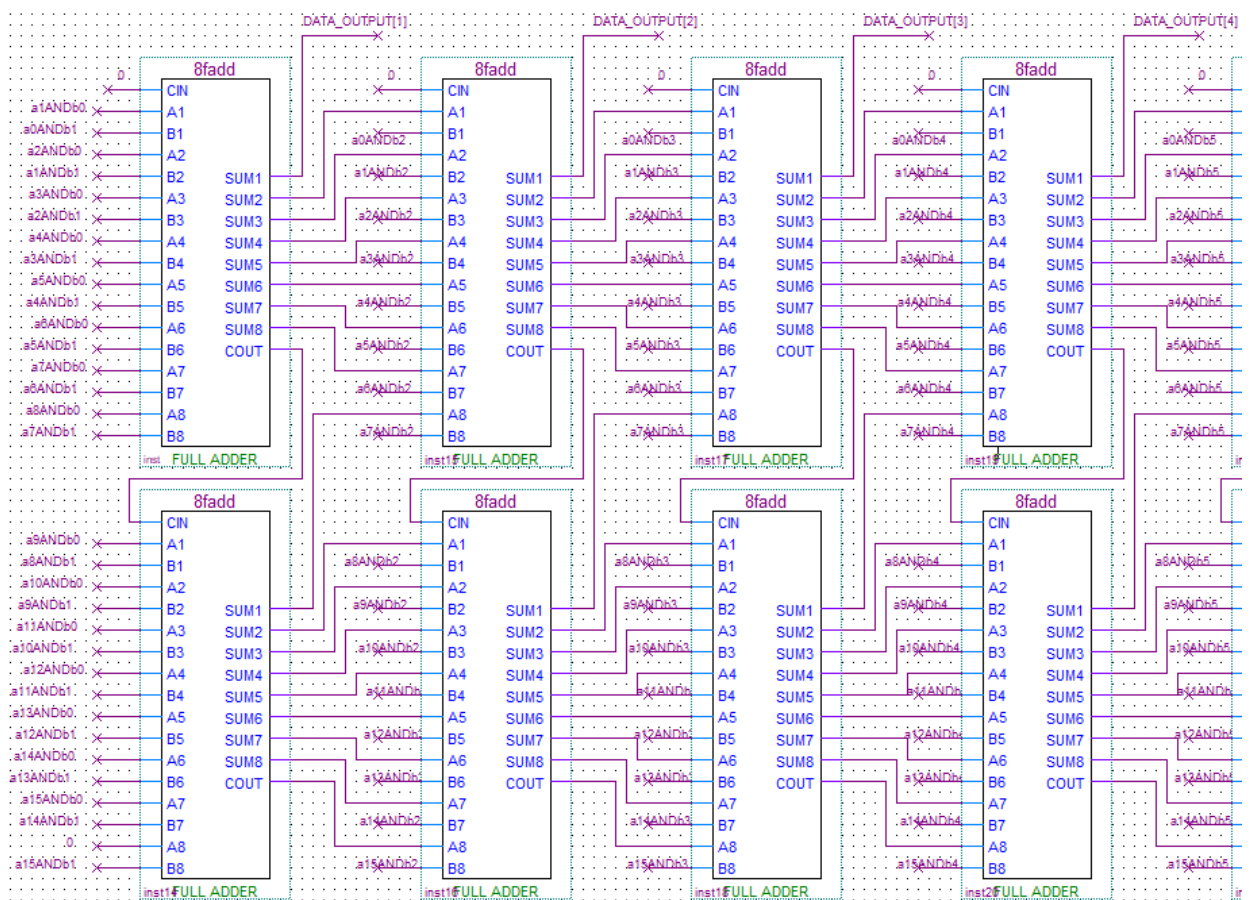


Рисунок 3 – Последовательность из сумматоров (разряды 1-4)

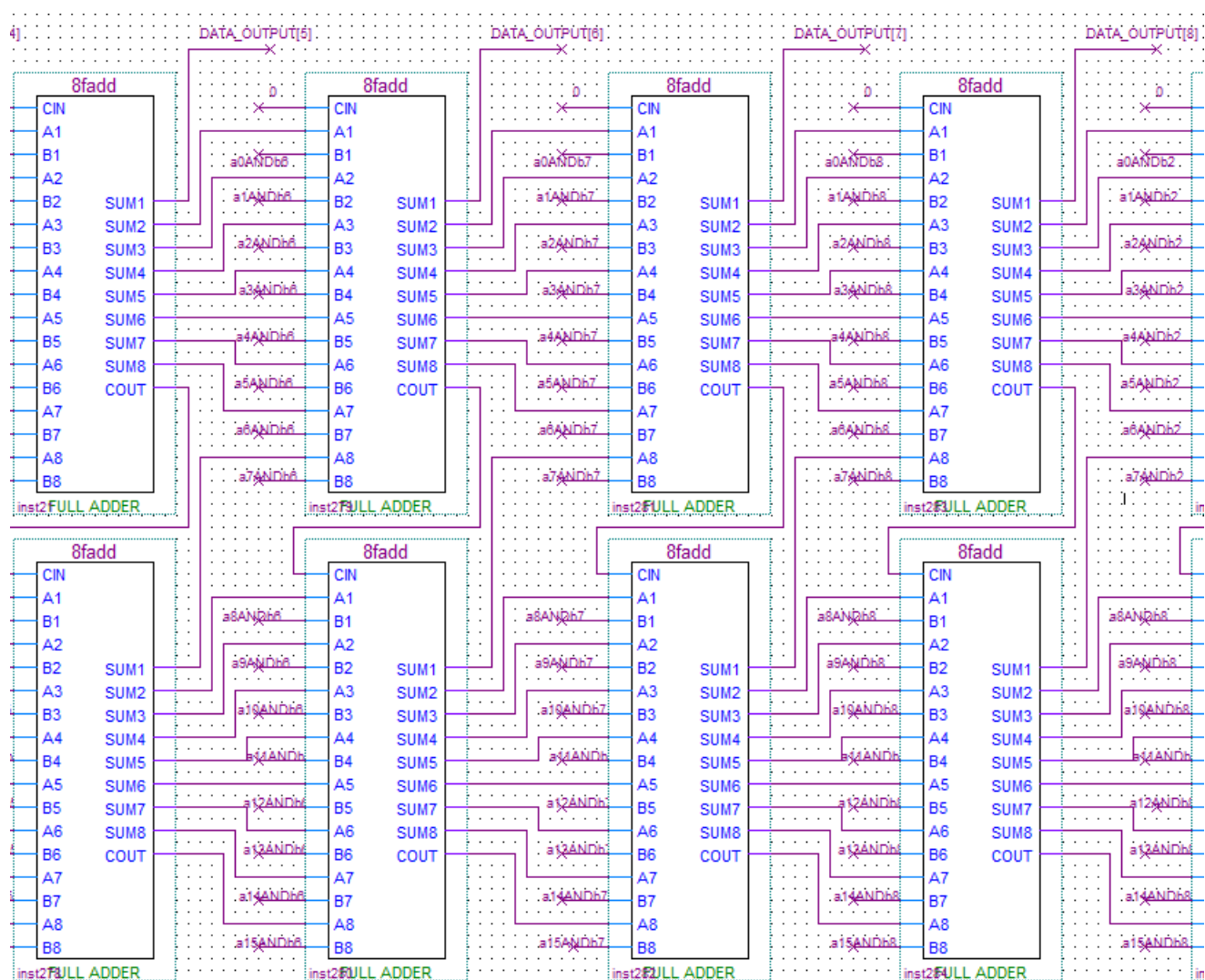


Рисунок 4 – Последовательность из сумматоров (разряды 5-8)



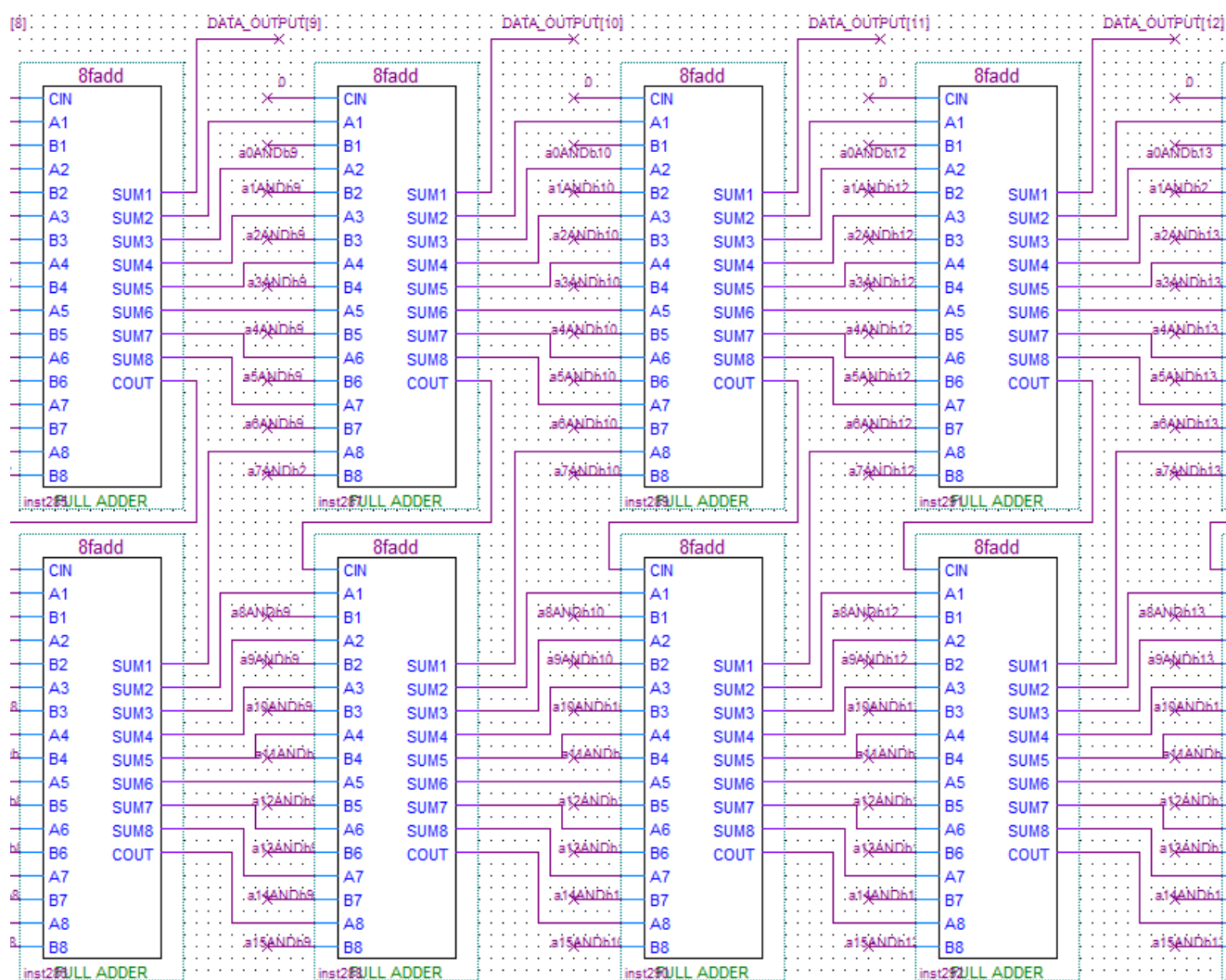


Рисунок 5 – Последовательность из сумматоров (разряды 9-12)

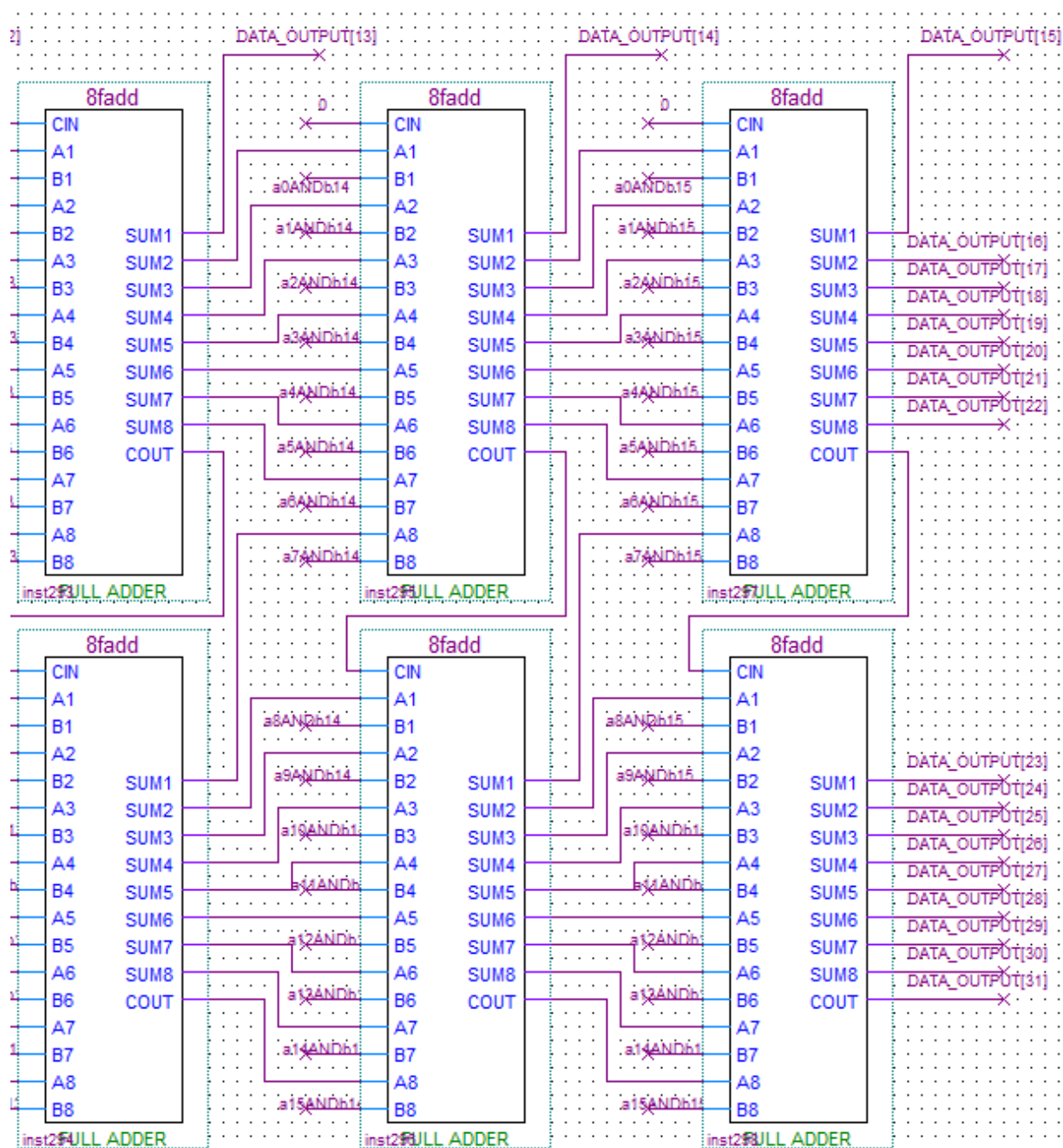


Рисунок 6 – Последовательность из сумматоров (разряды 13- 31)

