



**汇 编 与 接 口 课 程 设 计**

接口控制器设计团队实验报告

|  |  |
| --- | --- |
| 学 院 | 计算机学院 |
| 专 业 | 计算机科学与技术 |
| 指导老师 | 王娟 |
| 组 长 | 杨训迪 |
| 组 员 | 丁颖 肖成文 周雨佳 |
| 组长联系方式 | 18810783722 |

二O二O年九月

目 录

[第一章 设计题目 1](#_Toc52742769)

[第二章 组员分工 1](#_Toc52742770)

[第三章 设计目的 1](#_Toc52742771)

[第四章 设计环境 2](#_Toc52742772)

[第五章 设计原理及内容 2](#_Toc52742773)

[5.1 MIPS汇编测试 2](#_Toc52742774)

[5.2 VGA接口实现 3](#_Toc52742775)

[第六章 设计步骤 5](#_Toc52742776)

[6.1 MIPS汇编测试 5](#_Toc52742777)

[6.1.1 设计思路 5](#_Toc52742778)

[6.1.2 具体代码及程序解释 5](#_Toc52742779)

[6.1.3 测试结果 8](#_Toc52742780)

[6.2 VGA接口实现 8](#_Toc52742781)

[6.2.1 设计思路 8](#_Toc52742782)

[6.2.2 具体代码及程序解释 10](#_Toc52742783)

[6.2.3 测试结果 13](#_Toc52742784)

[第七章 问题及解决方法 13](#_Toc52742785)

[7.1 MIPS汇编测试 13](#_Toc52742786)

[7.2 VGA接口实现 14](#_Toc52742787)

[第八章 心得体会及总结 14](#_Toc52742788)

[第九章 参考文献 15](#_Toc52742789)

# 设计题目

接口控制器设计

# 组员分工

小组分工见表2.1 小组分工。

|  |  |
| --- | --- |
| **组员** | **分工任务** |
| 杨训迪 | * 41条指令的数据通路和控制逻辑的设计 * 跳转模块，控制模块，存储模块的verilog编写 * 代码整合，top文件，41条指令的单条测试 * 流水线CPU实验报告的部分编写 * soc文件，数据存储器IP核(未完成) |
| 丁颖 | * vga实现和mips实现 * vga与cpu接口对接 * vga下板演示 * 汇编与接口实验报告的部分编写 |
| 肖成文 | * 冒险处理的数据通路和控制逻辑的设计与实现 * 寄存器堆模块，ALU模块的verilog编写 * 冒险处理板块功能的测试，41条指令的综合测试及相关结果的数码管显示 * 流水线CPU实验报告的部分编写 |
| 周雨佳 | * 41条指令汇编测试代码编写 * CP0实现异常和中断处理（未整合） * PPT制作 * 汇编与接口实验报告的部分编写 |

表2.1 小组分工

# 设计目的

1. 设置接口控制器，实现cpu流水线汇编测试
2. 掌握mips的实现和编写
3. 掌握vga的原理和实现

# 设计环境

|  |  |
| --- | --- |
| 操作系统 | Windows 10 |
| 编程语言 | Verilog |
| 编辑器 | Vivado 2019 |
| 汇编语言 | MIPS |
| 汇编程序编辑器 | Mars |

# 设计原理及内容

## MIPS汇编测试

利用MARS汇编模拟器，针对小组设计的MIPS五级流水线CPU来设计一系列包括add、sll、and等41条MIPS指令的MIPS汇编程序（包括一个全覆盖的汇编测试程序）。

实验中使用的MIPS指令子集具体如下:

