

设计报告

课程名称: 数字逻辑

题目: 矩阵的卷积运算

学院: 信息科学与工程学院

班级: 计算机基地班

小组成员: 罗鹏, 胡臻初, 付俊博, 李卓奥, 王子豪

December 19, 2021

目录

1	项目描述	2
1.1	项目环境与级别	2
1.2	任务目标	2
2	主要模块	3
2.1	加法器	3
2.2	乘法器	4
3	明细表	5
4	仿真测试	6
4.1	测试样例	6
4.2	仿真结果	7
5	电路相关	8
5.1	复杂度	8
5.2	电路图	8
6	延迟分析	9
7	FPGA 部署结果	9
7.1	样例 1	9
7.2	样例 2	10
7.3	样例 3	11
8	组员分工	11

1 项目描述

1.1 项目环境与级别

设计语言：verilog 硬件描述语言
 仿真环境：Vivado
 设计级别：门级

1.2 任务目标

采用 Verilog 描述一个 5*5 的卷积运算，卷积核与待运算矩阵为 5*5 矩阵。
 卷积运算采用公式如下：

$$Result = \sum_{m=1}^5 \sum_{n=1}^5 A(m,n) * B(m,n)$$

$$A_5 \otimes B_5 = \begin{bmatrix} a_{11} & \cdots & a_{15} \\ \vdots & \ddots & \vdots \\ a_{51} & \cdots & a_{55} \end{bmatrix} \otimes \begin{bmatrix} b_{11} & \cdots & b_{15} \\ \vdots & \ddots & \vdots \\ b_{51} & \cdots & b_{55} \end{bmatrix}$$

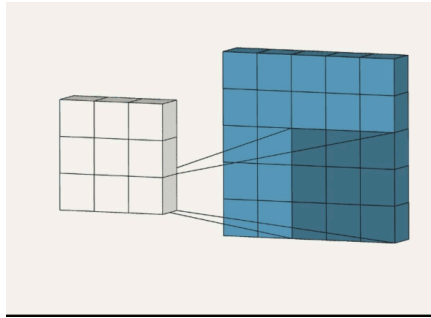
$$= a_{11}b_{11} + a_{12}b_{12} + \cdots + a_{55}b_{55}$$

其中 A 为被卷积矩阵，B 为卷积核，Result 为卷积结果。

因为 vivado 跑不了太大规模的程序，所以我们把任务改成实现 5*5 矩阵与 3*3 卷积核运算（元素位 8 位无符号整数）。

故计算公式为：

$$C_{ij} = \sum_{i=1}^3 \sum_{j=1}^3 \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix} \otimes \begin{bmatrix} A_{(i,j)} & A_{(i,j+1)} & A_{(i,j+2)} \\ A_{(i+1,j)} & A_{(i+1,j+1)} & A_{(i+1,j+2)} \\ A_{(i+2,j)} & A_{(i+2,j+1)} & A_{(i+2,j+2)} \end{bmatrix}$$



2 主要模块

2.1 加法器

设二进制加法器第 i 位为 A_i, B_i , 输出为 S_i , 进位输入为 C_i , 进位输出为 C_{i+1} , 则有:

$$\bullet S_i = A_i \oplus B_i \oplus C_i \quad (1)$$

$$\bullet C_{i+1} = A_i \cdot B_i + A_i \cdot C_i + B_i \cdot C_i = A_i \cdot B_i + (A_i + B_i) \cdot C_i \quad (2)$$

令 $G_i = A_i \cdot B_i$, $P_i = A_i + B_i$, 则 $C_{i+1} = G_i + P_i \cdot C_i$

- 当 A_i 和 B_i 都为 1 时, $G_i = 1$, 产生进位 $C_{i+1} = 1$
- 当 A_i 和 B_i 有一个为 1 时, $P_i = 1$, 传递进位 $C_i + 1 = C_i$

