# 实验报告模板

华南理工大学

《计算机组成原理》课程实验报告

实验题目： 运算器

姓名： 陈映松 学号： 202130442563

班级： 21计科(2)班 组别：

合作者：

指导教师： 张齐

|  |
| --- |
| **实验概述** |
| 【实验目的及要求】  实验目的：   1. 理解补码的概念和补码加减法运算器的原理。 2. 了解加法器先行进位的原理和电路结构。 3. 掌握串行和并行的原码乘法器和除法器结构。 4. 掌握补码乘法器的结构   实验要求：   1. 利用digiblock软件实现补码加减法运算 2. 利用digiblock软件实现先行进位加法器 3. 利用digiblock软件实现串行原码乘法器和除法器 4. 利用digiblock软件实现并行原码乘法器和除法器 5. 利用digiblock软件实现补码乘法器   【实验环境】  Proteus 8.1  Digiblock  Java配置环境 |
| **实验内容** |
| 【实验过程】   1. 实验步骤：（“实验步骤”）;   **1：补码加减法运算**   1. 补码加减法运算器如图所示，其基本原理是把减法运算变成加法运算：若S=0，则B路的异或门等同直通，S=[A+B]补=[A]补+[B]补;若 S=1，则 B 路的异或门等同反向器，再最低位加上 S=1，即 B 取反且末位加 1（等同对 B 求补），S=[A-B]补=[A]补+[-B]补。   (2) 补码运算中，位数不等的两数相加，须扩展短数的符号位使其与长数的位数相等，如下图 2.1.3 所示。图中端口 A 是 8 位，端口 B 是 4 位，所以端口 B 须借助符号位扩展器或其等效电路扩展符号位（符号位为 0 则扩展的高位补 0，符号位为 1 则扩展的高位补 1）。  (3) 根据指导书上图2.1.3的符号位扩展对齐位数的补码加法电路图，依据上述的描述，设置输入端的响应数据或控制相应输入端的电平；观察输出端的电平或数据，并将结果填入表中。  2.先行进位加法器  （1）上述图 2.1.2 所述的串行加减法运算器电路中，耗时最长的是串行进位链。如 下列表达式中的深色部分所示，每一级的进位 Ci 都必须等待前一级的 Ci-1，加法器的位数越长，则串行进位链的时间越长。然而，如下列表达式所示，C4 进位完全可以通过层层递推，变成一个 C0 的函数，其中除了 C0 以外的所有变量都可以由已知的 A0~A3 和 B0~B3 求出（G0~G3 和 P0~P3 是中间变量）。因此，下列 C4=f(C0)的展开表达式对应的 4 位先行进位电路原理图如下图 2.2.1 左所示，图 2.2.1 右则是包含了 4 位先行进位电路和 Pn/Qn 中间变量生成电路的 4 位先行进位加法器框图。从图 2.2.1 中可以看出，Pn/Qn 中间变量生成仅需 3T 时间（最长路径是异或门，即 Pn= (Xn⊕Yn)；而 4 位先行进位电路仅需 2T（与门和或门两级），总共 5T 即可求出 4 位加法器的进位 C4，而传统的串行加法器电路则需要耗时2n+9T=17T(n=4)。 Si = Ai⊕ Bi⊕Ci Ci+1=Ai·Bi+(Ai ⊕Bi)·Ci  令 Gn= An·Bn，Pn= (An⊕Bn) 则 Cn+1=Gn+PnCn Si = Pi⊕Ci  有 **C1 = G0+P0C0** **C2 = G1+P1C1** = G1+P1G0+P1P0C0  **C3 = G2+P2C2** = G2+P2G1+P2P1G0+P2P1P0C0  **C4 = G3+P3C3** = G3+P3G2+P3P2G1+P3P2P1G0+P3P2P1P0C0  (2) 串行和先行进位的 4 位加法器电路如下图 2.2.2 所示，图左是与前述图 2.1.2 类似的串行进位加法器电路，图右则是与上图 2.2.1 原理框图对应的 4 位先行进位加法器电路。  (3) 上图 2.2.1 的先行进位电路一般只能做 4 位，更多位数的先行进位电路太复杂，扇出和时延不可控。