## 实验一 运算器组成实验

## 实验目的

* 熟悉Logisim软件平台。
* 掌握运算器基本工作原理
* 掌握运算溢出检测的原理和实现方法；
* 理解有符号数和无符号数运算的区别；
* 理解基于补码的加/减运算实现原理；
* 熟悉运算器的数据传输通路。

## 实验环境

Logisim是一款数字电路模拟的教育软件，用户都可以通过它来学习如何创建逻辑电路，方便简单。 它是一款基于Java的应用程序，可运行在任何支持JAVA环境的平台，方便学生来学习设计和模仿数字逻辑电路。Logisim中的主要组成部分之一就在于设计并以图示来显示CPU。当然Logisim中还有其他多种组合分析模型来对你进行帮助，如转换电路，表达式，布尔型和真值表等等。同时还可以重新利用小规模的电路来作为大型电路的一部分。

<http://www.cburch.com/logisim/docs.html>

## 实验内容

1. **Logism实验**
2. 学习使用Logism工具栏上的功能
3. 学会使用子电路，并能将子电路放到main电路中使用
4. 学习使用时钟，并能使用时钟单步或自动运行
5. 学会使用分线器，理解线宽的概念
6. 学会使用隧道，学习使用探测器，了解logisim数据监测方法。
7. 熟悉按键、LED，数码管等基本输出设备

注（此部分要求可在作中学，相应部分在后续实验中均有要求，简单熟悉平台后可直接跳越到实验2）

1. **快速加法器设计**

利用基本逻辑门电路构造4位具有先行进位特征的快速加法器，并进行子电路封装。利用封装好的4位快速加法器构建32位组内先行进位，组间先行进位的加法器，并分析对应电路延迟。

1. **运算器封装试验**

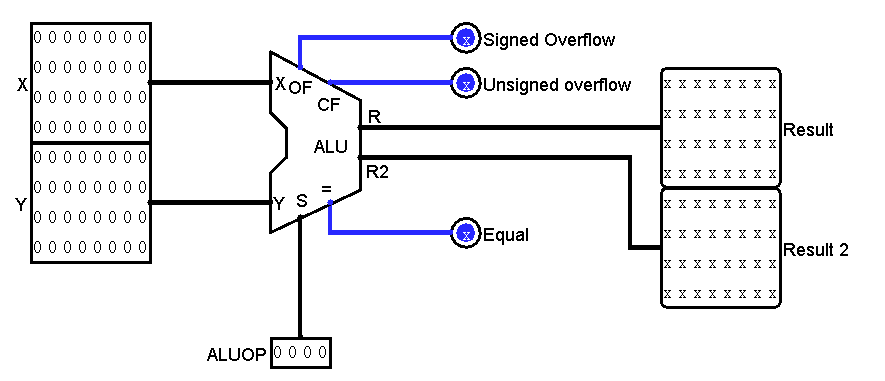
**构建32位运算器。**利用封装好的加法器以及logisim平台中现有运算部件**（禁用系统自带的加法器，减法器）**构建一个32位运算器，可支持算术加、减、乘、除，逻辑与、或、非、异或运算、逻辑左移、逻辑右移，算术右移运算，支持常用程序状态标志（有符号溢出OF、无符号溢出CF，结果相等Equal），运算器功能以及输入输出引脚见下表，在主电路中详细测试自己封装的运算器，在报告中分析该运算器的优缺点。

**表1. 芯片引脚与功能描述**

|  |  |  |  |
| --- | --- | --- | --- |
| 引脚 | 输入/输出 | 位宽 | 功能描述 |
| X | 输入 | 32 | 操作数X |
| Y | 输入 | 32 | 操作数Y |
| ALU\_OP | 输入 | 4 | 运算器功能码，具体功能见下表 |
| Result | 输出 | 32 | ALU运算结果 |
| Result2 | 输出 | 32 | ALU结果第二部分，用于乘法指令结果高位或除法指令的余数位，其他操作为零 |
| OF | 输出 | 1 | 有符号加减溢出标记，其他操作为零 |
| UOF | 输出 | 1 | 无符号加减溢出标记，其他操作为零 |
| Equal | 输出 | 1 | Equal=(x==y)?1:0, 对所有操作有效 |