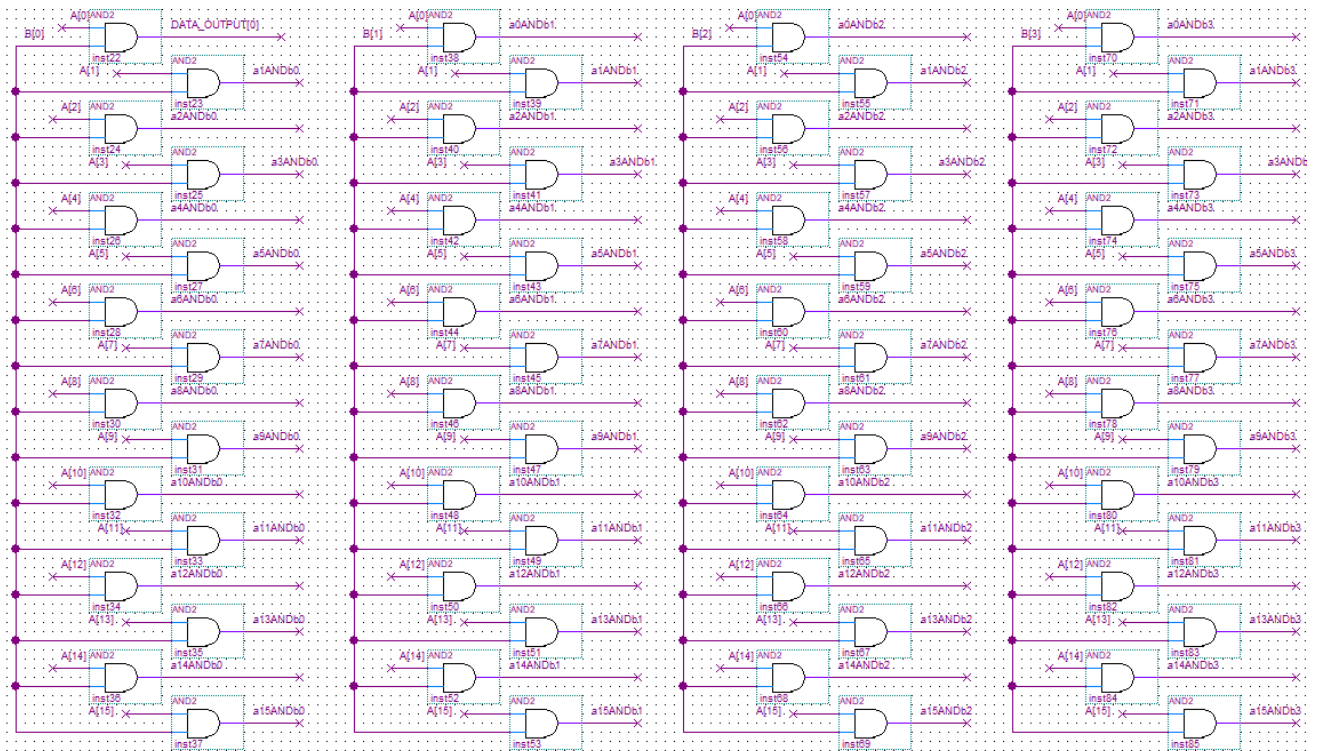


Рисунок 7 – Частные произведения разрядов 0-3 числа  $b$  на число  $a$

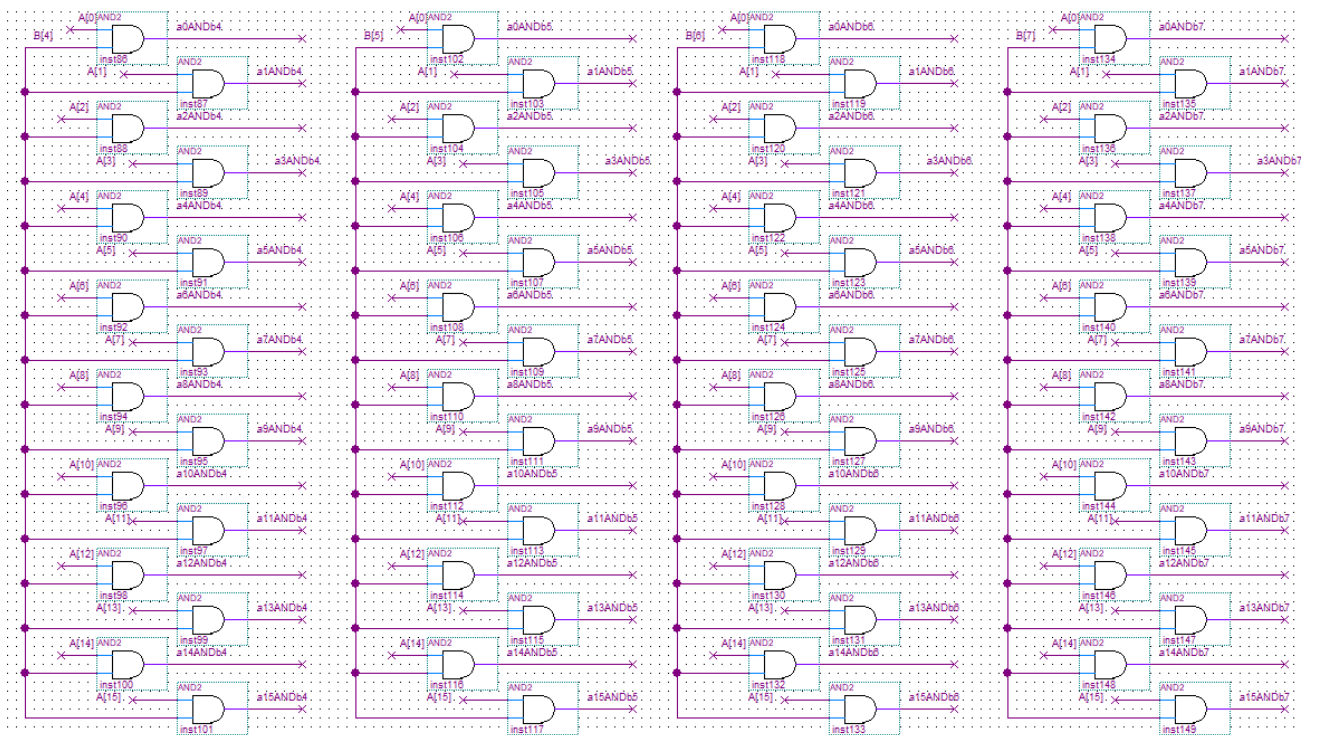


Рисунок 8 – Частные произведения разрядов 4-7 числа  $b$  на число  $a$

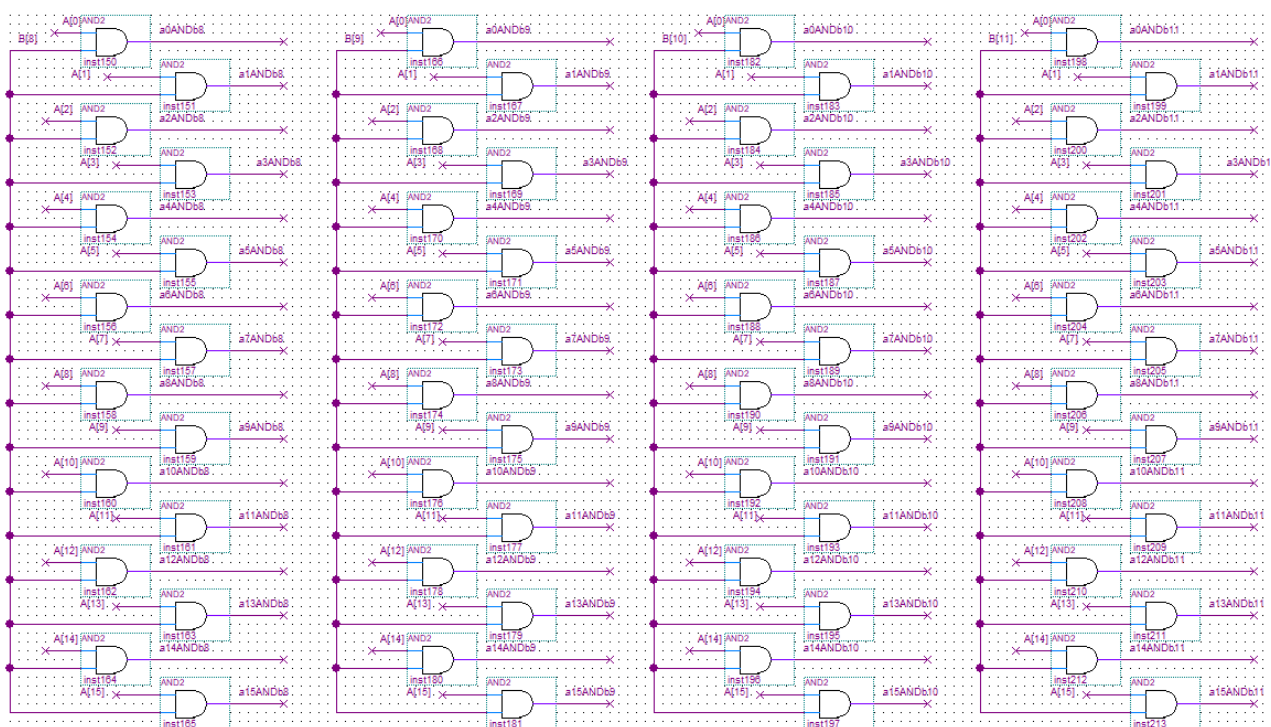


Рисунок 9 – Частные произведения разрядов 8-11 числа  $b$  на число  $a$

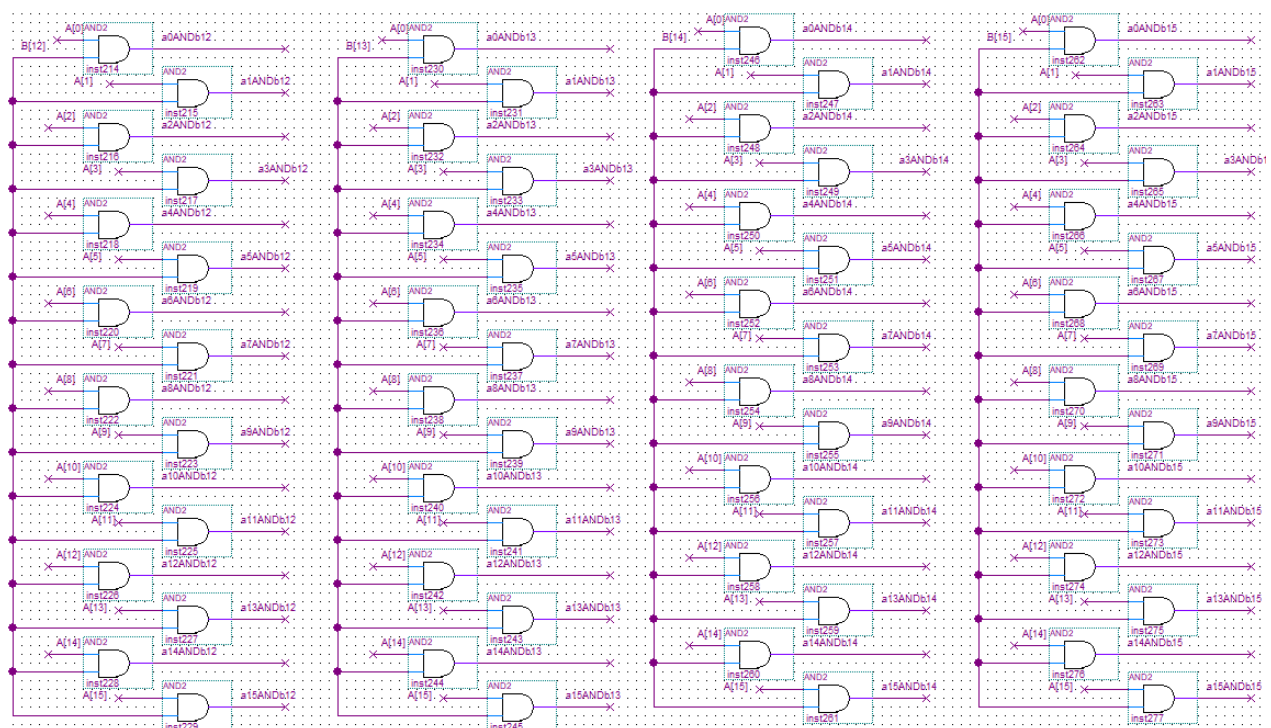


Рисунок 10 – Частные произведения разрядов 12-16 числа  $b$  на число  $a$

На рисунке 11 представлен результат симулирования схемы в виде временной диаграммы.

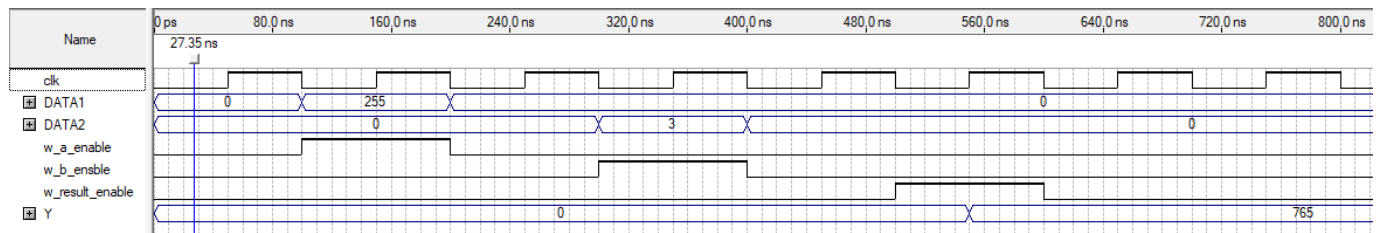


Рисунок 11 – Временная диаграмма

### 1.3. СОСТАВЛЕНИЕ СХЕМЫ УСТРОЙСТВА С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА ОПИСАНИЯ АППАРАТУРЫ SYSTEM VERILOG

Программа состоит из одного модуля: основной программы. Код программы представлен в листинге 1.

