浙江大学实验报告

专业：计算机科学与技术
姓名：吴同
学号：3170104848
日期：2020 年 3 月 26 日

课程名称：并行计算与多核编程　指导老师：楼学庆　电子邮件：wutongcs@zju.edu.cn
实验名称：矩阵乘法设计　实验类型：设计型　联系电话：18888922355

一、 实验目的

- 用 Verilog 设计 4x4 矩阵乘法器
- 对所设计的乘法器进行仿真测试
- 评价该乘法器的设计

二、 实验原理

1. Systolic 网络

本实验采用脉动（Systolic）阵列。脉动阵列早在 1982 年就已经被提出，近期 Google 的 TPU 中采用这一结构作为计算的核心。脉动阵列的核心思想是，让数据在运算单元中流动，以减少访存的次数，使硬件的结构规整、布线统一。

用于计算 4x4 矩阵乘法的网络结构如图 1所示。A 矩阵每行的数据逐个送入阵列中，B 矩阵每列的数据逐个送入阵列中，数据在阵列中流动，每个单元上保存运算的中间结果。当四个时钟周期过后，数据全部送入阵列中。再过三个时钟周期，全部计算完毕。

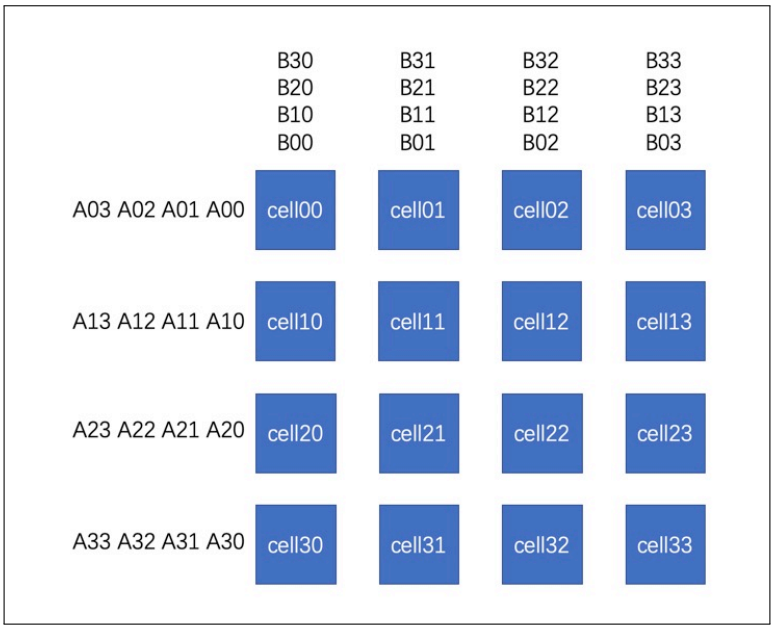

图 1: Systolic 阵列结构图

2. 网格设计思路

本实验采用中间结果保存在网格中，两个输入矩阵当数据流动的设计思路，而不是一个输入矩阵静止，另一个输入矩阵和中间结果流动的方式。每一个单元有两个输入口和三个输出口。如图 2所示，两个输入口的数据分别是来自上方和左方的单元，在该单元使用完毕后，将数据传送给下方和右方的单元。计算的中间结果保存在单元格中，并布置导线可以输出结果。