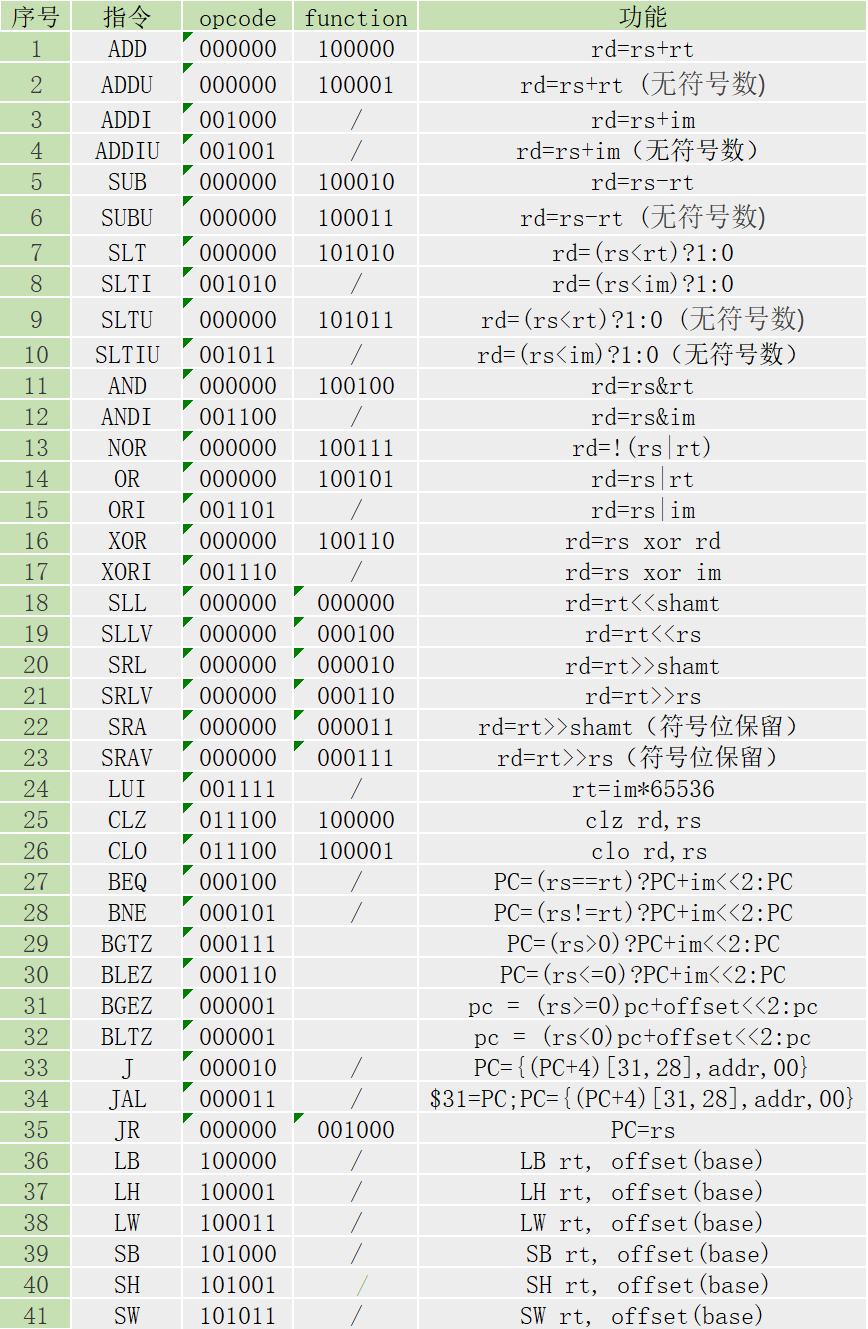


图5.1-1 全指令图

## VGA接口实现



图5.2-1 VGA显示模式图

如图5.2-1所示，a同步脉冲，b显示后沿，c显示时序段，d显示后沿。这四段组成了一个完整的时序，VGA 的行时序是以像素为单位的，场时序是以行为单位的。

计算时钟频率。用800\*525\*60，算出需要分频25Mhz。

显示器的扫描方式：逐行扫描。从左边追点扫描到右边，扫描一行，然后电子束回到此行扫描的起始位置的下一行继续扫描。每行结束时，用行同步信号进行同步；当扫描完所有的行，形成一帧，用场同步信号进行场同步，并使扫描回到屏幕开始扫描的位置,开始下一帧扫描