Листинг 1 – Модуль блока умножения

```
module kurs_6var(
    w_a_enable,
    w_b_ensble,
    clk,
    w_result_enable,
    DATA1,
    DATA2,
    Y
);

input w_a_enable;
input w_b_ensble;
input clk;
input w_result_enable;
input [15:0] DATA1;
input [15:0] DATA2;
output[31:0] Y;

wire 0;
wire [15:0] A;
wire a0ANDb1;
wire a0ANDb10;
```

```
wire a0ANDb11;
wire a0ANDb12;
wire a0ANDb13;
wire a0ANDb14;
wire a0ANDb15;
wire a0ANDb2;
wire a0ANDb3;
wire a0ANDb4;
wire a0ANDb5;
wire a0ANDb6;
wire a0ANDb7;
wire a0ANDb8;
wire a0ANDb9;
wire a10ANDb0;
wire a10ANDb1;
wire a10ANDb10;
wire a10ANDb11;
wire a10ANDb12;
wire a10ANDb13;
wire a10ANDb14;
wire a10ANDb15;
wire a10ANDb2;
wire a10ANDb3;
wire a10ANDb4;
wire a10ANDb5;
wire a10ANDb6;
wire a10ANDb7;
wire a10ANDb8;
wire a10ANDb9;
wire a11ANDb0;
```

```
wire a11ANDb1;  
wire a11ANDb10;  
wire a11ANDb11;  
wire a11ANDb12;  
wire a11ANDb13;  
wire a11ANDb14;  
wire a11ANDb15;  
wire a11ANDb2;  
wire a11ANDb3;  
wire a11ANDb4;  
wire a11ANDb5;  
wire a11ANDb6;  
wire a11ANDb7;  
wire a11ANDb8;  
wire a11ANDb9;  
wire a12ANDb0;  
wire a12ANDb1;  
wire a12ANDb10;  
wire a12ANDb11;  
wire a12ANDb12;  
wire a12ANDb13;  
wire a12ANDb14;  
wire a12ANDb15;  
wire a12ANDb2;  
wire a12ANDb3;  
wire a12ANDb4;  
wire a12ANDb5;  
wire a12ANDb6;  
wire a12ANDb7;  
wire a12ANDb8;
```



```
wire a12ANDb9;
wire a13ANDb0;
wire a13ANDb1;
wire a13ANDb10;
wire a13ANDb11;
wire a13ANDb12;
wire a13ANDb13;
wire a13ANDb14;
wire a13ANDb15;
wire a13ANDb2;
wire a13ANDb3;
wire a13ANDb4;
wire a13ANDb5;
wire a13ANDb6;
wire a13ANDb7;
wire a13ANDb8;
wire a13ANDb9;
wire a14ANDb0;
wire a14ANDb1;
wire a14ANDb10;
wire a14ANDb11;
wire a14ANDb12;
wire a14ANDb13;
wire a14ANDb14;
wire a14ANDb15;
wire a14ANDb2;
wire a14ANDb3;
wire a14ANDb4;
wire a14ANDb5;
wire a14ANDb6;
```

```
wire a14ANDb7;  
wire a14ANDb8;  
wire a14ANDb9;  
wire a15ANDb0;  
wire a15ANDb1;  
wire a15ANDb10;  
wire a15ANDb11;  
wire a15ANDb12;  
wire a15ANDb13;  
wire a15ANDb14;  
wire a15ANDb15;  
wire a15ANDb2;  
wire a15ANDb3;  
wire a15ANDb4;  
wire a15ANDb5;  
wire a15ANDb6;  
wire a15ANDb7;  
wire a15ANDb8;  
wire a15ANDb9;  
wire a1ANDb0;  
wire a1ANDb1;  
wire a1ANDb10;  
wire a1ANDb11;  
wire a1ANDb12;  
wire a1ANDb13;  
wire a1ANDb14;  
wire a1ANDb15;  
wire a1ANDb2;  
wire a1ANDb3;  
wire a1ANDb4;
```

```
wire a1ANDb5;
wire a1ANDb6;
wire a1ANDb7;
wire a1ANDb8;
wire a1ANDb9;
wire a2ANDb0;
wire a2ANDb1;
wire a2ANDb10;
wire a2ANDb11;
wire a2ANDb12;
wire a2ANDb13;
wire a2ANDb14;
wire a2ANDb15;
wire a2ANDb2;
wire a2ANDb3;
wire a2ANDb4;
wire a2ANDb5;
wire a2ANDb6;
wire a2ANDb7;
wire a2ANDb8;
wire a2ANDb9;
wire a3ANDb0;
wire a3ANDb1;
wire a3ANDb10;
wire a3ANDb11;
wire a3ANDb12;
wire a3ANDb13;
wire a3ANDb14;
wire a3ANDb15;
wire a3ANDb2;
```

wire a3ANDb3;  
wire a3ANDb4;  
wire a3ANDb5;  
wire a3ANDb6;  
wire a3ANDb7;  
wire a3ANDb8;  
wire a3ANDb9;  
wire a4ANDb0;  
wire a4ANDb1;  
wire a4ANDb10;  
wire a4ANDb11;  
wire a4ANDb12;  
wire a4ANDb13;  
wire a4ANDb14;  
wire a4ANDb15;  
wire a4ANDb2;  
wire a4ANDb3;  
wire a4ANDb4;  
wire a4ANDb5;  
wire a4ANDb6;  
wire a4ANDb7;  
wire a4ANDb8;  
wire a4ANDb9;  
wire a5ANDb0;  
wire a5ANDb1;  
wire a5ANDb10;  
wire a5ANDb11;  
wire a5ANDb12;  
wire a5ANDb13;  
wire a5ANDb14;

wire a5ANDb15;  
wire a5ANDb2;  
wire a5ANDb3;  
wire a5ANDb4;  
wire a5ANDb5;  
wire a5ANDb6;  
wire a5ANDb7;  
wire a5ANDb8;  
wire a5ANDb9;  
wire a6ANDb0;  
wire a6ANDb1;  
wire a6ANDb10;  
wire a6ANDb11;  
wire a6ANDb12;  
wire a6ANDb13;  
wire a6ANDb14;  
wire a6ANDb15;  
wire a6ANDb2;  
wire a6ANDb3;  
wire a6ANDb4;  
wire a6ANDb5;  
wire a6ANDb6;  
wire a6ANDb7;  
wire a6ANDb8;  
wire a6ANDb9;  
wire a7ANDb0;  
wire a7ANDb1;  
wire a7ANDb10;  
wire a7ANDb11;  
wire a7ANDb12;

wire a7ANDb13;  
wire a7ANDb14;  
wire a7ANDb15;  
wire a7ANDb2;  
wire a7ANDb3;  
wire a7ANDb4;  
wire a7ANDb5;  
wire a7ANDb6;  
wire a7ANDb7;  
wire a7ANDb8;  
wire a7ANDb9;  
wire a8ANDb0;  
wire a8ANDb1;  
wire a8ANDb10;  
wire a8ANDb11;  
wire a8ANDb12;  
wire a8ANDb13;  
wire a8ANDb14;  
wire a8ANDb15;  
wire a8ANDb2;  
wire a8ANDb3;  
wire a8ANDb4;  
wire a8ANDb5;  
wire a8ANDb6;  
wire a8ANDb7;  
wire a8ANDb8;  
wire a8ANDb9;  
wire a9ANDb0;  
wire a9ANDb1;  
wire a9ANDb10;

```
wire a9ANDb11;
wire a9ANDb12;
wire a9ANDb13;
wire a9ANDb14;
wire a9ANDb15;
wire a9ANDb2;
wire a9ANDb3;
wire a9ANDb4;
wire a9ANDb5;
wire a9ANDb6;
wire a9ANDb7;
wire a9ANDb8;
wire a9ANDb9;
wire [15:0] B;
wire [31:0] DATA_OUTPUT;
wire SYNTHESIZED_WIRE_0;
wire SYNTHESIZED_WIRE_1;
wire SYNTHESIZED_WIRE_2;
wire SYNTHESIZED_WIRE_3;
wire SYNTHESIZED_WIRE_4;
wire SYNTHESIZED_WIRE_5;
wire SYNTHESIZED_WIRE_6;
wire SYNTHESIZED_WIRE_7;
wire SYNTHESIZED_WIRE_8;
wire SYNTHESIZED_WIRE_9;
wire SYNTHESIZED_WIRE_10;
wire SYNTHESIZED_WIRE_11;
wire SYNTHESIZED_WIRE_12;
wire SYNTHESIZED_WIRE_13;
wire SYNTHESIZED_WIRE_14;
```

```
wire SYNTHESIZED_WIRE_15;
wire SYNTHESIZED_WIRE_16;
wire SYNTHESIZED_WIRE_17;
wire SYNTHESIZED_WIRE_18;
wire SYNTHESIZED_WIRE_19;
wire SYNTHESIZED_WIRE_20;
wire SYNTHESIZED_WIRE_21;
wire SYNTHESIZED_WIRE_22;
wire SYNTHESIZED_WIRE_23;
wire SYNTHESIZED_WIRE_24;
wire SYNTHESIZED_WIRE_25;
wire SYNTHESIZED_WIRE_26;
wire SYNTHESIZED_WIRE_27;
wire SYNTHESIZED_WIRE_28;
wire SYNTHESIZED_WIRE_29;
wire SYNTHESIZED_WIRE_30;
wire SYNTHESIZED_WIRE_31;
wire SYNTHESIZED_WIRE_32;
wire SYNTHESIZED_WIRE_33;
wire SYNTHESIZED_WIRE_34;
wire SYNTHESIZED_WIRE_35;
wire SYNTHESIZED_WIRE_36;
wire SYNTHESIZED_WIRE_37;
wire SYNTHESIZED_WIRE_38;
wire SYNTHESIZED_WIRE_39;
wire SYNTHESIZED_WIRE_40;
wire SYNTHESIZED_WIRE_41;
wire SYNTHESIZED_WIRE_42;
wire SYNTHESIZED_WIRE_43;
wire SYNTHESIZED_WIRE_44;
```



```
wire SYNTHESIZED_WIRE_45;
wire SYNTHESIZED_WIRE_46;
wire SYNTHESIZED_WIRE_47;
wire SYNTHESIZED_WIRE_48;
wire SYNTHESIZED_WIRE_49;
wire SYNTHESIZED_WIRE_50;
wire SYNTHESIZED_WIRE_51;
wire SYNTHESIZED_WIRE_52;
wire SYNTHESIZED_WIRE_53;
wire SYNTHESIZED_WIRE_54;
wire SYNTHESIZED_WIRE_55;
wire SYNTHESIZED_WIRE_56;
wire SYNTHESIZED_WIRE_57;
wire SYNTHESIZED_WIRE_58;
wire SYNTHESIZED_WIRE_59;
wire SYNTHESIZED_WIRE_60;
wire SYNTHESIZED_WIRE_61;
wire SYNTHESIZED_WIRE_62;
wire SYNTHESIZED_WIRE_63;
wire SYNTHESIZED_WIRE_64;
wire SYNTHESIZED_WIRE_65;
wire SYNTHESIZED_WIRE_66;
wire SYNTHESIZED_WIRE_67;
wire SYNTHESIZED_WIRE_68;
wire SYNTHESIZED_WIRE_69;
wire SYNTHESIZED_WIRE_70;
wire SYNTHESIZED_WIRE_71;
wire SYNTHESIZED_WIRE_72;
wire SYNTHESIZED_WIRE_73;
wire SYNTHESIZED_WIRE_74;
```

```
wire SYNTHESIZED_WIRE_75;
wire SYNTHESIZED_WIRE_76;
wire SYNTHESIZED_WIRE_77;
wire SYNTHESIZED_WIRE_78;
wire SYNTHESIZED_WIRE_79;
wire SYNTHESIZED_WIRE_80;
wire SYNTHESIZED_WIRE_81;
wire SYNTHESIZED_WIRE_82;
wire SYNTHESIZED_WIRE_83;
wire SYNTHESIZED_WIRE_84;
wire SYNTHESIZED_WIRE_85;
wire SYNTHESIZED_WIRE_86;
wire SYNTHESIZED_WIRE_87;
wire SYNTHESIZED_WIRE_88;
wire SYNTHESIZED_WIRE_89;
wire SYNTHESIZED_WIRE_90;
wire SYNTHESIZED_WIRE_91;
wire SYNTHESIZED_WIRE_92;
wire SYNTHESIZED_WIRE_93;
wire SYNTHESIZED_WIRE_94;
wire SYNTHESIZED_WIRE_95;
wire SYNTHESIZED_WIRE_96;
wire SYNTHESIZED_WIRE_97;
wire SYNTHESIZED_WIRE_98;
wire SYNTHESIZED_WIRE_99;
wire SYNTHESIZED_WIRE_100;
wire SYNTHESIZED_WIRE_101;
wire SYNTHESIZED_WIRE_102;
wire SYNTHESIZED_WIRE_103;
wire SYNTHESIZED_WIRE_104;
```

```
wire SYNTHESIZED_WIRE_105;
wire SYNTHESIZED_WIRE_106;
wire SYNTHESIZED_WIRE_107;
wire SYNTHESIZED_WIRE_108;
wire SYNTHESIZED_WIRE_109;
wire SYNTHESIZED_WIRE_110;
wire SYNTHESIZED_WIRE_111;
wire SYNTHESIZED_WIRE_112;
wire SYNTHESIZED_WIRE_113;
wire SYNTHESIZED_WIRE_114;
wire SYNTHESIZED_WIRE_115;
wire SYNTHESIZED_WIRE_116;
wire SYNTHESIZED_WIRE_117;
wire SYNTHESIZED_WIRE_118;
wire SYNTHESIZED_WIRE_119;
wire SYNTHESIZED_WIRE_120;
wire SYNTHESIZED_WIRE_121;
wire SYNTHESIZED_WIRE_122;
wire SYNTHESIZED_WIRE_123;
wire SYNTHESIZED_WIRE_124;
wire SYNTHESIZED_WIRE_125;
wire SYNTHESIZED_WIRE_126;
wire SYNTHESIZED_WIRE_127;
wire SYNTHESIZED_WIRE_128;
wire SYNTHESIZED_WIRE_129;
wire SYNTHESIZED_WIRE_130;
wire SYNTHESIZED_WIRE_131;
wire SYNTHESIZED_WIRE_132;
wire SYNTHESIZED_WIRE_133;
wire SYNTHESIZED_WIRE_134;
```

```
wire SYNTHESIZED_WIRE_135;
wire SYNTHESIZED_WIRE_136;
wire SYNTHESIZED_WIRE_137;
wire SYNTHESIZED_WIRE_138;
wire SYNTHESIZED_WIRE_139;
wire SYNTHESIZED_WIRE_140;
wire SYNTHESIZED_WIRE_141;
wire SYNTHESIZED_WIRE_142;
wire SYNTHESIZED_WIRE_143;
wire SYNTHESIZED_WIRE_144;
wire SYNTHESIZED_WIRE_145;
wire SYNTHESIZED_WIRE_146;
wire SYNTHESIZED_WIRE_147;
wire SYNTHESIZED_WIRE_148;
wire SYNTHESIZED_WIRE_149;
wire SYNTHESIZED_WIRE_150;
wire SYNTHESIZED_WIRE_151;
wire SYNTHESIZED_WIRE_152;
wire SYNTHESIZED_WIRE_153;
wire SYNTHESIZED_WIRE_154;
wire SYNTHESIZED_WIRE_155;
wire SYNTHESIZED_WIRE_156;
wire SYNTHESIZED_WIRE_157;
wire SYNTHESIZED_WIRE_158;
wire SYNTHESIZED_WIRE_159;
wire SYNTHESIZED_WIRE_160;
wire SYNTHESIZED_WIRE_161;
wire SYNTHESIZED_WIRE_162;
wire SYNTHESIZED_WIRE_163;
wire SYNTHESIZED_WIRE_164;
```