因此 G_i 定义为进位产生信号, P_i 定义为进位传递信号. G_i 的优先级比 P_i 高。

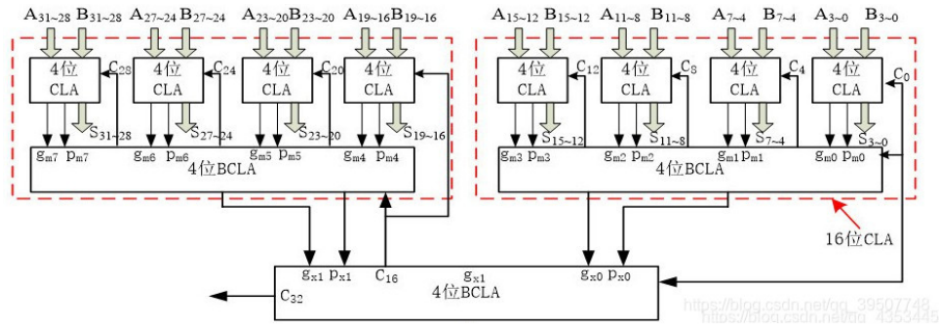
下面推导 4 位超前进位加法器。设 4 位加数和被加数为 A 和 B , 进位输入为 C_{in} , 进位输出为 C_{out} , 对于第 i 位的进位产生 $G_i = A_i \cdot B_i$, 进位传递 $P_i = A_i + B_i$, $i=0,1,2,3$ 。于是这各级进位输出, 递归的展开 C_i , 有

- $C_0 = C_{in}$
- $C_1 = G_0 + P_0 \cdot C_0$
- $C_2 = G_1 + P_1 \cdot C_1 = G_1 + P_1 \cdot G_0 + P_1 \cdot P_0 \cdot C_0$
- $C_3 = G_2 + P_2 \cdot C_2 = G_2 + P_2 \cdot G_1 + P_2 \cdot P_1 \cdot G_0 + P_2 \cdot P_1 \cdot P_0 \cdot C_0$
- $C_4 = G_3 + P_3 \cdot C_3$
 $= G_3 + P_3 \cdot G_2 + P_3 \cdot P_2 \cdot G_1 + P_3 \cdot P_2 \cdot P_1 \cdot G_0 + P_3 \cdot P_2 \cdot P_1 \cdot P_0 \cdot C_0 \quad (3)$
- $C_{out} = C_4$

由此可以看出, 各级的进位彼此独立产生, 只与输入数据 A_i 、 B_i 和 C_{in} 有关。

实现上述逻辑表达式 (3) 的电路称为超前进位部件 (Carry Lookahead Unit), 也称为 CLA 部件。通过这种进位方式实现的加法器称为超前进位加法器。

从公式 (1-3) 可知, 更多位数的 CLA 部件只会增加逻辑门的输入端个数, 而不会增加门的级数, 因此, 如果采用超前进位方式实现更多位的加法器, 从理论上讲, 门延迟不变。但是由于 CLA 部件中连线数量和输入端个数的增多, 使得电路中需要具有大驱动信号和大扇入门, 这会大大增加门的延迟, 起不到提高电路性能的作用。因此更多位数的加法器可通过 4 位 CLA 部件和 4 位超前进位加法器来实现, 如下图所示。将式 (1-3) 中进位 C_4 的逻辑方程改写为: $C_4 = G_{m0} + P_{m0} \cdot C_0 \quad (4)$
 C_4 表示 4 位加法器的进位输出, P_{m0} 、 G_{m0} 分别表示 4 位加法器的进位传递输出和进位产生输出, 分别为:



- $P_{m0} = P_3 \cdot P_2 \cdot P_1 \cdot P_0$
- $G_{m0} = G_3 + P_3 \cdot G_2 + P_3 \cdot P_2 \cdot G_1 + P_3 \cdot P_2 \cdot P_1 \cdot G_0$