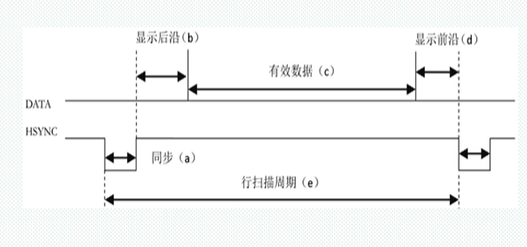


图5.2-2 VGA行扫描周期图

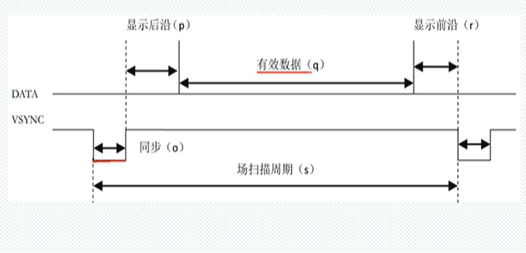


图5.2-3 VGA列扫描周期图

VGA时序对同步时间、显示后沿时间、视频有效时间和显示前沿时间有特定要求，一般为q有效数据期间段

# 设计步骤

## MIPS汇编测试

本次实验中，我们编写了一系列汇编测试程序来进行五级流水线CPU测试。在报告中，以一个完整的全覆盖汇编测试程序为例进行简述，其余的轻量测试程序则不再赘述。

### 设计思路

该测试代码的主要功能是：不使用乘除法指令的前提下，对输入的参数a，b进行除法运算并得到运算后的商和余数。代码中a、b被分别设为129和5。

### 具体代码及程序解释

该汇编测试程序的具体代码摘取见下图。

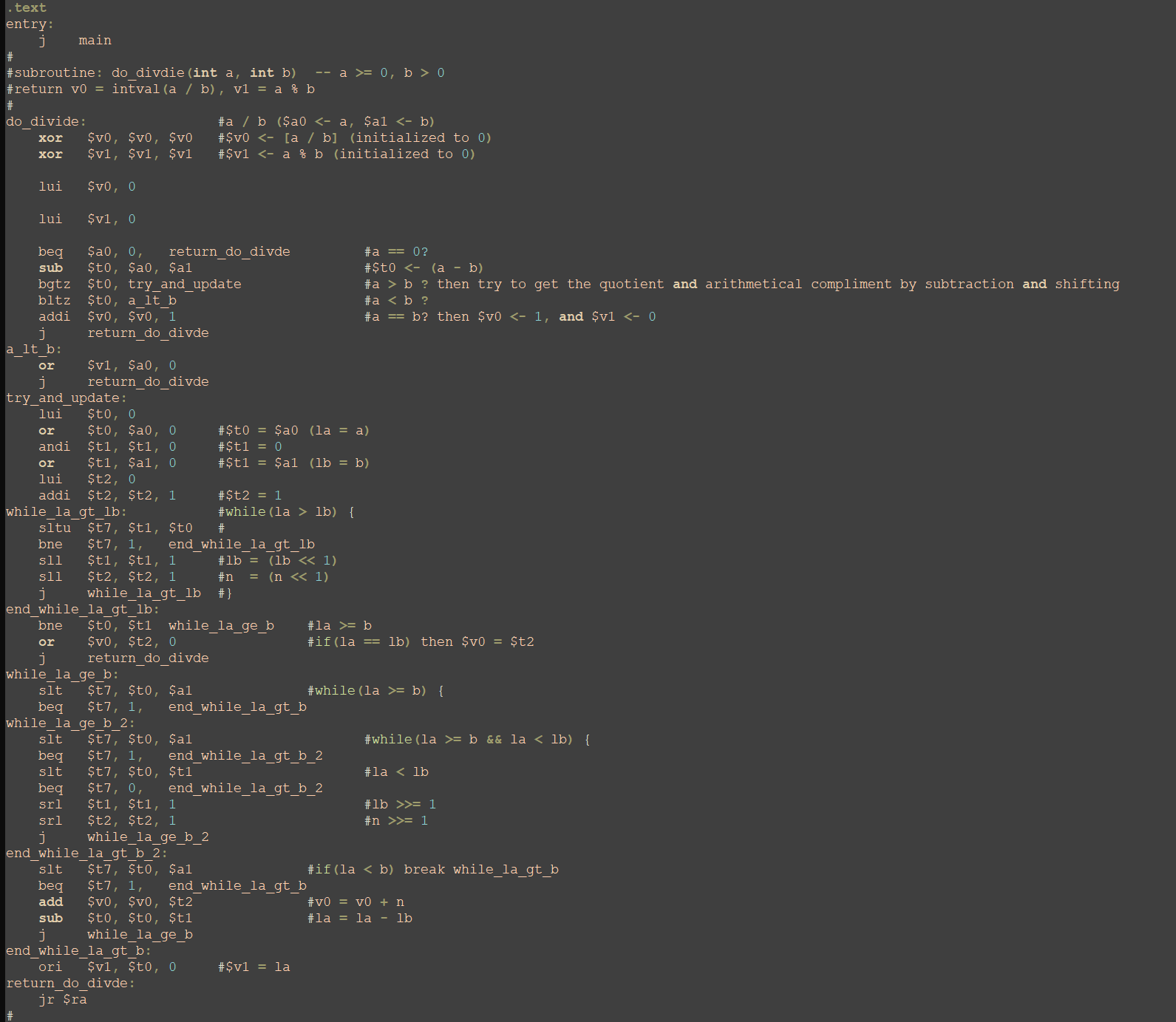


图6.1.2-1 汇编测试代码图

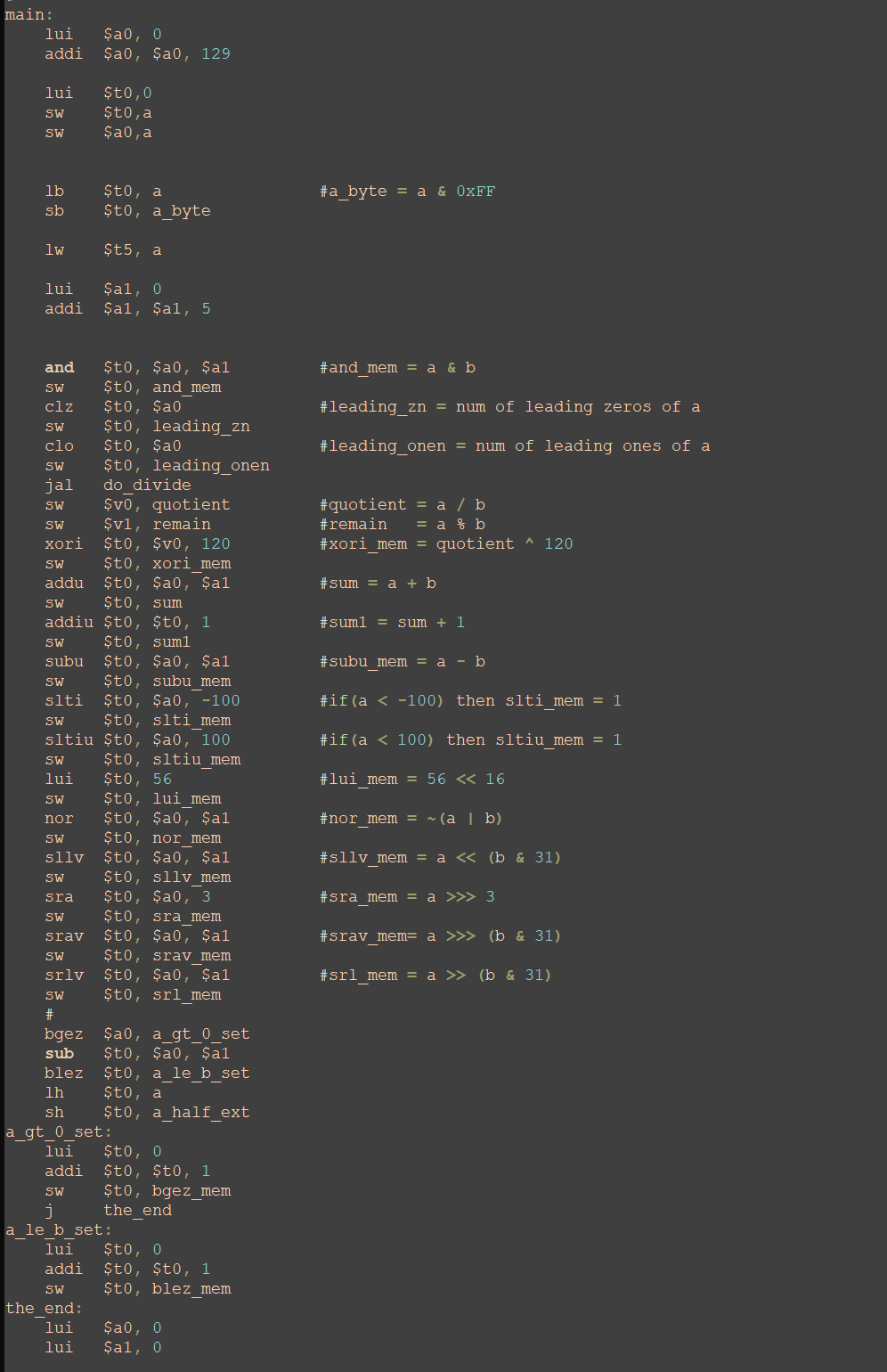


图6.1.2-2 汇编测试代码图