wire SYNTHESIZED\_WIRE\_165;  
wire SYNTHESIZED\_WIRE\_166;  
wire SYNTHESIZED\_WIRE\_167;  
wire SYNTHESIZED\_WIRE\_168;  
wire SYNTHESIZED\_WIRE\_169;  
wire SYNTHESIZED\_WIRE\_170;  
wire SYNTHESIZED\_WIRE\_171;  
wire SYNTHESIZED\_WIRE\_172;  
wire SYNTHESIZED\_WIRE\_173;  
wire SYNTHESIZED\_WIRE\_174;  
wire SYNTHESIZED\_WIRE\_175;  
wire SYNTHESIZED\_WIRE\_176;  
wire SYNTHESIZED\_WIRE\_177;  
wire SYNTHESIZED\_WIRE\_178;  
wire SYNTHESIZED\_WIRE\_179;  
wire SYNTHESIZED\_WIRE\_180;  
wire SYNTHESIZED\_WIRE\_181;  
wire SYNTHESIZED\_WIRE\_182;  
wire SYNTHESIZED\_WIRE\_183;  
wire SYNTHESIZED\_WIRE\_184;  
wire SYNTHESIZED\_WIRE\_185;  
wire SYNTHESIZED\_WIRE\_186;  
wire SYNTHESIZED\_WIRE\_187;  
wire SYNTHESIZED\_WIRE\_188;  
wire SYNTHESIZED\_WIRE\_189;  
wire SYNTHESIZED\_WIRE\_190;  
wire SYNTHESIZED\_WIRE\_191;  
wire SYNTHESIZED\_WIRE\_192;  
wire SYNTHESIZED\_WIRE\_193;  
wire SYNTHESIZED\_WIRE\_194;

wire SYNTHESIZED\_WIRE\_195;  
wire SYNTHESIZED\_WIRE\_196;  
wire SYNTHESIZED\_WIRE\_197;  
wire SYNTHESIZED\_WIRE\_198;  
wire SYNTHESIZED\_WIRE\_199;  
wire SYNTHESIZED\_WIRE\_200;  
wire SYNTHESIZED\_WIRE\_201;  
wire SYNTHESIZED\_WIRE\_202;  
wire SYNTHESIZED\_WIRE\_203;  
wire SYNTHESIZED\_WIRE\_204;  
wire SYNTHESIZED\_WIRE\_205;  
wire SYNTHESIZED\_WIRE\_206;  
wire SYNTHESIZED\_WIRE\_207;  
wire SYNTHESIZED\_WIRE\_208;  
wire SYNTHESIZED\_WIRE\_209;  
wire SYNTHESIZED\_WIRE\_210;  
wire SYNTHESIZED\_WIRE\_211;  
wire SYNTHESIZED\_WIRE\_212;  
wire SYNTHESIZED\_WIRE\_213;  
wire SYNTHESIZED\_WIRE\_214;  
wire SYNTHESIZED\_WIRE\_215;  
wire SYNTHESIZED\_WIRE\_216;  
wire SYNTHESIZED\_WIRE\_217;  
wire SYNTHESIZED\_WIRE\_218;  
wire SYNTHESIZED\_WIRE\_219;  
wire SYNTHESIZED\_WIRE\_220;  
wire SYNTHESIZED\_WIRE\_221;  
wire SYNTHESIZED\_WIRE\_222;  
wire SYNTHESIZED\_WIRE\_223;  
wire SYNTHESIZED\_WIRE\_224;

```

wire SYNTHESIZED_WIRE_225;
wire SYNTHESIZED_WIRE_226;
wire SYNTHESIZED_WIRE_227;
wire SYNTHESIZED_WIRE_228;
wire SYNTHESIZED_WIRE_229;
wire SYNTHESIZED_WIRE_230;
wire SYNTHESIZED_WIRE_231;
wire SYNTHESIZED_WIRE_232;
wire SYNTHESIZED_WIRE_233;
wire SYNTHESIZED_WIRE_234;
wire SYNTHESIZED_WIRE_235;
wire SYNTHESIZED_WIRE_236;
wire SYNTHESIZED_WIRE_237;
wire SYNTHESIZED_WIRE_238;

```

```

\8fadd      b2v_inst(
    .A1(a1ANDb0),
    .A3(a3ANDb0),
    .CIN(0),
    .B1(a0ANDb1),
    .A2(a2ANDb0),
    .B2(a1ANDb1),
    .A4(a4ANDb0),
    .B4(a3ANDb1),
    .A5(a5ANDb0),
    .A7(a7ANDb0),

```

```
.B6(a5ANDb1),  
.B3(a2ANDb1),  
.B8(a7ANDb1),  
.B5(a4ANDb1),  
.A6(a6ANDb0),  
.B7(a6ANDb1),  
.A8(a8ANDb0),  
.SUM2(SYNTHESIZED_WIRE_1),  
.SUM5(SYNTHESIZED_WIRE_4),  
.SUM6(SYNTHESIZED_WIRE_5),  
.SUM1(DATA_OUTPUT[1]),  
.COUT(SYNTHESIZED_WIRE_0),  
.SUM3(SYNTHESIZED_WIRE_3),  
.SUM4(SYNTHESIZED_WIRE_2),  
.SUM7(SYNTHESIZED_WIRE_7),  
.SUM8(SYNTHESIZED_WIRE_6));
```

```
assigna14ANDb4 = A[14] & B[4];
```

```
assigna15ANDb4 = A[15] & B[4];
```

```
assigna0ANDb5 = A[0] & B[5];
```

```
assigna1ANDb5 = A[1] & B[5];
```

```
assigna2ANDb5 = A[2] & B[5];
```

```
assigna3ANDb5 = A[3] & B[5];
```



assigna4ANDb5 = A[4] & B[5];

assigna5ANDb5 = A[5] & B[5];

assigna6ANDb5 = A[6] & B[5];

assigna7ANDb5 = A[7] & B[5];

assigna8ANDb5 = A[8] & B[5];

assigna9ANDb5 = A[9] & B[5];

assigna10ANDb5 = A[10] & B[5];

assigna11ANDb5 = A[11] & B[5];

assigna12ANDb5 = A[12] & B[5];

assigna13ANDb5 = A[13] & B[5];

assigna14ANDb5 = A[14] & B[5];

assigna15ANDb5 = A[15] & B[5];

assigna0ANDb6 = A[0] & B[6];

assigna1ANDb6 = A[1] & B[6];

assigna2ANDb6 = A[2] & B[6];

`assigna3ANDb6 = A[3] & B[6];`

`assigna4ANDb6 = A[4] & B[6];`

`assigna5ANDb6 = A[5] & B[6];`

`assigna6ANDb6 = A[6] & B[6];`

`assigna7ANDb6 = A[7] & B[6];`

`assigna8ANDb6 = A[8] & B[6];`

`assigna9ANDb6 = A[9] & B[6];`

`assigna10ANDb6 = A[10] & B[6];`

`assigna11ANDb6 = A[11] & B[6];`

`assigna12ANDb6 = A[12] & B[6];`

`assigna13ANDb6 = A[13] & B[6];`

`assigna14ANDb6 = A[14] & B[6];`

`assigna15ANDb6 = A[15] & B[6];`

`assigna0ANDb7 = A[0] & B[7];`

`assigna1ANDb7 = A[1] & B[7];`

assigna2ANDb7 = A[2] & B[7];

assigna3ANDb7 = A[3] & B[7];

assigna4ANDb7 = A[4] & B[7];

assigna5ANDb7 = A[5] & B[7];

```
\8fadd      b2v_inst14(  
    .A1(a9ANDb0),  
    .A3(a11ANDb0),  
    .CIN(SYNTHESIZED_WIRE_0),  
    .B1(a8ANDb1),  
    .A2(a10ANDb0),  
    .B2(a9ANDb1),  
    .A4(a12ANDb0),  
    .B4(a11ANDb1),  
    .A5(a13ANDb0),  
    .A7(a15ANDb0),  
    .B6(a13ANDb1),  
    .B3(a10ANDb1),  
    .B8(a15ANDb1),  
    .B5(a12ANDb1),  
    .A6(a14ANDb0),  
    .B7(a14ANDb1),  
    .A8(0),  
    .SUM2(SYNTHESIZED_WIRE_9),  
    .SUM5(SYNTHESIZED_WIRE_13),  
    .SUM6(SYNTHESIZED_WIRE_14),
```

```
.SUM1(SYNTHESIZED_WIRE_8),  
.COUT(SYNTHESIZED_WIRE_17),  
.SUM3(SYNTHESIZED_WIRE_12),  
.SUM4(SYNTHESIZED_WIRE_10),  
.SUM7(SYNTHESIZED_WIRE_16),  
.SUM8(SYNTHESIZED_WIRE_15));
```

```
assigna6ANDb7 = A[6] & B[7];
```

```
assigna7ANDb7 = A[7] & B[7];
```

```
assigna8ANDb7 = A[8] & B[7];
```

```
assigna9ANDb7 = A[9] & B[7];
```

```
assigna10ANDb7 = A[10] & B[7];
```

```
assigna11ANDb7 = A[11] & B[7];
```

```
assigna12ANDb7 = A[12] & B[7];
```

```
assigna13ANDb7 = A[13] & B[7];
```

```
assigna14ANDb7 = A[14] & B[7];
```

```
assigna15ANDb7 = A[15] & B[7];
```

```
\8fadd      b2v_inst15(  
      .A1(SYNTHESIZED_WIRE_1),
```

```

.A3(SYNTHESIZED_WIRE_2),
.CIN(0),
.B1(a0ANDb2),
.A2(SYNTHESIZED_WIRE_3),
.B2(a1ANDb2),
.A4(SYNTHESIZED_WIRE_4),
.B4(a3ANDb2),
.A5(SYNTHESIZED_WIRE_5),
.A7(SYNTHESIZED_WIRE_6),
.B6(a5ANDb2),
.B3(a2ANDb2),
.B8(a7ANDb2),
.B5(a4ANDb2),
.A6(SYNTHESIZED_WIRE_7),
.B7(a6ANDb2),
.A8(SYNTHESIZED_WIRE_8),
.SUM2(SYNTHESIZED_WIRE_18),
.SUM5(SYNTHESIZED_WIRE_21),
.SUM6(SYNTHESIZED_WIRE_22),
.SUM1(DATA_OUTPUT[2]),
.COUT(SYNTHESIZED_WIRE_11),
.SUM3(SYNTHESIZED_WIRE_20),
.SUM4(SYNTHESIZED_WIRE_19),
.SUM7(SYNTHESIZED_WIRE_24),
.SUM8(SYNTHESIZED_WIRE_23));

```

```

assigna0ANDb8 = A[0] & B[8];

```

```

assigna1ANDb8 = A[1] & B[8];