将式 (4) 应用于 4 个 4 位先行进位加法器，则有：

- $C_4 = G_{m0} + P_{m0} \cdot C_0$
- $C_8 = G_{m1} + P_{m1} \cdot C_4 = G_{m1} + P_{m1} \cdot G_{m0} + P_{m1} \cdot P_{m0} \cdot C_0$
- $C_{12} = G_{m2} + P_{m2} \cdot C_8 = G_{m2} + P_{m2} \cdot G_{m1} + P_{m2} \cdot P_{m1} \cdot G_{m0} + P_{m2} \cdot P_{m1} \cdot P_{m0} \cdot C_0$
- $C_{16} = G_{m3} + P_{m3} \cdot C_{12} = G_{m3} + P_{m3} \cdot G_{m2} + P_{m3} \cdot P_{m2} \cdot G_{m1} + P_{m3} \cdot P_{m2} \cdot P_{m1} \cdot G_{m0} + P_{m3} \cdot P_{m2} \cdot P_{m1} \cdot P_{m0} \cdot C_0$ (5)

比较式 (3) 和式 (5)，可以看出这两组进位逻辑表达式是类似的。不过式 (3) 表示的是组内进位，式 (5) 表示的是组间的进位。实现逻辑方程组 (5) 的电路称为成组先行进位部件。

2.2 乘法器

乘法器采用移位相加原理。

移位相加原理：从被乘数最低位开始，依次乘以乘数，得到的结果向左移动若干位，最后将结果相加。

	被乘数:				X ₃	X ₂	X ₁	X ₀
×)	乘 数:				Y ₃	Y ₂	Y ₁	Y ₀
<hr/>								
					Y ₀ X ₃	Y ₀ X ₂	Y ₀ X ₁	Y ₀ X ₀
					Y ₁ X ₃	Y ₁ X ₂	Y ₁ X ₁	Y ₁ X ₀
					Y ₂ X ₃	Y ₂ X ₂	Y ₂ X ₁	Y ₂ X ₀
					Y ₃ X ₃	Y ₃ X ₂	Y ₃ X ₁	Y ₃ X ₀
<hr/>								
乘积:	Z ₇	Z ₆	Z ₅	Z ₄	Z ₃	Z ₂	Z ₁	Z ₀

3 明细表

模块名	模块输入			模块输出			模块描述
Conv_53	clk	时钟信号	1 位	out1l-	输出	8 位	用 3*3 的卷积核对 5*5 的输入矩阵进行运算结果为 3*3 的矩阵
	rst	复位信号	1 位	out33			
	A11-A55	输入矩阵	8 位				
	k11-k33	卷积核	8 位				
Conv	A11-A33	输入矩阵	8 位	out	输出	8 位	用 3*3 的卷积核对 3*3 的输入矩阵进行运算结果为 1*1 的矩阵
	k11-k33	卷积核	8 位				
multiplier	num1	被乘数	8 位	result	积	16 位	自制的快速乘法器
	num2	乘数	8 位				
shifterr-1bit	num1	被移数	16 位	result	移位结果	16 位	1 位右移移位器
	num2	被动判断	1 位				
shifterr-1bit	num1	被移数	16 位	result	移位结果	16 位	1 位左移移位器
	num2	被动判断	1 位				
shifterr-2bit	num1	被移数	16 位	result	移位结果	16 位	2 位左移移位器
	num2	被动判断	1 位				
shifterr-3bit	num1	被移数	16 位	result	移位结果	16 位	3 位左移移位器
	num2	被动判断	1 位				
shifterr-4bit	num1	被移数	16 位	result	移位结果	16 位	4 位左移移位器
	num2	被动判断	1 位				
shifterr-5bit	num1	被移数	16 位	result	移位结果	16 位	5 位左移移位器
	num2	被动判断	1 位				
shifterr-6bit	num1	被移数	16 位	result	移位结果	16 位	6 位左移移位器
	num2	被动判断	1 位				
shifterr-7bit	num1	被移数	16 位	result	移位结果	16 位	7 位左移移位器
	num2	被动判断	1 位				