该程序中包括了ADD、ADDU、ADDI、ADDIU、SUB、SUBU、SLT、SLTI、SLTU、SLTIU、AND、ANDI、NOR、OR、ORI、XOR、XORI、SLL、SLLV、SRL、SRLV、SRA、SRAV、LUI、CLZ、CLO、BEQ、BNE、BGTZ、BLEZ、BGEZ、BLTZ、J、JAL、JR、LB、LH、LW、SB、SH、SW共计41条指令。

程序可被分为以下三部分。

1. 根据.data段，初始化对应的寄存器（但是并不对其进行赋值，否则会出错）。
2. 进行循环减法计算来实现除法计算。每次循环中，比较一次被除数与除数的大小来进行跳转以实现不同的功能。通过商的不断+1以及不能进行减法后的余数赋值来得到最终结果。
3. 商和余数变量赋值并进行其他指令的具体测试。

该程序中，被除数对应$a0寄存器，除数对应$a1寄存器，商被保存在$v0寄存器，余数被保存在$v1寄存器。

### 测试结果

在导出该汇编程序对应的机器码后，对CPU工程中的指令存储器地址进行修改来进行CPU的仿真测试。具体的实验结果见下图。

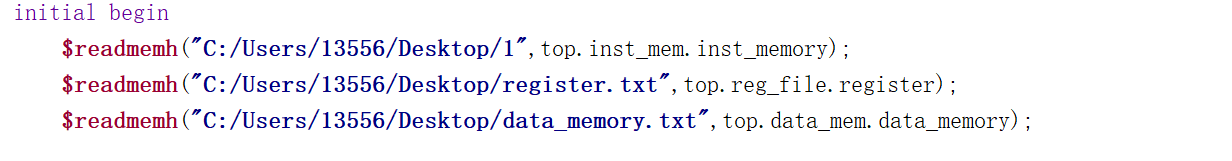


图6.1.3-1 CPU机器码地址图