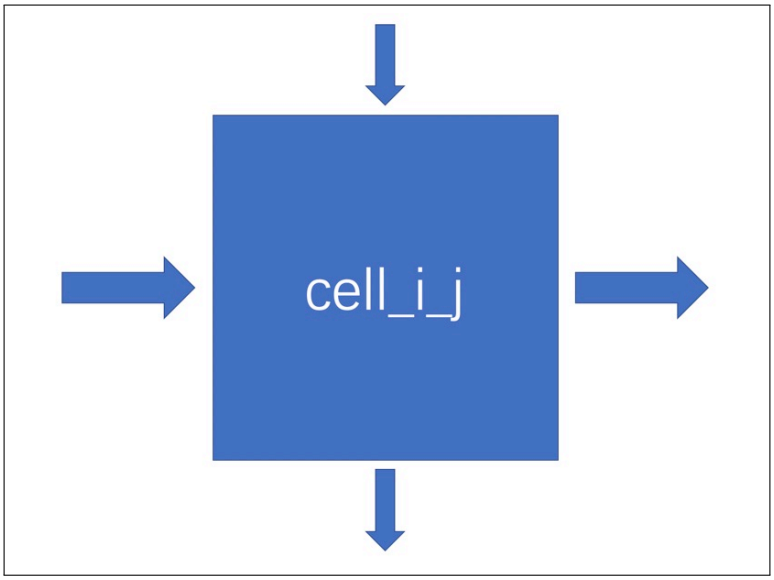

图 2: 数据流动方向

三、 实验过程

1. 设计网格

编写如下的 Verilog 代码。经过 Vivado 软件综合，生成的电路图如图 3所示。

```
module Cell(input clk,
        input rst,
        input [7:0] a_in,
        input [7:0] b_in,
        output reg [7:0] a_out,
        output reg [7:0] b_out,
        output reg [7:0] result);

   always @ (posedge clk or negedge rst)
   begin
      if (rst)
      begin
         a_out <= 0;
         b_out <= 0;
         result <= 0;
      end
      else begin
```

```
            result = result + a_in * b_in;
            a_out <= a_in;
            b_out <= b_in;
        end
    end

endmodule
```