```

assigna2ANDb8 = A[2] & B[8];

assigna3ANDb8 = A[3] & B[8];

assigna4ANDb8 = A[4] & B[8];

assigna5ANDb8 = A[5] & B[8];

assigna6ANDb8 = A[6] & B[8];

assigna7ANDb8 = A[7] & B[8];

assigna8ANDb8 = A[8] & B[8];

assigna9ANDb8 = A[9] & B[8];

```
\8fadd      b2v_inst16(  
    .A1(SYNTHESIZED_WIRE_9),  
    .A3(SYNTHESIZED_WIRE_10),  
    .CIN(SYNTHESIZED_WIRE_11),  
    .B1(a8ANDb2),  
    .A2(SYNTHESIZED_WIRE_12),  
    .B2(a9ANDb2),  
    .A4(SYNTHESIZED_WIRE_13),  
    .B4(a11ANDb2),  
    .A5(SYNTHESIZED_WIRE_14),  
    .A7(SYNTHESIZED_WIRE_15),  
    .B6(a13ANDb2),  
    .B3(a10ANDb2),
```

```
.B8(a15ANDb2),  
.B5(a12ANDb2),  
.A6(SYNTHESIZED_WIRE_16),  
.B7(a14ANDb2),  
.A8(SYNTHESIZED_WIRE_17),  
.SUM2(SYNTHESIZED_WIRE_26),  
.SUM5(SYNTHESIZED_WIRE_30),  
.SUM6(SYNTHESIZED_WIRE_31),  
.SUM1(SYNTHESIZED_WIRE_25),  
.COUT(SYNTHESIZED_WIRE_34),  
.SUM3(SYNTHESIZED_WIRE_29),  
.SUM4(SYNTHESIZED_WIRE_27),  
.SUM7(SYNTHESIZED_WIRE_33),  
.SUM8(SYNTHESIZED_WIRE_32));
```

```
assigna10ANDb8 = A[10] & B[8];
```

```
assigna11ANDb8 = A[11] & B[8];
```

```
assigna12ANDb8 = A[12] & B[8];
```

```
assigna13ANDb8 = A[13] & B[8];
```

```
assigna14ANDb8 = A[14] & B[8];
```

```
assigna15ANDb8 = A[15] & B[8];
```

```
assigna0ANDb9 = A[0] & B[9];
```

```
assigna1ANDb9 = A[1] & B[9];
```

assigna2ANDb9 = A[2] & B[9];

assigna3ANDb9 = A[3] & B[9];

```
\8fadd      b2v_inst17(  
    .A1(SYNTHESIZED_WIRE_18),  
    .A3(SYNTHESIZED_WIRE_19),  
    .CIN(0),  
    .B1(a0ANDb3),  
    .A2(SYNTHESIZED_WIRE_20),  
    .B2(a1ANDb3),  
    .A4(SYNTHESIZED_WIRE_21),  
    .B4(a3ANDb3),  
    .A5(SYNTHESIZED_WIRE_22),  
    .A7(SYNTHESIZED_WIRE_23),  
    .B6(a5ANDb3),  
    .B3(a2ANDb3),  
    .B8(a7ANDb3),  
    .B5(a4ANDb3),  
    .A6(SYNTHESIZED_WIRE_24),  
    .B7(a6ANDb3),  
    .A8(SYNTHESIZED_WIRE_25),  
    .SUM2(SYNTHESIZED_WIRE_35),  
    .SUM5(SYNTHESIZED_WIRE_38),  
    .SUM6(SYNTHESIZED_WIRE_39),  
    .SUM1(DATA_OUTPUT[3]),  
    .COUT(SYNTHESIZED_WIRE_28),  
    .SUM3(SYNTHESIZED_WIRE_37),
```



```
.SUM4(SYNTHESIZED_WIRE_36),  
.SUM7(SYNTHESIZED_WIRE_41),  
.SUM8(SYNTHESIZED_WIRE_40));
```

```
assigna4ANDb9 = A[4] & B[9];
```

```
assigna5ANDb9 = A[5] & B[9];
```

```
assigna6ANDb9 = A[6] & B[9];
```

```
assigna8ANDb9 = A[8] & B[9];
```

```
assigna9ANDb9 = A[9] & B[9];
```

```
assigna10ANDb9 = A[10] & B[9];
```

```
assigna11ANDb9 = A[11] & B[9];
```

```
assigna12ANDb9 = A[12] & B[9];
```

```
assigna13ANDb9 = A[13] & B[9];
```

```
\8fadd      b2v_inst18(  
    .A1(SYNTHESIZED_WIRE_26),  
    .A3(SYNTHESIZED_WIRE_27),  
    .CIN(SYNTHESIZED_WIRE_28),  
    .B1(a8ANDb3),  
    .A2(SYNTHESIZED_WIRE_29),
```

```

.B2(a9ANDb3),
.A4(SYNTHESIZED_WIRE_30),
.B4(a11ANDb3),
.A5(SYNTHESIZED_WIRE_31),
.A7(SYNTHESIZED_WIRE_32),
.B6(a13ANDb3),
.B3(a10ANDb3),
.B8(a15ANDb3),
.B5(a12ANDb2),
.A6(SYNTHESIZED_WIRE_33),
.B7(a14ANDb3),
.A8(SYNTHESIZED_WIRE_34),
.SUM2(SYNTHESIZED_WIRE_43),
.SUM5(SYNTHESIZED_WIRE_47),
.SUM6(SYNTHESIZED_WIRE_48),
.SUM1(SYNTHESIZED_WIRE_42),
.COUT(SYNTHESIZED_WIRE_51),
.SUM3(SYNTHESIZED_WIRE_46),
.SUM4(SYNTHESIZED_WIRE_44),
.SUM7(SYNTHESIZED_WIRE_50),
.SUM8(SYNTHESIZED_WIRE_49));

```

```

assigna14ANDb9 = A[14] & B[9];

```

```

assigna15ANDb9 = A[15] & B[9];

```

```

assigna0ANDb10 = A[0] & B[10];

```

```

assigna1ANDb10 = A[1] & B[10];

```

assigna2ANDb10 = A[2] & B[10];

assigna3ANDb10 = A[3] & B[10];

assigna4ANDb10 = A[4] & B[10];

assigna5ANDb10 = A[5] & B[10];

assigna6ANDb10 = A[6] & B[10];

assigna7ANDb10 = A[7] & B[10];

```
\8fadd      b2v_inst19(  
    .A1(SYNTHESIZED_WIRE_35),  
    .A3(SYNTHESIZED_WIRE_36),  
    .CIN(0),  
    .B1(a0ANDb4),  
    .A2(SYNTHESIZED_WIRE_37),  
    .B2(a1ANDb4),  
    .A4(SYNTHESIZED_WIRE_38),  
    .B4(a3ANDb4),  
    .A5(SYNTHESIZED_WIRE_39),  
    .A7(SYNTHESIZED_WIRE_40),  
    .B6(a5ANDb4),  
    .B3(a2ANDb4),  
    .B8(a7ANDb4),  
    .B5(a4ANDb4),  
    .A6(SYNTHESIZED_WIRE_41),  
    .B7(a6ANDb4),
```

```
.A8(SYNTHESIZED_WIRE_42),  
.SUM2(SYNTHESIZED_WIRE_52),  
.SUM5(SYNTHESIZED_WIRE_55),  
.SUM6(SYNTHESIZED_WIRE_56),  
.SUM1(DATA_OUTPUT[4]),  
.COUT(SYNTHESIZED_WIRE_45),  
.SUM3(SYNTHESIZED_WIRE_54),  
.SUM4(SYNTHESIZED_WIRE_53),  
.SUM7(SYNTHESIZED_WIRE_58),  
.SUM8(SYNTHESIZED_WIRE_57));
```

```
assigna8ANDb10 = A[8] & B[10];
```

```
assigna9ANDb10 = A[9] & B[10];
```

```
assigna10ANDb10 = A[10] & B[10];
```

```
assigna11ANDb10 = A[11] & B[10];
```

```
assigna12ANDb10 = A[12] & B[10];
```

```
assigna13ANDb10 = A[13] & B[10];
```

```
assigna14ANDb10 = A[14] & B[10];
```

```
assigna15ANDb10 = A[15] & B[10];
```

```
lpm_dff0    b2v_inst2(
    .clock(clk),
    .enable(w_a_enable),
    .data(DATA1),
    .q(A));
```

```
\8fadd      b2v_inst20(
    .A1(SYNTHESIZED_WIRE_43),
    .A3(SYNTHESIZED_WIRE_44),
    .CIN(SYNTHESIZED_WIRE_45),
    .B1(a8ANDb4),
    .A2(SYNTHESIZED_WIRE_46),
    .B2(a9ANDb4),
    .A4(SYNTHESIZED_WIRE_47),
    .B4(a11ANDb4),
    .A5(SYNTHESIZED_WIRE_48),
    .A7(SYNTHESIZED_WIRE_49),
    .B6(a13ANDb4),
    .B3(a10ANDb4),
    .B8(a15ANDb4),
    .B5(a12ANDb4),
    .A6(SYNTHESIZED_WIRE_50),
    .B7(a14ANDb4),
    .A8(SYNTHESIZED_WIRE_51),
    .SUM2(SYNTHESIZED_WIRE_60),
    .SUM5(SYNTHESIZED_WIRE_64),
    .SUM6(SYNTHESIZED_WIRE_65),
    .SUM1(SYNTHESIZED_WIRE_59),
    .COUT(SYNTHESIZED_WIRE_68),
```

```
.SUM3(SYNTHESIZED_WIRE_63),  
.SUM4(SYNTHESIZED_WIRE_61),  
.SUM7(SYNTHESIZED_WIRE_67),  
.SUM8(SYNTHESIZED_WIRE_66));
```

```
\8fadd      b2v_inst21(  
    .A1(SYNTHESIZED_WIRE_52),  
    .A3(SYNTHESIZED_WIRE_53),  
    .CIN(0),  
    .B1(a0ANDb5),  
    .A2(SYNTHESIZED_WIRE_54),  
    .B2(a1ANDb5),  
    .A4(SYNTHESIZED_WIRE_55),  
    .B4(a3ANDb5),  
    .A5(SYNTHESIZED_WIRE_56),  
    .A7(SYNTHESIZED_WIRE_57),  
    .B6(a5ANDb5),  
    .B3(a2ANDb5),  
    .B8(a7ANDb5),
```

```
.B5(a4ANDb5),  
.A6(SYNTHESIZED_WIRE_58),  
.B7(a6ANDb5),  
.A8(SYNTHESIZED_WIRE_59),  
.SUM2(SYNTHESIZED_WIRE_69),  
.SUM5(SYNTHESIZED_WIRE_72),  
.SUM6(SYNTHESIZED_WIRE_73),  
.SUM1(DATA_OUTPUT[5]),  
.COUT(SYNTHESIZED_WIRE_62),  
.SUM3(SYNTHESIZED_WIRE_71),  
.SUM4(SYNTHESIZED_WIRE_70),  
.SUM7(SYNTHESIZED_WIRE_75),  
.SUM8(SYNTHESIZED_WIRE_74));
```

```
assigna0ANDb12 = A[0] & B[12];
```

```
assigna1ANDb12 = A[1] & B[12];
```

```
assigna2ANDb12 = A[2] & B[12];
```

```
assigna3ANDb12 = A[3] & B[12];
```

```
assigna4ANDb12 = A[4] & B[12];
```

```
assigna5ANDb12 = A[5] & B[12];
```

assignDATA\_OUTPUT[0] = A[0] & B[0];

assigna6ANDb12 = A[6] & B[12];

assigna7ANDb12 = A[7] & B[12];

assigna8ANDb12 = A[8] & B[12];

assigna9ANDb12 = A[9] & B[12];

assigna10ANDb12 = A[10] & B[12];

assigna11ANDb12 = A[11] & B[12];

assigna12ANDb12 = A[12] & B[12];

assigna13ANDb12 = A[13] & B[12];

assigna14ANDb12 = A[14] & B[12];

assigna15ANDb12 = A[15] & B[12];

assigna1ANDb0 = A[1] & B[0];

assigna0ANDb13 = A[0] & B[13];

assigna2ANDb13 = A[2] & B[13];

assigna3ANDb13 = A[3] & B[13];



assigna4ANDb13 = A[4] & B[13];

assigna5ANDb13 = A[5] & B[13];

assigna6ANDb13 = A[6] & B[13];

assigna7ANDb13 = A[7] & B[13];

assigna8ANDb13 = A[8] & B[13];

assigna9ANDb13 = A[9] & B[13];

assigna2ANDb0 = A[2] & B[0];

assigna10ANDb13 = A[10] & B[13];

assigna11ANDb13 = A[11] & B[13];

assigna12ANDb13 = A[12] & B[13];

assigna13ANDb13 = A[13] & B[13];

assigna14ANDb13 = A[14] & B[13];

assigna15ANDb13 = A[15] & B[13];

assigna0ANDb14 = A[0] & B[14];

assigna1ANDb14 = A[1] & B[14];

`assigna2ANDb14 = A[2] & B[14];`

`assigna3ANDb14 = A[3] & B[14];`

`assigna3ANDb0 = A[3] & B[0];`

`assigna4ANDb14 = A[4] & B[14];`

`assigna5ANDb14 = A[5] & B[14];`

`assigna6ANDb14 = A[6] & B[14];`

`assigna7ANDb14 = A[7] & B[14];`

`assigna8ANDb14 = A[8] & B[14];`

`assigna9ANDb14 = A[9] & B[14];`

`assigna10ANDb14 = A[10] & B[14];`

`assigna11ANDb14 = A[11] & B[14];`

`assigna12ANDb14 = A[12] & B[14];`

`assigna13ANDb14 = A[13] & B[14];`

`assigna4ANDb0 = A[4] & B[0];`

`assigna14ANDb14 = A[14] & B[14];`

`assigna15ANDb14 = A[15] & B[14];`

`assigna0ANDb15 = A[0] & B[15];`

`assigna1ANDb15 = A[1] & B[15];`

`assigna2ANDb15 = A[2] & B[15];`

`assigna3ANDb15 = A[3] & B[15];`

`assigna4ANDb15 = A[4] & B[15];`

`assigna5ANDb15 = A[5] & B[15];`

`assigna6ANDb15 = A[6] & B[15];`

`assigna7ANDb15 = A[7] & B[15];`

`assigna5ANDb0 = A[5] & B[0];`

`assigna8ANDb15 = A[8] & B[15];`

`assigna9ANDb15 = A[9] & B[15];`

`assigna10ANDb15 = A[10] & B[15];`

`assigna11ANDb15 = A[11] & B[15];`

`assigna12ANDb15 = A[12] & B[15];`

assigna13ANDb15 = A[13] & B[15];

assigna14ANDb15 = A[14] & B[15];