图6.1.3-2 汇编测试仿真波形结果图

由仿真波形图可见2号寄存器和3号寄存器对应的值分别为0x19和0x4，即$v0寄存器和$v1寄存器对应的值分别为25和4。而测试程序中的被除数$a0被赋值为129，除数$a1被赋值为5，且129/5=25，129%5=4。从而可知测试结果正确，在该测试程序下CPU运行正确。

其他轻量测试程序的测试结果不再赘述。

## VGA接口实现

本次实验中，我们编写了一个移动的木条来进行实验。在以下报告以其中关键代码为例来进行简述。

### 设计思路

该代码实现的主要功能是：通过vga接口展示不断移动的木条实验

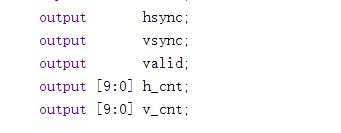


图6.2.1-1 VGA代码参数图

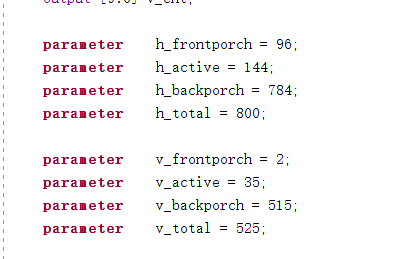


图6.2.1-2 VGA代码参数图

其中各个参数分别为行同步信号，场同步信号，行显示后延和场显示后延等h\_frontporch，h\_active，h\_backporch，h\_total，v\_frontporch，v\_active，v\_backporch，v\_total