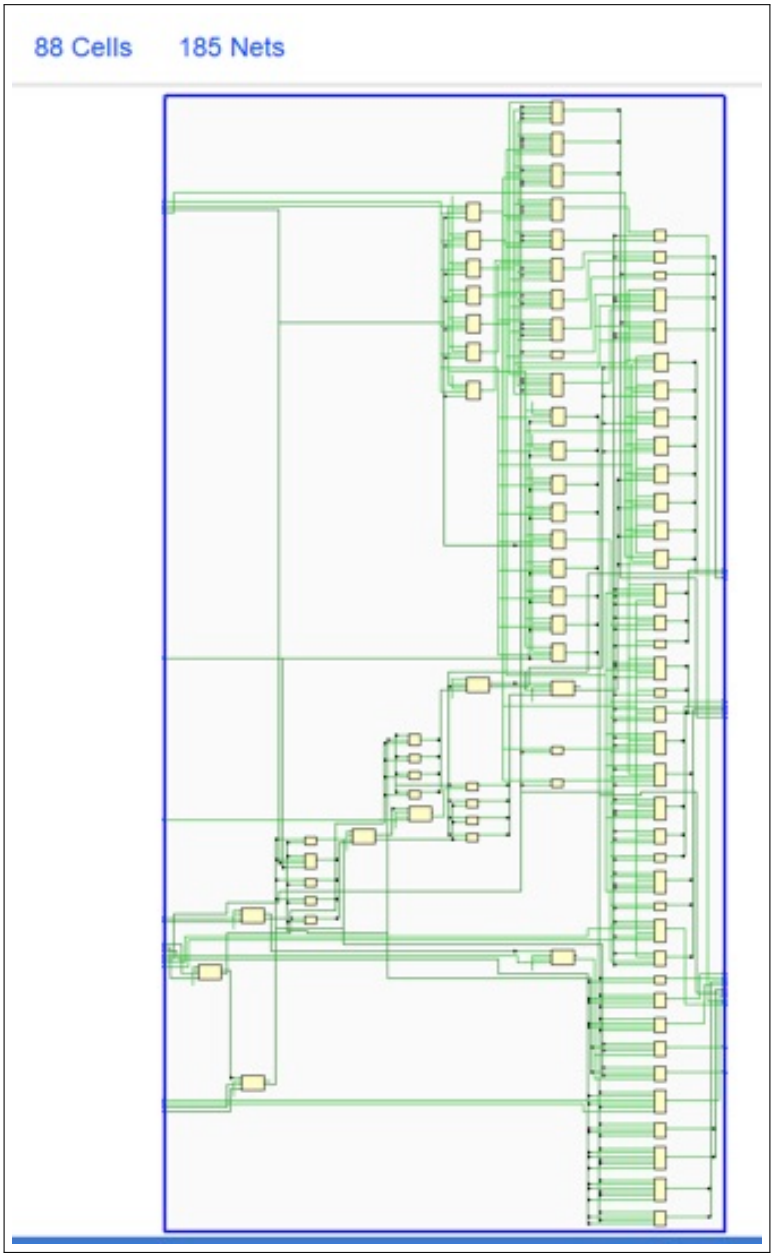

图 3: 单元格结构图

2. 搭建网络

将 16 个单元格连接起来，并在最左侧和最上侧各布置 4 个寄存器用于向网格中输入数据。综合生成的电路图如图 4所示。其中，输入模块中有若干个计数器和寄存器，用于向网络中逐个送入数据。