assigna15ANDb15 = A[15] & B[15];

```
\8fadd      b2v_inst278(  
    .A1(SYNTHESIZED_WIRE_60),  
    .A3(SYNTHESIZED_WIRE_61),  
    .CIN(SYNTHESIZED_WIRE_62),  
    .B1(a8ANDb5),  
    .A2(SYNTHESIZED_WIRE_63),  
    .B2(a9ANDb5),  
    .A4(SYNTHESIZED_WIRE_64),  
    .B4(a11ANDb5),  
    .A5(SYNTHESIZED_WIRE_65),  
    .A7(SYNTHESIZED_WIRE_66),  
    .B6(a13ANDb5),  
    .B3(a10ANDb5),  
    .B8(a15ANDb5),  
    .B5(a12ANDb5),  
    .A6(SYNTHESIZED_WIRE_67),  
    .B7(a14ANDb5),  
    .A8(SYNTHESIZED_WIRE_68),  
    .SUM2(SYNTHESIZED_WIRE_77),  
    .SUM5(SYNTHESIZED_WIRE_81),  
    .SUM6(SYNTHESIZED_WIRE_82),  
    .SUM1(SYNTHESIZED_WIRE_76),
```

```
.COUT(SYNTHESIZED_WIRE_85),
.SUM3(SYNTHESIZED_WIRE_80),
.SUM4(SYNTHESIZED_WIRE_78),
.SUM7(SYNTHESIZED_WIRE_84),
.SUM8(SYNTHESIZED_WIRE_83));
```

```
\8fadd      b2v_inst279(
.A1(SYNTHESIZED_WIRE_69),
.A3(SYNTHESIZED_WIRE_70),
.CIN(0),
.B1(a0ANDb6),
.A2(SYNTHESIZED_WIRE_71),
.B2(a1ANDb6),
.A4(SYNTHESIZED_WIRE_72),
.B4(a3ANDb6),
.A5(SYNTHESIZED_WIRE_73),
.A7(SYNTHESIZED_WIRE_74),
.B6(a5ANDb6),
.B3(a2ANDb6),
.B8(a7ANDb6),
.B5(a4ANDb6),
.A6(SYNTHESIZED_WIRE_75),
.B7(a6ANDb6),
.A8(SYNTHESIZED_WIRE_76),
.SUM2(SYNTHESIZED_WIRE_86),
.SUM5(SYNTHESIZED_WIRE_89),
.SUM6(SYNTHESIZED_WIRE_90),
.SUM1(DATA_OUTPUT[6]),
.COUT(SYNTHESIZED_WIRE_79),
```

```
.SUM3(SYNTHESIZED_WIRE_88),
.SUM4(SYNTHESIZED_WIRE_87),
.SUM7(SYNTHESIZED_WIRE_92),
.SUM8(SYNTHESIZED_WIRE_91));
```

```
assign a6ANDb0 = A[6] & B[0];
```

```
\8fadd      b2v_inst280(
.A1(SYNTHESIZED_WIRE_77),
.A3(SYNTHESIZED_WIRE_78),
.CIN(SYNTHESIZED_WIRE_79),
.B1(a8ANDb6),
.A2(SYNTHESIZED_WIRE_80),
.B2(a9ANDb6),
.A4(SYNTHESIZED_WIRE_81),
.B4(a11ANDb6),
.A5(SYNTHESIZED_WIRE_82),
.A7(SYNTHESIZED_WIRE_83),
.B6(a13ANDb6),
.B3(a10ANDb6),
.B8(a15ANDb6),
.B5(a12ANDb6),
.A6(SYNTHESIZED_WIRE_84),
.B7(a14ANDb6),
.A8(SYNTHESIZED_WIRE_85),
.SUM2(SYNTHESIZED_WIRE_94),
.SUM5(SYNTHESIZED_WIRE_98),
.SUM6(SYNTHESIZED_WIRE_99),
.SUM1(SYNTHESIZED_WIRE_93),
```

```
.COUT(SYNTHESIZED_WIRE_102),
.SUM3(SYNTHESIZED_WIRE_97),
.SUM4(SYNTHESIZED_WIRE_95),
.SUM7(SYNTHESIZED_WIRE_101),
.SUM8(SYNTHESIZED_WIRE_100));
```

```
\8fadd      b2v_inst281(
.A1(SYNTHESIZED_WIRE_86),
.A3(SYNTHESIZED_WIRE_87),
.CIN(0),
.B1(a0ANDb7),
.A2(SYNTHESIZED_WIRE_88),
.B2(a1ANDb7),
.A4(SYNTHESIZED_WIRE_89),
.B4(a3ANDb7),
.A5(SYNTHESIZED_WIRE_90),
.A7(SYNTHESIZED_WIRE_91),
.B6(a5ANDb7),
.B3(a2ANDb7),
.B8(a7ANDb7),
.B5(a4ANDb7),
.A6(SYNTHESIZED_WIRE_92),
.B7(a6ANDb7),
.A8(SYNTHESIZED_WIRE_93),
.SUM2(SYNTHESIZED_WIRE_103),
.SUM5(SYNTHESIZED_WIRE_106),
.SUM6(SYNTHESIZED_WIRE_107),
.SUM1(DATA_OUTPUT[7]),
.COUT(SYNTHESIZED_WIRE_96),
```

```
.SUM3(SYNTHESIZED_WIRE_105),  
.SUM4(SYNTHESIZED_WIRE_104),  
.SUM7(SYNTHESIZED_WIRE_109),  
.SUM8(SYNTHESIZED_WIRE_108));
```

```
\8fadd      b2v_inst282(  
  .A1(SYNTHESIZED_WIRE_94),  
  .A3(SYNTHESIZED_WIRE_95),  
  .CIN(SYNTHESIZED_WIRE_96),  
  .B1(a8ANDb7),  
  .A2(SYNTHESIZED_WIRE_97),  
  .B2(a9ANDb7),  
  .A4(SYNTHESIZED_WIRE_98),  
  .B4(a11ANDb7),  
  .A5(SYNTHESIZED_WIRE_99),  
  .A7(SYNTHESIZED_WIRE_100),  
  .B6(a13ANDb7),  
  .B3(a10ANDb7),  
  .B8(a15ANDb7),  
  .B5(a12ANDb7),  
  .A6(SYNTHESIZED_WIRE_101),  
  .B7(a14ANDb7),  
  .A8(SYNTHESIZED_WIRE_102),  
  .SUM2(SYNTHESIZED_WIRE_111),  
  .SUM5(SYNTHESIZED_WIRE_115),  
  .SUM6(SYNTHESIZED_WIRE_116),  
  .SUM1(SYNTHESIZED_WIRE_110),  
  .COUT(SYNTHESIZED_WIRE_119),  
  .SUM3(SYNTHESIZED_WIRE_114),
```



```
.SUM4(SYNTHESIZED_WIRE_112),  
.SUM7(SYNTHESIZED_WIRE_118),  
.SUM8(SYNTHESIZED_WIRE_117));
```

```
\8fadd      b2v_inst283(  
  .A1(SYNTHESIZED_WIRE_103),  
  .A3(SYNTHESIZED_WIRE_104),  
  .CIN(0),  
  .B1(a0ANDB8),  
  .A2(SYNTHESIZED_WIRE_105),  
  .B2(a1ANDB8),  
  .A4(SYNTHESIZED_WIRE_106),  
  .B4(a3ANDB8),  
  .A5(SYNTHESIZED_WIRE_107),  
  .A7(SYNTHESIZED_WIRE_108),  
  .B6(a5ANDB8),  
  .B3(a2ANDB8),  
  .B8(a7ANDB8),  
  .B5(a4ANDB8),  
  .A6(SYNTHESIZED_WIRE_109),  
  .B7(a6ANDB8),  
  .A8(SYNTHESIZED_WIRE_110),  
  .SUM2(SYNTHESIZED_WIRE_120),  
  .SUM5(SYNTHESIZED_WIRE_123),  
  .SUM6(SYNTHESIZED_WIRE_124),  
  .SUM1(DATA_OUTPUT[8]),  
  .COUT(SYNTHESIZED_WIRE_113),  
  .SUM3(SYNTHESIZED_WIRE_122),  
  .SUM4(SYNTHESIZED_WIRE_121),
```

```
.SUM7(SYNTHESIZED_WIRE_126),  
.SUM8(SYNTHESIZED_WIRE_125));
```

```
\8fadd      b2v_inst284(  
  .A1(SYNTHESIZED_WIRE_111),  
  .A3(SYNTHESIZED_WIRE_112),  
  .CIN(SYNTHESIZED_WIRE_113),  
  .B1(a8ANDb8),  
  .A2(SYNTHESIZED_WIRE_114),  
  .B2(a9ANDb8),  
  .A4(SYNTHESIZED_WIRE_115),  
  .B4(a11ANDb8),  
  .A5(SYNTHESIZED_WIRE_116),  
  .A7(SYNTHESIZED_WIRE_117),  
  .B6(a13ANDb8),  
  .B3(a10ANDb8),  
  .B8(a15ANDb8),  
  .B5(a12ANDb8),  
  .A6(SYNTHESIZED_WIRE_118),  
  .B7(a14ANDb8),  
  .A8(SYNTHESIZED_WIRE_119),  
  .SUM2(SYNTHESIZED_WIRE_128),  
  .SUM5(SYNTHESIZED_WIRE_132),  
  .SUM6(SYNTHESIZED_WIRE_133),  
  .SUM1(SYNTHESIZED_WIRE_127),  
  .COUT(SYNTHESIZED_WIRE_136),  
  .SUM3(SYNTHESIZED_WIRE_131),  
  .SUM4(SYNTHESIZED_WIRE_129),  
  .SUM7(SYNTHESIZED_WIRE_135),
```

```
.SUM8(SYNTHESIZED_WIRE_134));
```

```
\8fadd      b2v_inst285(  
    .A1(SYNTHESIZED_WIRE_120),  
    .A3(SYNTHESIZED_WIRE_121),  
    .CIN(0),  
    .B1(a0ANDB2),  
    .A2(SYNTHESIZED_WIRE_122),  
    .B2(a1ANDB2),  
    .A4(SYNTHESIZED_WIRE_123),  
    .B4(a3ANDB2),  
    .A5(SYNTHESIZED_WIRE_124),  
    .A7(SYNTHESIZED_WIRE_125),  
    .B6(a5ANDB2),  
    .B3(a2ANDB2),  
    .B8(a7ANDB2),  
    .B5(a4ANDB2),  
    .A6(SYNTHESIZED_WIRE_126),  
    .B7(a6ANDB2),  
    .A8(SYNTHESIZED_WIRE_127),  
    .SUM2(SYNTHESIZED_WIRE_137),  
    .SUM5(SYNTHESIZED_WIRE_140),  
    .SUM6(SYNTHESIZED_WIRE_141),  
    .SUM1(DATA_OUTPUT[9]),  
    .COUT(SYNTHESIZED_WIRE_130),  
    .SUM3(SYNTHESIZED_WIRE_139),  
    .SUM4(SYNTHESIZED_WIRE_138),  
    .SUM7(SYNTHESIZED_WIRE_143),  
    .SUM8(SYNTHESIZED_WIRE_142));
```

```

\8fadd      b2v_inst286(
    .A1(SYNTHESIZED_WIRE_128),
    .A3(SYNTHESIZED_WIRE_129),
    .CIN(SYNTHESIZED_WIRE_130),
    .B1(a8ANDb8),
    .A2(SYNTHESIZED_WIRE_131),
    .B2(a9ANDb8),
    .A4(SYNTHESIZED_WIRE_132),
    .B4(a11ANDb8),
    .A5(SYNTHESIZED_WIRE_133),
    .A7(SYNTHESIZED_WIRE_134),
    .B6(a13ANDb8),
    .B3(a10ANDb8),
    .B8(a15ANDb8),
    .B5(a12ANDb8),
    .A6(SYNTHESIZED_WIRE_135),
    .B7(a14ANDb8),
    .A8(SYNTHESIZED_WIRE_136),
    .SUM2(SYNTHESIZED_WIRE_145),
    .SUM5(SYNTHESIZED_WIRE_149),
    .SUM6(SYNTHESIZED_WIRE_150),
    .SUM1(SYNTHESIZED_WIRE_144),
    .COUT(SYNTHESIZED_WIRE_153),
    .SUM3(SYNTHESIZED_WIRE_148),
    .SUM4(SYNTHESIZED_WIRE_146),
    .SUM7(SYNTHESIZED_WIRE_152),
    .SUM8(SYNTHESIZED_WIRE_151));

```

```

\8fadd      b2v_inst287(
    .A1(SYNTHESIZED_WIRE_137),
    .A3(SYNTHESIZED_WIRE_138),
    .CIN(0),
    .B1(a0ANDb9),
    .A2(SYNTHESIZED_WIRE_139),
    .B2(a1ANDb9),
    .A4(SYNTHESIZED_WIRE_140),
    .B4(a3ANDb9),
    .A5(SYNTHESIZED_WIRE_141),
    .A7(SYNTHESIZED_WIRE_142),
    .B6(a5ANDb9),
    .B3(a2ANDb9),
    .B8(a7ANDb2),
    .B5(a4ANDb9),
    .A6(SYNTHESIZED_WIRE_143),
    .B7(a6ANDb9),
    .A8(SYNTHESIZED_WIRE_144),
    .SUM2(SYNTHESIZED_WIRE_154),
    .SUM5(SYNTHESIZED_WIRE_157),
    .SUM6(SYNTHESIZED_WIRE_158),
    .SUM1(DATA_OUTPUT[10]),
    .COUT(SYNTHESIZED_WIRE_147),
    .SUM3(SYNTHESIZED_WIRE_156),
    .SUM4(SYNTHESIZED_WIRE_155),
    .SUM7(SYNTHESIZED_WIRE_160),
    .SUM8(SYNTHESIZED_WIRE_159));

```

```

\8fadd      b2v_inst288(
    .A1(SYNTHESIZED_WIRE_145),
    .A3(SYNTHESIZED_WIRE_146),
    .CIN(SYNTHESIZED_WIRE_147),
    .B1(a8ANDb9),
    .A2(SYNTHESIZED_WIRE_148),
    .B2(a9ANDb9),
    .A4(SYNTHESIZED_WIRE_149),
    .B4(a11ANDb9),
    .A5(SYNTHESIZED_WIRE_150),
    .A7(SYNTHESIZED_WIRE_151),
    .B6(a13ANDb9),
    .B3(a10ANDb9),
    .B8(a15ANDb9),
    .B5(a12ANDb9),
    .A6(SYNTHESIZED_WIRE_152),
    .B7(a14ANDb9),
    .A8(SYNTHESIZED_WIRE_153),
    .SUM2(SYNTHESIZED_WIRE_162),
    .SUM5(SYNTHESIZED_WIRE_166),
    .SUM6(SYNTHESIZED_WIRE_167),
    .SUM1(SYNTHESIZED_WIRE_161),
    .COUT(SYNTHESIZED_WIRE_170),
    .SUM3(SYNTHESIZED_WIRE_165),
    .SUM4(SYNTHESIZED_WIRE_163),
    .SUM7(SYNTHESIZED_WIRE_169),
    .SUM8(SYNTHESIZED_WIRE_168));

```

```

\8fadd      b2v_inst289(

```

```

.A1(SYNTHESIZED_WIRE_154),
.A3(SYNTHESIZED_WIRE_155),
.CIN(0),
.B1(a0ANDb10),
.A2(SYNTHESIZED_WIRE_156),
.B2(a1ANDb10),
.A4(SYNTHESIZED_WIRE_157),
.B4(a3ANDb10),
.A5(SYNTHESIZED_WIRE_158),
.A7(SYNTHESIZED_WIRE_159),
.B6(a5ANDb10),
.B3(a2ANDb10),
.B8(a7ANDb10),
.B5(a4ANDb10),
.A6(SYNTHESIZED_WIRE_160),
.B7(a6ANDb10),
.A8(SYNTHESIZED_WIRE_161),
.SUM2(SYNTHESIZED_WIRE_171),
.SUM5(SYNTHESIZED_WIRE_174),
.SUM6(SYNTHESIZED_WIRE_175),
.SUM1(DATA_OUTPUT[11]),
.COUT(SYNTHESIZED_WIRE_164),
.SUM3(SYNTHESIZED_WIRE_173),
.SUM4(SYNTHESIZED_WIRE_172),
.SUM7(SYNTHESIZED_WIRE_177),
.SUM8(SYNTHESIZED_WIRE_176));