程序可以分为以下三个部分

### 具体代码及程序解释

#### 分频

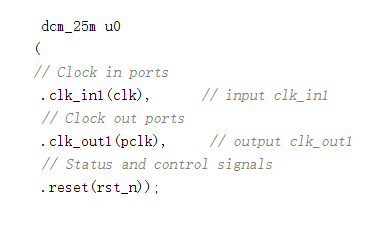


图6.2.2-1 VGA代码分频图

输入时钟信号，输出分频信号，通过vivado中内置的ip核实现

#### 计算信号

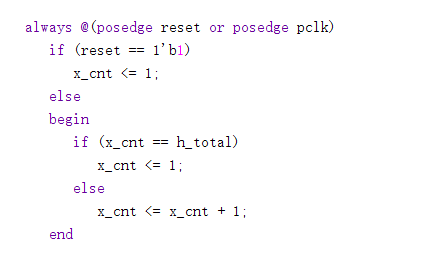


图6.2.2-2 VGA计算行信号x代码图

扫描一行，每次x+1

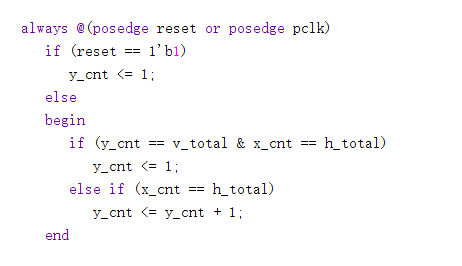


图6.2.2-3 VGA计算列信号y代码图

当扫描完所有行形成一帧后，y+1

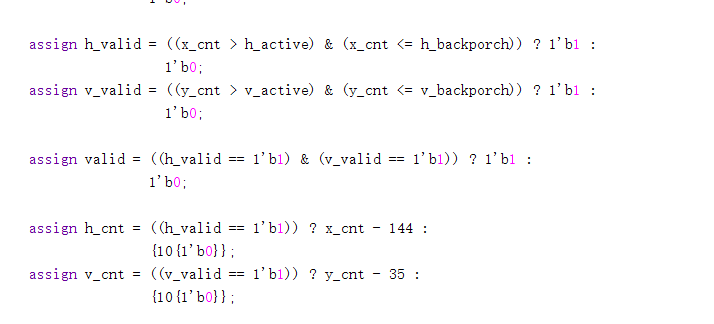


图6.2.2-4 VGA计算信号代码图

判断信号是否有效和像素的位置

#### 根据信号给像素点不同的颜色

设置两个颜色，一个color1，一个color2，当电子束在规定面积范围则赋值color2，否则赋值背景颜色color1



图6.2.2-5 VGA计算范围图

规定面积范围如图

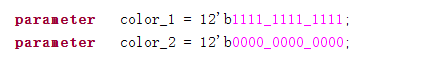


图6.2.2-6 VGA规定像素颜色图

color1和color2设置如图

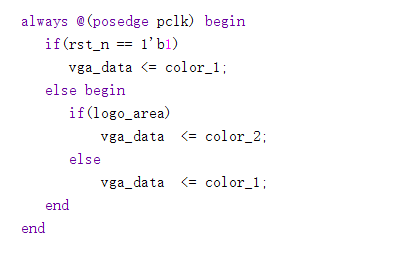


图6.2.2-7 VGA规定像素颜色图

根据分频时钟信号，当满足像素区域时为color2，否则赋值背景颜色color1

设置一个延时使能信号move\_en，当信号来时则将x+1

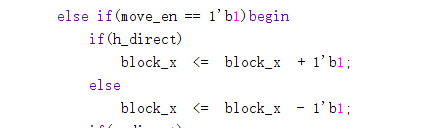


图6.2.2-8 VGA相关像素颜色移动轨迹图

然后将颜色通过vga\_r,vga\_g,vga\_b信号传递出去

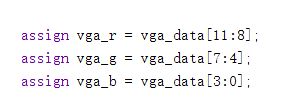


图6.2.2-9 VGA像素颜色rgb图

### 测试结果

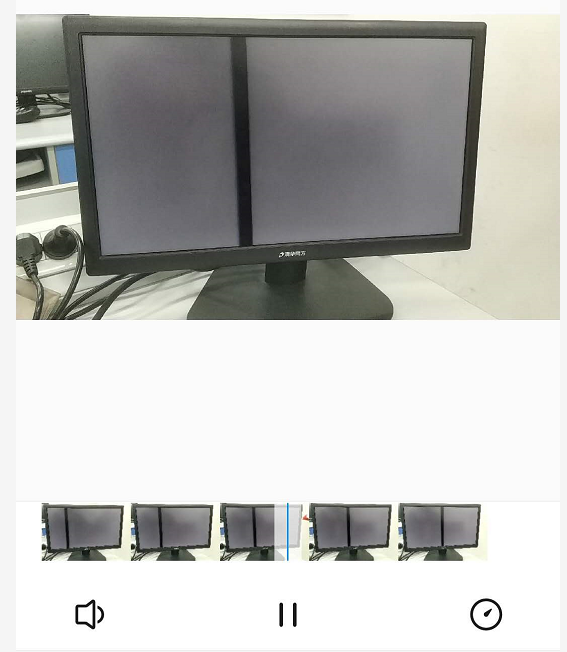


图6.2.3-1 VGA测试结果图

由上图截图可知，获得了一个向右移动的木条，背景色和木条颜色不同

# 问题及解决方法

## MIPS汇编测试

在本次流水线CPU汇编测试程序设计过程中，主要遇到了两个困难。

1. 问题1：当完成41条完整分条测试程序后发现测试过于分散，并不能够实现在一个文件完整测试CPU，并且不能够直观地得到运行结果。