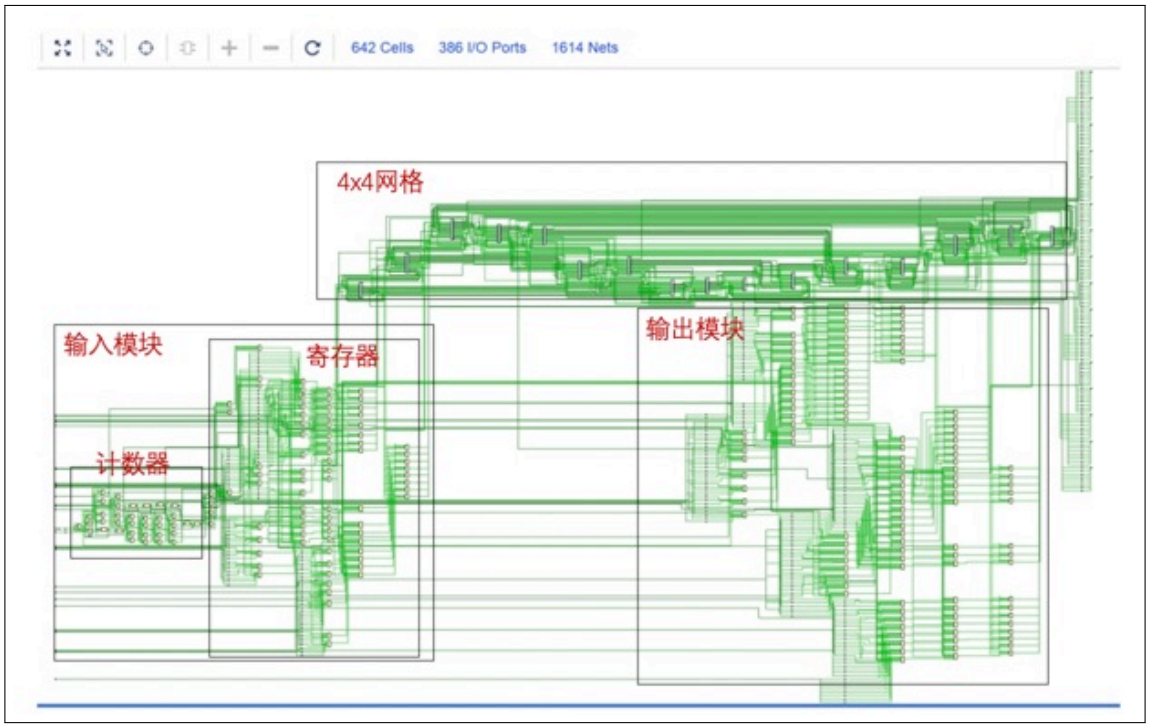

图 4: 网络电路图

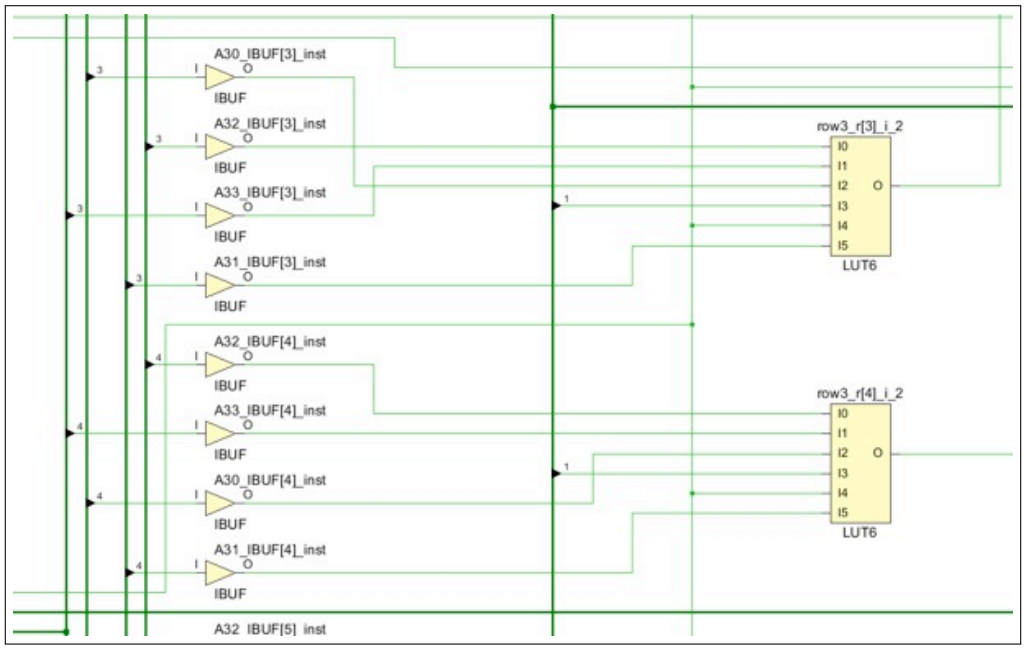

图 5: 输入模块局部图

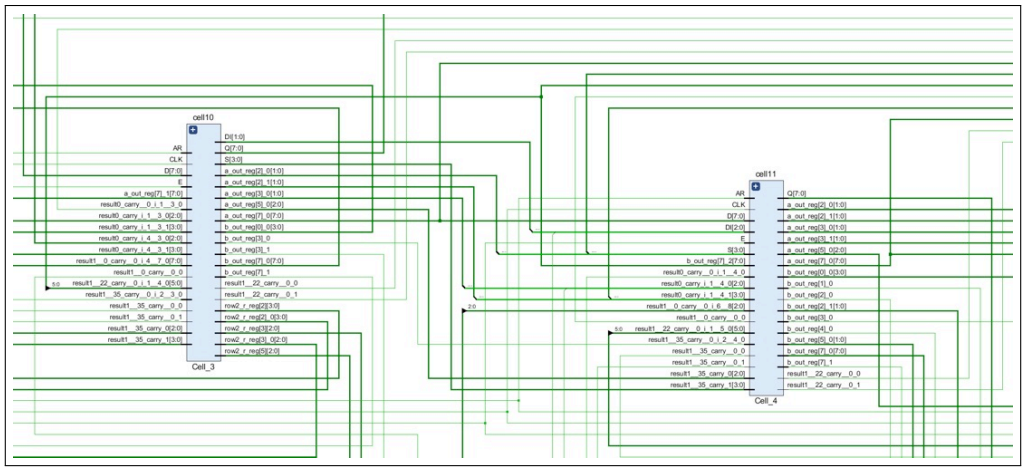

图 6: 单元格连接图

四、 实验结果

对所设计的乘法器进行仿真测试。

首先测试单个单元格，单元格能够在一个时钟周期内将送入的数据的乘积加到运算结果上，并将这对数据输出。

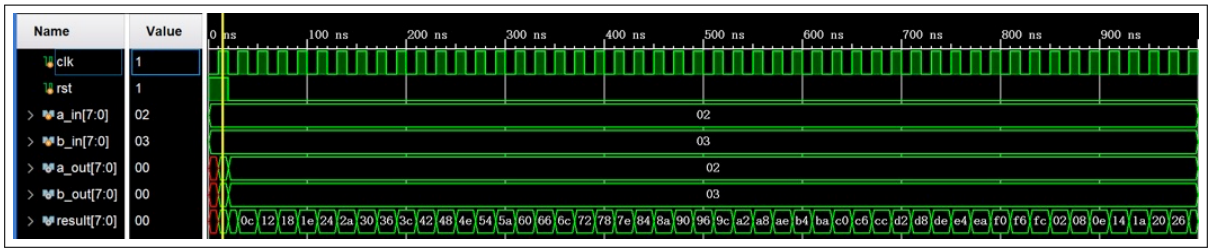

图 7: 单元格仿真测试图

测试乘法器。乘法器用七个时钟周期完成运算。运算结果正确。