**表2. 运算符功能**

|  |  |  |
| --- | --- | --- |
| ALU OP | 十进制 | 运算功能 |
| 0000 | 0 | Result = X << Y 逻辑左移 （Y取低五位） Result2=0 |
| 0001 | 1 | Result = X >>>Y 算术右移 （Y取低五位） Result2=0 |
| 0010 | 2 | Result = X >> Y 逻辑右移 （Y取低五位） Result2=0 |
| 0011 | 3 | Result = (X \* Y)[31:0]; Result2 = (X \* Y)[63:32] 有符号 |
| 0100 | 4 | Result = X/Y; Result2 = X%Y 无符号 |
| 0101 | 5 | Result = X + Y (Set OF/UOF) |
| 0110 | 6 | Result = X - Y (Set OF/UOF) |
| 0111 | 7 | Result = X & Y |
| 1000 | 8 | Result = X | Y |
| 1001 | 9 | Result = X⊕Y |
| 1010 | 10 | Result = ~(X |Y) |
| 1011 | 11 | Result = (X < Y) ? 1 : 0 符号比较 |
| 1100 | 12 | Result = (X < Y) ? 1 : 0 无符号比较 |



**图1.运算器封装示意图**

**（请直接在实验包中的alu.circ中构建，构建后的电路应可以直接在alutest.circ中使用，相应封装应正好连接对应电路），最终结果将在alutest.circ文件中由教师进行详细测试并自动评分，请各位同学在alutest.circ中详细测试自己的电路，可以申请多次测试，但每多一次测试总分扣减5%。**

**学生掌握技能**

* + 有符号无符号数加、减法溢出检测
  + 熟悉logisim 基本部件使用
  + 了解logisim的子电路的使用封装方法
  + 掌握隧道和探测器使用方法
  + 掌握分线器的使用方法
  + 掌握部件标注的方法
  + 掌握利用逻辑表达式、真值表自动生成电路的方法

1. **选作扩展实验（本次不选）。**

利用上述步骤封装运算器子电路，以及寄存器模块，构建一个可自动计算等差数列求和的运算器电路，（不允许使用其他功能模块，运算由时钟驱动，每一个时钟完成一次加法），可设置等差数列初始值以及等差值（只能通过引脚输入）。（求和的值利用16位进制数码管输出显示） （扩展部分，可加分）