```

```

assign a7ANDb0 = A[7] & B[0];

```

```

\8fadd      b2v_inst290(
    .A1(SYNTHESIZED_WIRE_162),
    .A3(SYNTHESIZED_WIRE_163),
    .CIN(SYNTHESIZED_WIRE_164),
    .B1(a8ANDb10),
    .A2(SYNTHESIZED_WIRE_165),
    .B2(a9ANDb10),
    .A4(SYNTHESIZED_WIRE_166),
    .B4(a11ANDb10),
    .A5(SYNTHESIZED_WIRE_167),
    .A7(SYNTHESIZED_WIRE_168),
    .B6(a13ANDb10),
    .B3(a10ANDb10),
    .B8(a15ANDb10),
    .B5(a12ANDb10),
    .A6(SYNTHESIZED_WIRE_169),
    .B7(a14ANDb10),
    .A8(SYNTHESIZED_WIRE_170),
    .SUM2(SYNTHESIZED_WIRE_179),
    .SUM5(SYNTHESIZED_WIRE_183),
    .SUM6(SYNTHESIZED_WIRE_184),
    .SUM1(SYNTHESIZED_WIRE_178),
    .COUT(SYNTHESIZED_WIRE_187),
    .SUM3(SYNTHESIZED_WIRE_182),
    .SUM4(SYNTHESIZED_WIRE_180),
    .SUM7(SYNTHESIZED_WIRE_186),
    .SUM8(SYNTHESIZED_WIRE_185));

```

```

\8fadd      b2v_inst291(

```



```

.A1(SYNTHESIZED_WIRE_171),
.A3(SYNTHESIZED_WIRE_172),
.CIN(0),
.B1(a0ANDb12),
.A2(SYNTHESIZED_WIRE_173),
.B2(a1ANDb12),
.A4(SYNTHESIZED_WIRE_174),
.B4(a3ANDb12),
.A5(SYNTHESIZED_WIRE_175),
.A7(SYNTHESIZED_WIRE_176),
.B6(a5ANDb12),
.B3(a2ANDb12),
.B8(a7ANDb12),
.B5(a4ANDb12),
.A6(SYNTHESIZED_WIRE_177),
.B7(a6ANDb12),
.A8(SYNTHESIZED_WIRE_178),
.SUM2(SYNTHESIZED_WIRE_188),
.SUM5(SYNTHESIZED_WIRE_191),
.SUM6(SYNTHESIZED_WIRE_192),
.SUM1(DATA_OUTPUT[12]),
.COUT(SYNTHESIZED_WIRE_181),
.SUM3(SYNTHESIZED_WIRE_190),
.SUM4(SYNTHESIZED_WIRE_189),
.SUM7(SYNTHESIZED_WIRE_194),
.SUM8(SYNTHESIZED_WIRE_193));

```

```

\8fadd      b2v_inst292(
.A1(SYNTHESIZED_WIRE_179),

```

```

.A3(SYNTHESIZED_WIRE_180),
.CIN(SYNTHESIZED_WIRE_181),
.B1(a8ANDb12),
.A2(SYNTHESIZED_WIRE_182),
.B2(a9ANDb12),
.A4(SYNTHESIZED_WIRE_183),
.B4(a11ANDb12),
.A5(SYNTHESIZED_WIRE_184),
.A7(SYNTHESIZED_WIRE_185),
.B6(a13ANDb12),
.B3(a10ANDb12),
.B8(a15ANDb12),
.B5(a12ANDb12),
.A6(SYNTHESIZED_WIRE_186),
.B7(a14ANDb12),
.A8(SYNTHESIZED_WIRE_187),
.SUM2(SYNTHESIZED_WIRE_196),
.SUM5(SYNTHESIZED_WIRE_200),
.SUM6(SYNTHESIZED_WIRE_201),
.SUM1(SYNTHESIZED_WIRE_195),
.COUT(SYNTHESIZED_WIRE_204),
.SUM3(SYNTHESIZED_WIRE_199),
.SUM4(SYNTHESIZED_WIRE_197),
.SUM7(SYNTHESIZED_WIRE_203),
.SUM8(SYNTHESIZED_WIRE_202));

```

```

\8fadd      b2v_inst293(
.A1(SYNTHESIZED_WIRE_188),
.A3(SYNTHESIZED_WIRE_189),

```

```

.CIN(0),
.B1(a0ANDb13),
.A2(SYNTHESIZED_WIRE_190),
.B2(a1ANDb2),
.A4(SYNTHESIZED_WIRE_191),
.B4(a3ANDb13),
.A5(SYNTHESIZED_WIRE_192),
.A7(SYNTHESIZED_WIRE_193),
.B6(a5ANDb13),
.B3(a2ANDb13),
.B8(a7ANDb13),
.B5(a4ANDb13),
.A6(SYNTHESIZED_WIRE_194),
.B7(a6ANDb13),
.A8(SYNTHESIZED_WIRE_195),
.SUM2(SYNTHESIZED_WIRE_205),
.SUM5(SYNTHESIZED_WIRE_208),
.SUM6(SYNTHESIZED_WIRE_209),
.SUM1(DATA_OUTPUT[13]),
.COUT(SYNTHESIZED_WIRE_198),
.SUM3(SYNTHESIZED_WIRE_207),
.SUM4(SYNTHESIZED_WIRE_206),
.SUM7(SYNTHESIZED_WIRE_211),
.SUM8(SYNTHESIZED_WIRE_210));

```

```

\8fadd      b2v_inst294(
.A1(SYNTHESIZED_WIRE_196),
.A3(SYNTHESIZED_WIRE_197),
.CIN(SYNTHESIZED_WIRE_198),

```

```

.B1(a8ANDb13),
.A2(SYNTHESIZED_WIRE_199),
.B2(a9ANDb13),
.A4(SYNTHESIZED_WIRE_200),
.B4(a11ANDb13),
.A5(SYNTHESIZED_WIRE_201),
.A7(SYNTHESIZED_WIRE_202),
.B6(a13ANDb13),
.B3(a10ANDb13),
.B8(a15ANDb13),
.B5(a12ANDb13),
.A6(SYNTHESIZED_WIRE_203),
.B7(a14ANDb13),
.A8(SYNTHESIZED_WIRE_204),
.SUM2(SYNTHESIZED_WIRE_213),
.SUM5(SYNTHESIZED_WIRE_217),
.SUM6(SYNTHESIZED_WIRE_218),
.SUM1(SYNTHESIZED_WIRE_212),
.COUT(SYNTHESIZED_WIRE_221),
.SUM3(SYNTHESIZED_WIRE_216),
.SUM4(SYNTHESIZED_WIRE_214),
.SUM7(SYNTHESIZED_WIRE_220),
.SUM8(SYNTHESIZED_WIRE_219));

```

```

\8fadd      b2v_inst295(
.A1(SYNTHESIZED_WIRE_205),
.A3(SYNTHESIZED_WIRE_206),
.CIN(0),
.B1(a0ANDb14),

```

```

.A2(SYNTHESIZED_WIRE_207),
.B2(a1ANDb14),
.A4(SYNTHESIZED_WIRE_208),
.B4(a3ANDb14),
.A5(SYNTHESIZED_WIRE_209),
.A7(SYNTHESIZED_WIRE_210),
.B6(a5ANDb14),
.B3(a2ANDb14),
.B8(a7ANDb14),
.B5(a4ANDb14),
.A6(SYNTHESIZED_WIRE_211),
.B7(a6ANDb14),
.A8(SYNTHESIZED_WIRE_212),
.SUM2(SYNTHESIZED_WIRE_222),
.SUM5(SYNTHESIZED_WIRE_225),
.SUM6(SYNTHESIZED_WIRE_226),
.SUM1(DATA_OUTPUT[14]),
.COUT(SYNTHESIZED_WIRE_215),
.SUM3(SYNTHESIZED_WIRE_224),
.SUM4(SYNTHESIZED_WIRE_223),
.SUM7(SYNTHESIZED_WIRE_228),
.SUM8(SYNTHESIZED_WIRE_227));

```

```

\8fadd      b2v_inst296(
.A1(SYNTHESIZED_WIRE_213),
.A3(SYNTHESIZED_WIRE_214),
.CIN(SYNTHESIZED_WIRE_215),
.B1(a8ANDb14),
.A2(SYNTHESIZED_WIRE_216),

```

```

.B2(a9ANDb14),
.A4(SYNTHESIZED_WIRE_217),
.B4(a11ANDb14),
.A5(SYNTHESIZED_WIRE_218),
.A7(SYNTHESIZED_WIRE_219),
.B6(a13ANDb14),
.B3(a10ANDb14),
.B8(a15ANDb14),
.B5(a12ANDb14),
.A6(SYNTHESIZED_WIRE_220),
.B7(a14ANDb14),
.A8(SYNTHESIZED_WIRE_221),
.SUM2(SYNTHESIZED_WIRE_230),
.SUM5(SYNTHESIZED_WIRE_234),
.SUM6(SYNTHESIZED_WIRE_235),
.SUM1(SYNTHESIZED_WIRE_229),
.COUT(SYNTHESIZED_WIRE_238),
.SUM3(SYNTHESIZED_WIRE_233),
.SUM4(SYNTHESIZED_WIRE_231),
.SUM7(SYNTHESIZED_WIRE_237),
.SUM8(SYNTHESIZED_WIRE_236));

```

```

\8fadd      b2v_inst297(
.A1(SYNTHESIZED_WIRE_222),
.A3(SYNTHESIZED_WIRE_223),
.CIN(0),
.B1(a0ANDb15),
.A2(SYNTHESIZED_WIRE_224),
.B2(a1ANDb15),

```

```

.A4(SYNTHESIZED_WIRE_225),
.B4(a3ANDb15),
.A5(SYNTHESIZED_WIRE_226),
.A7(SYNTHESIZED_WIRE_227),
.B6(a5ANDb15),
.B3(a2ANDb15),
.B8(a7ANDb15),
.B5(a4ANDb15),
.A6(SYNTHESIZED_WIRE_228),
.B7(a6ANDb15),
.A8(SYNTHESIZED_WIRE_229),
.SUM2(DATA_OUTPUT[16]),
.SUM5(DATA_OUTPUT[19]),
.SUM6(DATA_OUTPUT[20]),
.SUM1(DATA_OUTPUT[15]),
.COUT(SYNTHESIZED_WIRE_232),
.SUM3(DATA_OUTPUT[17]),
.SUM4(DATA_OUTPUT[18]),
.SUM7(DATA_OUTPUT[21]),
.SUM8(DATA_OUTPUT[22]));

```

```

\8fadd      b2v_inst298(
.A1(SYNTHESIZED_WIRE_230),
.A3(SYNTHESIZED_WIRE_231),
.CIN(SYNTHESIZED_WIRE_232),
.B1(a8ANDb15),
.A2(SYNTHESIZED_WIRE_233),
.B2(a9ANDb15),
.A4(SYNTHESIZED_WIRE_234),

```

```

.B4(a11ANDb15),
.A5(SYNTHESIZED_WIRE_235),
.A7(SYNTHESIZED_WIRE_236),
.B6(a13ANDb15),
.B3(a10ANDb15),
.B8(a15ANDb15),
.B5(a12ANDb15),
.A6(SYNTHESIZED_WIRE_237),
.B7(a14ANDb15),
.A8(SYNTHESIZED_WIRE_238),
.SUM2(DATA_OUTPUT[24]),
.SUM5(DATA_OUTPUT[27]),
.SUM6(DATA_OUTPUT[28]),
.SUM1(DATA_OUTPUT[23]),
.COUT(DATA_OUTPUT[31]),
.SUM3(DATA_OUTPUT[25]),
.SUM4(DATA_OUTPUT[26]),
.SUM7(DATA_OUTPUT[29]),
.SUM8(DATA_OUTPUT[30]));

```

```

lpm_dff0    b2v_inst3(
    .clock(clk),
    .enable(w_b_ensble),
    .data(DATA2),
    .q(B));

```

```

assign a8ANDb0 = A[8] & B[0];

```

```

assign a9ANDb0 = A[9] & B[0];

```



```
assigna10ANDb0 = A[10] & B[0];
```

```
assigna11ANDb0 = A[11] & B[0];
```

```
assigna12ANDb0 = A[12] & B[0];
```

```
assigna13ANDb0 = A[13] & B[0];
```

```
assigna14ANDb0 = A[14] & B[0];
```

```
assigna15ANDb0 = A[15] & B[0];
```

```
assigna0ANDb1 = A[0] & B[1];
```

```
assigna1ANDb1 = A[1] & B[1];
```

```
lpm_dff1    b2v_inst4(  
    .clock(clk),  
    .enable(w_result_enable),  
    .data(DATA_OUTPUT),  
    .q(Y));
```

```
assigna2ANDb1 = A[2] & B[1];
```

```
assigna3ANDb1 = A[3] & B[1];
```

```
assigna4ANDb1 = A[4] & B[1];
```

`assigna5ANDb1 = A[5] & B[1];`

`assigna6ANDb1 = A[6] & B[1];`

`assigna7ANDb1 = A[7] & B[1];`

`assigna8ANDb1 = A[8] & B[1];`

`assigna9ANDb1 = A[9] & B[1];`

`assigna10ANDb1 = A[10] & B[1];`

`assigna11ANDb1 = A[11] & B[1];`

`assigna12ANDb1 = A[12] & B[1];`

`assigna13ANDb1 = A[13] & B[1];`

`assigna14ANDb1 = A[14] & B[1];`

`assigna15ANDb1 = A[15] & B[1];`

`assigna0ANDb2 = A[0] & B[2];`

`assigna1ANDb2 = A[1] & B[2];`

`assigna2ANDb2 = A[2] & B[2];`

`assigna3ANDb2 = A[3] & B[2];`

assigna4ANDb2 = A[4] & B[2];

assigna5ANDb2 = A[5] & B[2];

assigna6ANDb2 = A[6] & B[2];

assigna7ANDb2 = A[7] & B[2];

assigna8ANDb2 = A[8] & B[2];

assigna9ANDb2 = A[9] & B[2];

assigna10ANDb2 = A[10] & B[2];

assigna11ANDb2 = A[11] & B[2];

assigna12ANDb2 = A[12] & B[2];

assigna13ANDb2 = A[13] & B[2];

assigna14ANDb2 = A[14] & B[2];

assigna15ANDb2 = A[15] & B[2];

assigna0ANDb3 = A[0] & B[3];

assigna1ANDb3 = A[1] & B[3];

assigna2ANDb3 = A[2] & B[3];

$\text{assigna3ANDb3} = A[3] \ \& \ B[3];$

$\text{assigna4ANDb3} = A[4] \ \& \ B[3];$

$\text{assigna5ANDb3} = A[5] \ \& \ B[3];$

$\text{assigna6ANDb3} = A[6] \ \& \ B[3];$

$\text{assigna7ANDb3} = A[7] \ \& \ B[3];$

$\text{assigna8ANDb3} = A[8] \ \& \ B[3];$

$\text{assigna9ANDb3} = A[9] \ \& \ B[3];$

$\text{assigna10ANDb3} = A[10] \ \& \ B[3];$

$\text{assigna11ANDb3} = A[11] \ \& \ B[3];$

$\text{assigna13ANDb3} = A[13] \ \& \ B[3];$

$\text{assigna14ANDb3} = A[14] \ \& \ B[3];$

$\text{assigna15ANDb3} = A[15] \ \& \ B[3];$

$\text{assigna0ANDb4} = A[0] \ \& \ B[4];$

$\text{assigna1ANDb4} = A[1] \ \& \ B[4];$

$\text{assigna2ANDb4} = A[2] \ \& \ B[4];$

```
assign a3ANDb4 = A[3] & B[4];
```

```
assign a4ANDb4 = A[4] & B[4];
```

```
assign a5ANDb4 = A[5] & B[4];
```

```
assign a6ANDb4 = A[6] & B[4];
```

```
assign a7ANDb4 = A[7] & B[4];
```

```
assign a8ANDb4 = A[8] & B[4];
```

```
assign a9ANDb4 = A[9] & B[4];
```

```
assign a10ANDb4 = A[10] & B[4];
```

```
assign a11ANDb4 = A[11] & B[4];
```

```
assign a12ANDb4 = A[12] & B[4];
```

```
assign a13ANDb4 = A[13] & B[4];
```

```
assign 0 = 0;
```

```
endmodule
```

На рисунке 12 представлен результат симуляции схемы, реализованной с помощью языка описания аппаратуры System Verilog.