模块名	模块输入			模块输出		模块描述
adder20	a	加数 1	20 位	S	和	20 位超前进位加法器模块
	b	加数 2	20 位			

模块名	模块输入			模块输出			模块描述
CLA_16	A	输入	16 位	S	和	16 位	16 位 CLA 部件
	B	输入	16 位	Px		1 位	
	C0	进位	1 位	Gx		1 位	
adder_4	A	输入	4 位	S	和	4 位	四位并行进位加法器
	B	输入	4 位	C4	进位	1 位	
	C0	进位	1 位	Pm		1 位	
				Gm		1 位	
CLA	C0		1 位	C1		1 位	4 位 CLA 部件
	G0		1 位	C1		1 位	
	G1		1 位	C2		1 位	
	G2		1 位	C3		1 位	
	G3		1 位	C4		1 位	
	P0		1 位				
	P1		1 位				
	P1		1 位				
adder	A	输入	32 位	S	和	32 位	一位全加器
	B	输入	32 位	Cout	进位	1 位	
	Cin	进位	1 位				

4 仿真测试

4.1 测试样例

输入：

卷积矩阵

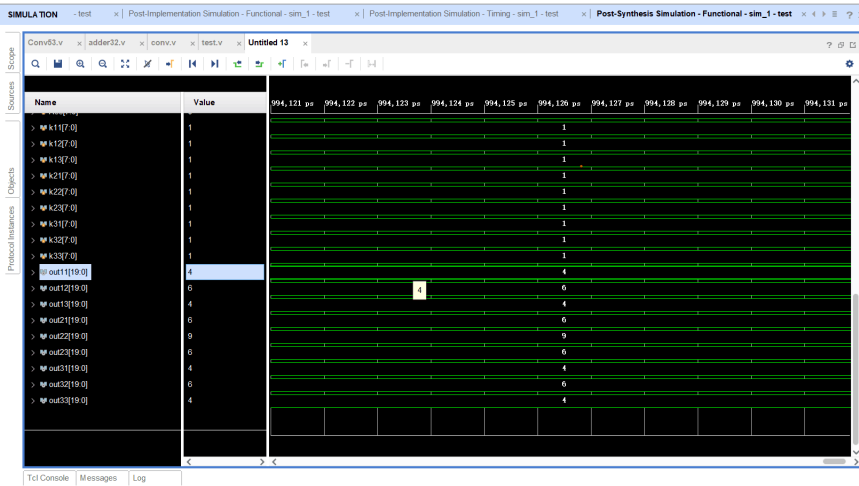
0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