解决办法：重新编写41条全覆盖有意义的测试程序以及一系列能够覆盖全部指令的轻量测试程序，来得到直观测试运行结果。

1. 问题2：当完成41条全覆盖有意义测试程序后发现，在MARS模拟器上能够正常运行的程序并不能够在流水线上正确运行。

解决办法：通过仔细分析仿真波形图来找出是在哪些指令处发生了运行错误，从而进行代码修改。后来发现，在数据段不能直接进行变量赋值，并且存储器初始值都是未知的不能够通过异或来进行初始化赋值为0。

## VGA接口实现

在本次实验中，主要遇到了以下问题

1. 关于像素点坐标的计算

一开始直接使用x\_cnt和y\_cnt忘记减去行场前沿显示信号等所需时间，后x\_cnt-144,y\_cnt-35才是正确的值

1. 关于行扫描和列扫描

对于计组之前所学的行扫描和列扫描不太弄得明白，后续经过实验，分清楚了行扫描和列扫描的计算

# 心得体会及总结

在本次实验中，我遇到了一些困难。比如在分别作业的情况下难以将编写好的CP0模块整合进另一个工程中等，但这也更让我体会到了实验过程中团队合作与交流沟通的重要性。在我之后的小组合作中我也会更加注重早期的合作与交流，提高团队协作能力，从而更好地完成实验项目。

在本次实验过程中，我也有了很多收获。我更好地理解了CP0模块处理异常与中断的具体流程与实现方法；我学会了如何利用有限的41条MIPS指令编写出有意义测汇编测试程序；我知道了自己编写的CPU与MARS模拟器运行是存在很多区别的，也明白了要如何通过观察仿真结果来找出区别并进行适当的代码修改；我也更好地理解了流水线CPU的整体结构以及指令运行的具体流程。

在本次实验中，虽然我遇到了一些问题，比如一开始使用mips将机器码存入rom中，关于vga的modul多了一个关于data和address的输入和输出，将计算出来的address穿给cpu从ram中读取data再传回来，但后续出现一系列问题所以没能完成。同样在此次实验中我也有很多收获，知道了关于vga的原理，知道了行扫描和列扫描信号，也明白了一些计组中知识点的实际操作时的应该如何把知识用于实践。

本次实验让我能够将之前课程学习到的知识进行具体的实际应用，极大地提升了我的学习实践能力和综合运用知识的能力。并且在实验过程中通过小组合作与交流，我的团队协作能力也提升了很多。在今后的学习生活中，我也应当秉承在实践中学习的精神更加努力！

# 参考文献

1. David A.Patterson, John L. Hennessy.郑纬民等译.计算机组成与设计 硬件/软件接口[M].机械工业出版社.
2. 雷思磊.自己动手写CPU[M]. 北京:电子工业出版社.