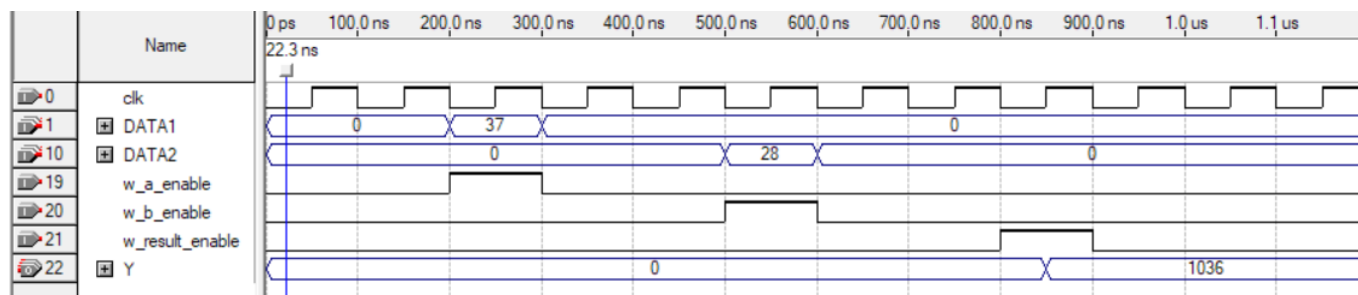


Рисунок 12 – Временная диаграмма

## **2. ВЫБОР FPGA**

В ходе выполнения проекта было принято решение использовать ПЛИС MAX II – EPM1279F256A5 ввиду хорошей производительности и большого наличия пинов ввода/вывода. На рисунке 13 представлена схема подключения к ПЛИС.

# Top View

## MAX II - EPM1270F256A5

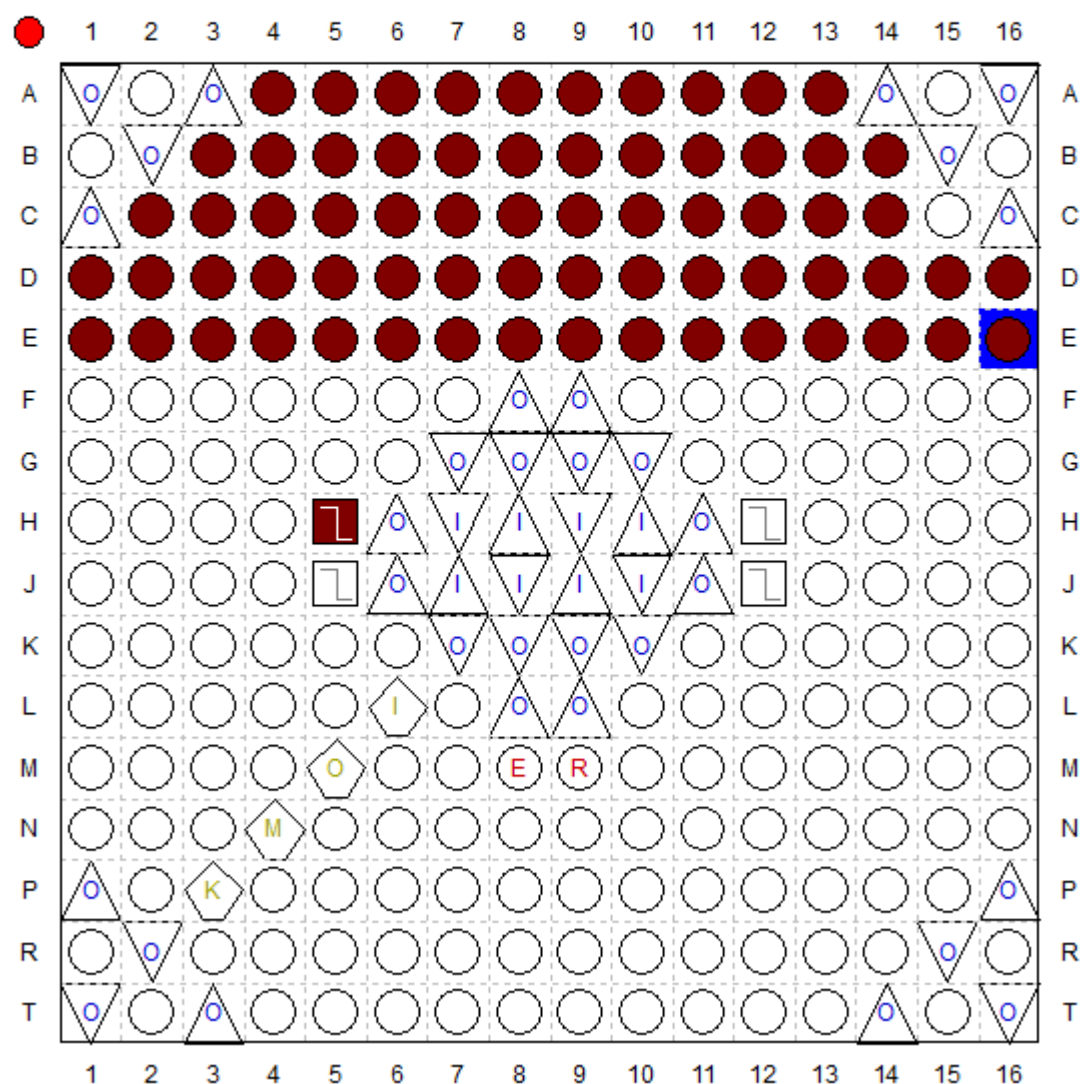


Рисунок 13 – Схема подключения выходов



На рисунке 14 представлена таблица назначения входов и выходов схемы.

	Node Name	Direction	Location	I/O Bank	VREF Group	I/O Standard
1	clk	Input	PIN_H5	1		3.3-V LVTTTL (default)
2	DATA1[15]	Input	PIN_B8	2		3.3-V LVTTTL (default)
3	DATA1[14]	Input	PIN_B7	2		3.3-V LVTTTL (default)
4	DATA1[13]	Input	PIN_B6	2		3.3-V LVTTTL (default)
5	DATA1[12]	Input	PIN_B5	2		3.3-V LVTTTL (default)
6	DATA1[11]	Input	PIN_B4	2		3.3-V LVTTTL (default)
7	DATA1[10]	Input	PIN_B3	2		3.3-V LVTTTL (default)
8	DATA1[9]	Input	PIN_A13	2		3.3-V LVTTTL (default)
9	DATA1[8]	Input	PIN_A12	2		3.3-V LVTTTL (default)
10	DATA1[7]	Input	PIN_A11	2		3.3-V LVTTTL (default)
11	DATA1[6]	Input	PIN_A10	2		3.3-V LVTTTL (default)
12	DATA1[5]	Input	PIN_A9	2		3.3-V LVTTTL (default)
13	DATA1[4]	Input	PIN_A8	2		3.3-V LVTTTL (default)
14	DATA1[3]	Input	PIN_A7	2		3.3-V LVTTTL (default)
15	DATA1[2]	Input	PIN_A6	2		3.3-V LVTTTL (default)
16	DATA1[1]	Input	PIN_A5	2		3.3-V LVTTTL (default)
17	DATA1[0]	Input	PIN_A4	2		3.3-V LVTTTL (default)
18	DATA2[15]	Input	PIN_C11	2		3.3-V LVTTTL (default)
19	DATA2[14]	Input	PIN_C10	2		3.3-V LVTTTL (default)
20	DATA2[13]	Input	PIN_C9	2		3.3-V LVTTTL (default)
21	DATA2[12]	Input	PIN_C8	2		3.3-V LVTTTL (default)
22	DATA2[11]	Input	PIN_C7	2		3.3-V LVTTTL (default)
23	DATA2[10]	Input	PIN_C6	2		3.3-V LVTTTL (default)
24	DATA2[9]	Input	PIN_C5	2		3.3-V LVTTTL (default)
25	DATA2[8]	Input	PIN_C4	2		3.3-V LVTTTL (default)
26	DATA2[7]	Input	PIN_C3	1		3.3-V LVTTTL (default)
27	DATA2[6]	Input	PIN_C2	1		3.3-V LVTTTL (default)
28	DATA2[5]	Input	PIN_B14	2		3.3-V LVTTTL (default)
29	DATA2[4]	Input	PIN_B13	2		3.3-V LVTTTL (default)
30	DATA2[3]	Input	PIN_B12	2		3.3-V LVTTTL (default)
31	DATA2[2]	Input	PIN_B11	2		3.3-V LVTTTL (default)
32	DATA2[1]	Input	PIN_B10	2		3.3-V LVTTTL (default)
33	DATA2[0]	Input	PIN_B9	2		3.3-V LVTTTL (default)
34	w_a_enable	Input	PIN_C12	2		3.3-V LVTTTL (default)
35	w_b_enable	Input	PIN_C13	2		3.3-V LVTTTL (default)
36	w_result_enable	Input	PIN_C14	3		3.3-V LVTTTL (default)
37	Y[31]	Output	PIN_E16	3		3.3-V LVTTTL (default)
38	Y[30]	Output	PIN_E15	3		3.3-V LVTTTL (default)
39	Y[29]	Output	PIN_E14	3		3.3-V LVTTTL (default)
40	Y[28]	Output	PIN_E13	3		3.3-V LVTTTL (default)
41	Y[27]	Output	PIN_E12	3		3.3-V LVTTTL (default)
42	Y[26]	Output	PIN_E11	2		3.3-V LVTTTL (default)
43	Y[25]	Output	PIN_E10	2		3.3-V LVTTTL (default)
44	Y[24]	Output	PIN_E9	2		3.3-V LVTTTL (default)
45	Y[23]	Output	PIN_E8	2		3.3-V LVTTTL (default)
46	Y[22]	Output	PIN_E7	2		3.3-V LVTTTL (default)
47	Y[21]	Output	PIN_E6	2		3.3-V LVTTTL (default)

a)

б)

48		Y[20]	Output	PIN_E5	1		3.3-V LVTTTL (default)
49		Y[19]	Output	PIN_E4	1		3.3-V LVTTTL (default)
50		Y[18]	Output	PIN_E3	1		3.3-V LVTTTL (default)
51		Y[17]	Output	PIN_E2	1		3.3-V LVTTTL (default)
52		Y[16]	Output	PIN_E1	1		3.3-V LVTTTL (default)
53		Y[15]	Output	PIN_D16	3		3.3-V LVTTTL (default)
54		Y[14]	Output	PIN_D15	3		3.3-V LVTTTL (default)
55		Y[13]	Output	PIN_D14	3		3.3-V LVTTTL (default)
56		Y[12]	Output	PIN_D13	3		3.3-V LVTTTL (default)
57		Y[11]	Output	PIN_D12	2		3.3-V LVTTTL (default)
58		Y[10]	Output	PIN_D11	2		3.3-V LVTTTL (default)
59		Y[9]	Output	PIN_D10	2		3.3-V LVTTTL (default)
60		Y[8]	Output	PIN_D9	2		3.3-V LVTTTL (default)
61		Y[7]	Output	PIN_D8	2		3.3-V LVTTTL (default)
62		Y[6]	Output	PIN_D7	2		3.3-V LVTTTL (default)
63		Y[5]	Output	PIN_D6	2		3.3-V LVTTTL (default)
64		Y[4]	Output	PIN_D5	2		3.3-V LVTTTL (default)
65		Y[3]	Output	PIN_D4	2		3.3-V LVTTTL (default)
66		Y[2]	Output	PIN_D3	1		3.3-V LVTTTL (default)
67		Y[1]	Output	PIN_D2	1		3.3-V LVTTTL (default)
68		Y[0]	Output	PIN_D1	1		3.3-V LVTTTL (default)

Рисунок 14 (а, б) – Таблица назначения входов и выходов схемы

## **ЗАКЛЮЧЕНИЕ**

В ходе выполнения проекта было получено устройство умножения 16-ти разрядных двоичных чисел. Моделирование устройства выполнено с помощью графической среды САПР Quartus и с помощью языка описания аппаратуры System Verilog. Обе реализации дают одинаковый и подобный друг другу результат.

Устройство работает конкретно, соответствует поставленной задаче и заданному функционалу, исправно умножает шестнадцатиразрядные числа.

## **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Чернышев С. А. Синтез и компьютерный анализ элементов и узлов ЦВМ на FPGA с использованием САПР Quartus: метод. указания по курсовому проектированию / С.А. Чернышев. – СПб., 2018. – 31 с.