卷积核

1	1	1
1	1	1
1	1	1

4.2 仿真结果

- 前仿



- 后仿



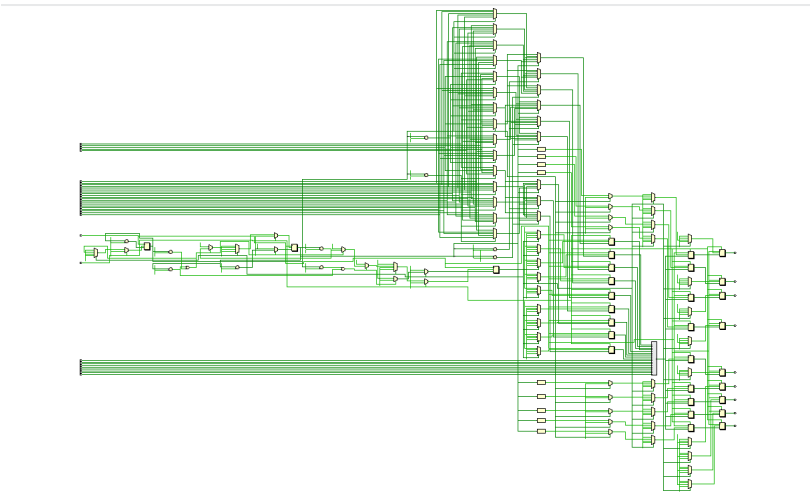
5 电路相关

5.1 复杂度

复杂度统计

模块	Cells	I/Oports	Nets
Conv_53	592	454	1158

5.2 电路图

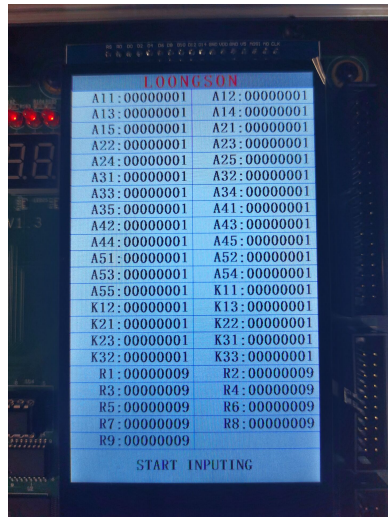


6 延迟分析

Unconstrained Paths - NONE - NONE - Setup												
Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock	Destination Cloc
Path 1	∞	17	18	16	k13[3]	out_r_reg[1][15]D	14.692	4.296	10.396	∞	input port clock	
Path 2	∞	17	18	16	k13[3]	out_r_reg[1][215]D	14.692	4.296	10.396	∞	input port clock	
Path 3	∞	17	18	16	k13[3]	out_r_reg[1][315]D	14.692	4.296	10.396	∞	input port clock	
Path 4	∞	17	18	16	k13[3]	out_r_reg[2][15]D	14.692	4.296	10.396	∞	input port clock	
Path 5	∞	17	18	16	k13[3]	out_r_reg[2][215]D	14.692	4.296	10.396	∞	input port clock	
Path 6	∞	17	18	16	k13[3]	out_r_reg[2][315]D	14.692	4.296	10.396	∞	input port clock	
Path 7	∞	17	18	16	k13[3]	out_r_reg[3][15]D	14.692	4.296	10.396	∞	input port clock	
Path 8	∞	17	18	16	k13[3]	out_r_reg[3][215]D	14.692	4.296	10.396	∞	input port clock	
Path 9	∞	17	18	16	k13[3]	out_r_reg[3][315]D	14.692	4.296	10.396	∞	input port clock	
Path 10	∞	17	18	16	k13[3]	out_r_reg[1][113]D	14.030	4.296	9.734	∞	input port clock	

7 FPGA 部署结果

7.1 样例 1



卷积矩阵

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

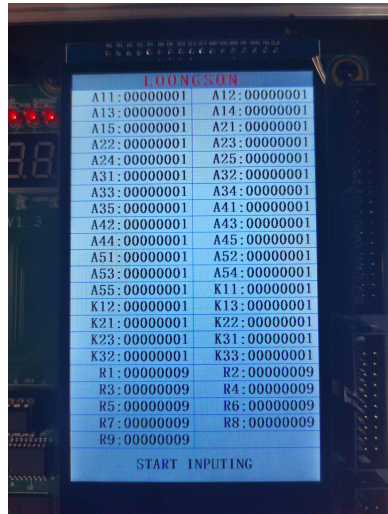
卷积核

1	1	1
1	1	1
1	1	1

卷积结果

9	9	9
9	9	9
9	9	9

7.2 样例 2



卷积矩阵

0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

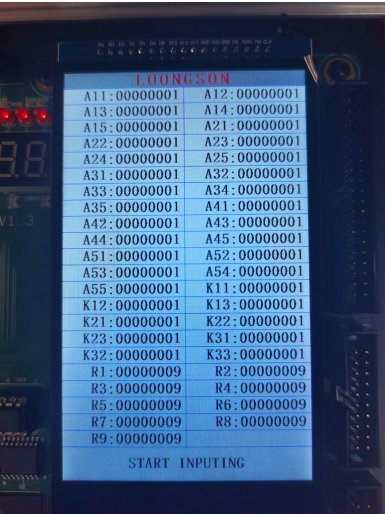
卷积核

1	1	1
1	1	1
1	1	1

卷积结果

4	6	4
6	9	6
4	6	4

7.3 样例 3



卷积矩阵

0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

卷积核

5	5	5
5	5	5
5	5	5

卷积结果

14	1E	14
1E	2D	1E
14	1E	14

8 组员分工

- 罗鹏 (320200946111): 卷积模块与 20 位超前进位加法器。
- 付俊博 (320200946011): 测试。
- 胡臻初 (320200946051): 8 位快速乘法器、快速移位器与约束文件
- 李卓奥 (320200942771): 答辩 ppt 制作。
- 王子豪 (320200945691): 设计文档与 ppt 展示。