因此，一般 n 个 4 位先行进位加法器串联形成 4\*n 位加法器，同时在每个4位先行进位加法器上配置进位旁路电路，如下图 2.2.3 所示。图中的 LCS 电路是图 2.2.1右所示的 4 位先行进位加法器，配置进位旁路电路后就是图下方的 block 电路，4 个 4 位 block电路级联形成图下方的 16 位先行进位加法器。进位旁路电路的原理如下：先行进位电路 C4=G3+P3G2 +P3P2G1 +P3P2P1G0 +P3P2P1P0C0若 P3P2P1P0=1，因为 Pn= (An⊕Bn)所有的 An 与 Bn 皆互异所以，Gn= An·Bn =0，则 C4=C0 ，进接传递若 P3P2P1P0=0，则 C4=G3+P3G2 +P3P2G1 +P3P2P1G0,与前一级的进位 C0 无关.  (4) 与上图 2.2.3 中配置进位旁路的 4 位先行进位加法器原理图对应的电路图如下图 2.2.4左所示。图左的电路即是图右中的子电路，四个子电路级联，就形成了 16 位先行进位加法器电路图，如下图 2.2.4 右所示。  (5) 根据图 2.2.4 的加法器电路图，设置输入端的相应数据或控制相应输入端的电平；观察输出端的电平或数据，并将结果填入表 2.2 中。  **3.串行原码乘法器和除法器**  （1）原码乘法的手工计算如下图 2.3.1 左所示（4bit 原码相乘），对应每 1 位乘数求得 1 项位积，并将位积逐位左移，然后将所有的位积一次相加，得到最后的乘积。而原码乘法的机 器算法则是加法器级联的原码二叉树乘法算法，电路如下图 2.3.1 右所示（8bit 原码相乘）： 从乘数的最低位开始，每次根据乘数位得到其位积，乘数位为 0，位积为 0，乘数位为1，则位积为被乘数；用原部分积右移 1 位加上本次位积，得新部分积；初始部分积为 0。  （2）上图 2.3.1 中，二叉树乘法每级的加法器逐渐增多，导致中间电路的面积和延迟太大，所以，原码串行乘法思路如下图 2.3.2 所示：部分积左移改为右移，循环相加。  （3）原码串行乘法器如下图 2.3.3 所示：与原码二叉树乘法器电路相比，该电路把加法器级联改为循环使用加法器，以时间换取空间；而部分积改为右移使得加法器位数保持稳定，以利于复用加法器。  （4）原码除法的手工计算如下图 2.3.4 左所示（定点小数 0.10010/0.1011），除法的原理就是用除数的{1/2, 1/4,1/8…}倍数去不断减被除数，如果余数为正就继续减，如果余数为负数，则不减当前倍数。但恢复余数的步骤计算机处理比较繁琐，所以原码除法的计算机计算如下图 2.3.4 右所示（定点小数 0.101001/0.111，[x]补＝0.101001，[y]补＝0.111，[–y]补＝1.001），采取加减法交替的不恢复余数除法算法：余数为正时，商上 1，求下一位商的办法是余数左移一位，再减去除数；当余数为负时，商上 0，求下一位商的办法是余数左移一位，再加上 除数。注意：若最后一次上商为 0，而必须得到正确余数，则在这最后一次仍需恢复余数。  （5） 上图 2.3.4 中，余数末位补 0 后不断减去除数右移的值，导致每级的加法器位数逐渐增多，直到加法器的位数必须是被除数的两倍，导致中间电路的面积和延迟太大，原码串行除法的思路如下图 2.3.5 所示：除数右移改为余数左移，循环相减。  （6） 原码串行除法器如下图 2.3.6 所示：与原码串行乘法器电路类似，该电路把减法级联改为循环使用加法器（补码），以时间换取空间；而余数改为左移使得加法器位数保持稳定，以利于复用加法器。  （7） 根据图 2.3.6 的原码串行除法器电路图，设置输入端的相应数据或控制相应输入端的电平；观察输出端的电平或数据，并将结果填入表 2.3 中  4.并行原码乘法器和除法器  (1) 如下图 2.4.1 左所示的原码乘法手工计算过程（4bit 原码相乘）中，位积的每一位 XiYj都可以用一个与门实现。而行波阵列乘法器结构图如下图 2.4.1 右所示（4bit 原码相乘），其中每一位的相加均可以使用一个全加器来实现。  (2) 测试电路如下图 2.4.2 所示，图中给出了 4 位无符号乘法器及其等效的原码行波阵列乘法器电路，图中的 Test 部件是无符号乘法器的测试用例。  (3) 如下图 2.4.3 左所示的原码行波阵列乘法器示意图中，每次相加都必须等待串行加法器的进位链，造成时延较大。倘若将每位的全加器的进位输出及和输出都算作下一次相加的输入,则取消了同级加法器各位之间的进位等待，即原码 Walace 树阵列乘法器的原理，其示意图如下图 2.4.3 右所示。  (4) 8 位原码乘法（10100111×11011001）的行波阵列乘法器求解过程和 Walace 树阵列乘法器求解过程分别如下图 2.4.4 左和右所示  (5) 测试电路如下图 2.4.5 所示，图中给出了 4 位无符号乘法器及其等效的原码 Walace树阵列乘法器电路，图中的 Test 部件是无符号乘法器的测试用例。  (6) 计算机不恢复余数除法计算过程（定点小数 0.101001/0.111，[x]补＝0.101001，[y]补 ＝0.111，[–y]补＝1.001）如下图 2.4.6 左所示。其加减交替的原理可以由可控加减单元 CAS组成的阵列结构实现，如下图 2.4.6 右所示（6 位被除数和 3 位除数，商为 3 位）。其中每一 级 CAS 单元受上一级 CAS 单元的进位影响，做加法/减法运算（进位 0/1），同时本级CAS单元的进位即对应的商位。  (7) 测试电路如下图 2.4.7 所示，图中给出了如上图 2.4.6 右所示结构图对应的原码阵列除法器电路。  (8) 根据图 2.4.7 的原码阵列除法器电路图，设置输入端的相应数据或控制相应输入端的电平；观察输出端的电平或数据，并将结果填入表中。  **5.补码乘法器**  （1）如下图 2.5.1 左所示，重用原码阵列乘法器电路，绝对值电路求补其输入和输出是最直观方法实现补码乘法运算。下图 2.5.1 右所示：5 位有符号乘法器和绝对值转换后重用4位无符号数乘法器的等效电路相比较，图中的 Test 部件是 5 位有符号乘法器的测试用例。  （2）如下图 2.5.2 左下所示的手工补码乘法计算过程（01101×11011）中，符号位和有符号位参与的位积标注()，表示位权相反，该位积必须执行减法运算，即“-1”（仅单位运算，位权取反即可）。所以，基于位权的补码阵列乘法器（即 Baugh-Wooley 阵列乘法器）结构图如下图 2.5.2 右所示（16 位有符号数相乘）。蓝色单元是带有普通位积（与门）的全加器，绿色单元是带有位权相反位积（与非门）的全加器，为了压缩电路规模，部分积右移。  （3） 测试电路如下图 2.5.3 所示，图中给出了 4 位有符号乘法器（运算结果 10 位，双符号位）及其等效 Baugh-Wooley 阵列乘法器电路，图中的 Test 部件是有符号乘法器的测试用例。  （4）根据图 2.5.3 的基于位权的补码阵列乘法器电路图，设置输入端的相应数据或控 制相应输入端的电平；观察输出端的电平或数据，并将结果填入表中。   1. 实验数据：   **实验1：**    **实验2**    **实验3：**    **实验4：**    **实验5：** |
| **小结** |
| 这是计算机组成原理课程的第二次实验，依然是由五个小实验组成。这五个小实验也是一脉相承，内容相关性很强。这次依然遇到了实验一中同样的情况：理论课程上我们只学习了加减法器的构成与运行原理，而乘法器和除法器未涉及。因此跟上次一样，为了顺利完成这次实验，我也是借鉴了许多资料，然后认真阅读指导书上内容，一步一步操作软件上已有的电路，调试参数并填写表格，记录数据。不得不说这次的实验还是有一定难度的，但最终我还是顺利完成了此次实验。经历了实验二，感觉自己又学到了许多知识！ |
| **指导教师评语及成绩** |
| 评语：  成绩：           指导教师签名：                                                 批阅日期： |