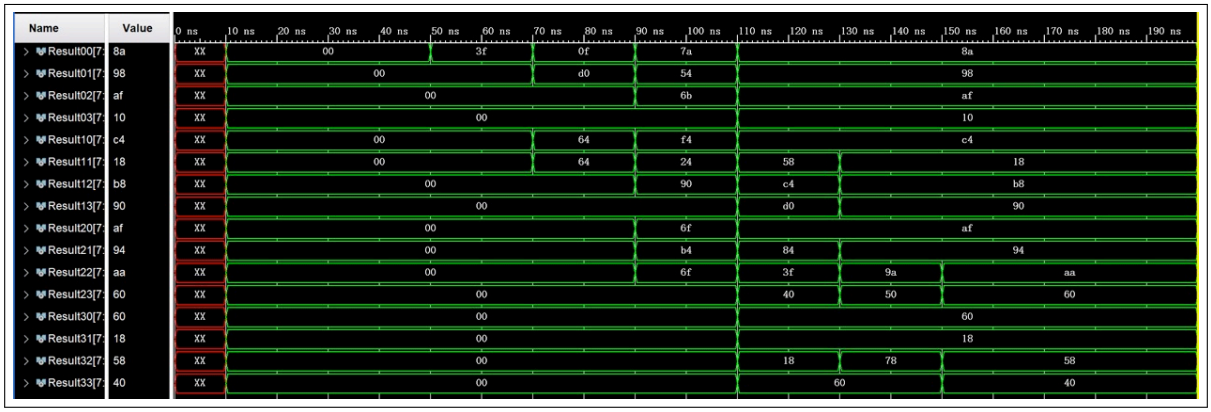

图 8: 乘法器仿真测试图

五、 分析讨论

在以往的学习中，我了解过矩阵运算可以通过设计专门的硬件进行加速。通过本次实验，我亲自体验了设计一种矩阵运算器件的过程，感到收获很大。通过硬件并行的方法，仅用七个时钟周期就完成了4x4 矩阵的乘法运算，这在软件层面上是很难达到的。

虽然本实验所设计的乘法器规模较小，但所使用的网络十分清晰，扩展性良好，只需要增加单元格个数，就很容易完成器件的扩展。并且随着矩阵规模的增大，所用时间仅随其线性增长。NxN 的矩阵乘法完成运算只需要 2N-1 个时钟周期。

事实上，Google 在其生产的神经网络专用芯片上，正是用这种网络实现的矩阵运算。这也说明了这一网络结构的性能好、扩展性强。