如设置初始值为1，等差值为1，应该能自动完成1+2+3+4+5+6+7+8

如设置初始值为0，等差值为2，应该能自动完成2+4+6+8+10+12+14+16

## 实验步骤

**1、实验准备**

1) 复习有关运算器的内容，对数据通路的构成、数据在数据通路中的流动及控制方法有基本的了解。

2) 熟悉电路中各部分的关系及信号间的逻辑关系

3) 设计实验电路，画出各模块的图，注意各引脚的标注，节省实验的时间。

**2、实验步骤**

实验可按照自己设计的电路或参考电路按照搭积木的方式进行。先完成运算器的数据通路部分，在运算器部分能够正确完成各类运算的基础上，再增加累加器等其他部件。

## 结果提交

请将完成后的alu.circ文件按以下命名规范命名后作为实验结果提交给班级知道教师当场检查并归档。

* **专业命名规范**

信安 IS 物联网 IT 计算机 CS 卓越班 ZY ACM班 ACM

* **文件命名规范**

CS1201\_U201214795\_姓名\_alu.circ

## 实验报告要求

1) 实验目的；

2) 各模块的设计电路和系统的整体电路,对设计要进行详细的分析与说明；

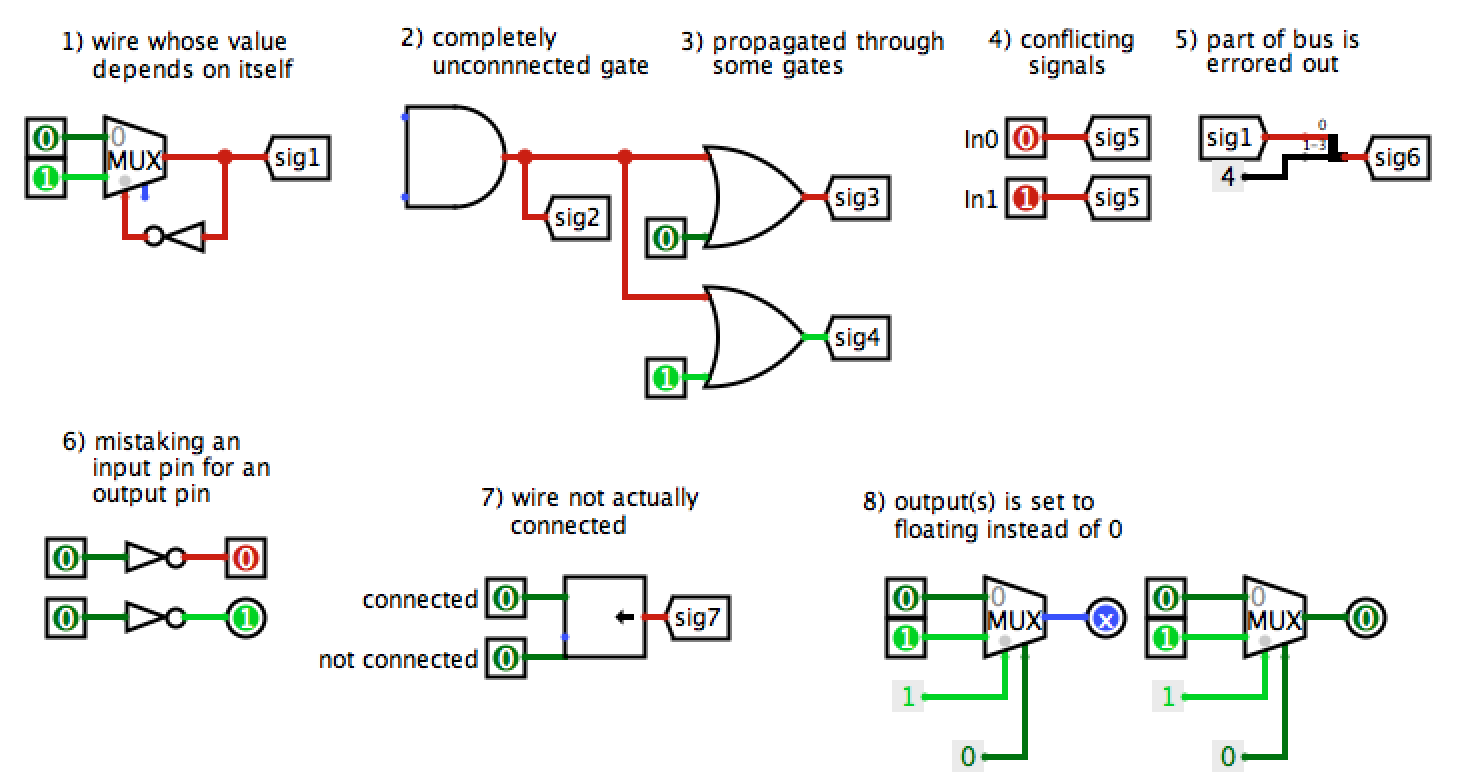
3) 实验结果的记录与分析；

4) 列出操作步骤及顺序,标出重要的开关控制端；

5) 实验收获和体会；

6) 实验中碰到的问题和解决的方法。

## 注意事项

* **不要对时钟信号进行门级操作，在实际电路中这是非常糟糕的设计，会导致一系列严重的故障，如险象。**
* **大区域拷贝粘贴移动电路可能会导致logisim崩溃，请随时ctrl+s保存电路。**
* **Logisim工具栏器件可以改变其默认属性，可以根据需要修改。**
* **红色信号线肯定是明显的错误，通常在复杂电路中会出现，调试的时候应注意是否出现以下情况引起：**