MIPS

作 者1, 作 者2，作 者3

（1.班级 学号；2.班级，序号；3.班级，序号）

1. 引言
2. 课程任务和实验要求
   1. 实验目的

MIPS 处理器是一种较为经典的处理器架构，之前在《数字逻辑与处理器基础》理论课上围绕着单周期、多周期、流水线三种模式进行了初步的学习。在《数字逻辑与处理器基础实验》 夏季学期的课程中，我们将围绕理论课学习内容，使用 Verilog 来实现一种 32 位的 MIPS 处理器设计， 并实现单周期和流水线两种模式。同时，我们也将实现 MIPS 指令编译器，将MIPS汇编指令编译为机器代码，从而使得处理器能够进行处理。在接下来的内容中，我们将重述课程的任务及要求、介绍我们小组的分工、介绍汇编代码的设计和编译器的设计、ALU 的设计、单周期 MIPS 处理器的设计、流水线 MIPS 处理器的设计。并在原理介绍之后介绍该处理器的仿真及硬件调试情况。最后，我们将对我们的实验进行总结。相关源代码在文件中给出。

* 1. 实验要求

1) 设计一个 32 位的 ALU，实现基本的算术、逻辑、关系、位与移位运算；

2) 设计一个单周期 MIPS 处理器，要求完成以下 指令：

a) 空指令：nop (0x00000000, 即 sll $0,$0,0)

b) 存储访问指令：lw, sw, lui；

c) 算术逻辑指令：add, addu, sub, subu, addi, addiu, and, or, xor, nor, andi, sll, srl, sra, slt, slti, sltiu；

d) 分支和跳转指令 branch(beq, bne, blez, bgtz, bltz) 和 jump(j, jal, jr, jalr)；

e) 其他指令可以根据情况自行添加。

3) 在单周期 MIPS 处理器的基础上，设计一个 5 级流水线的 MIPS 处理器。

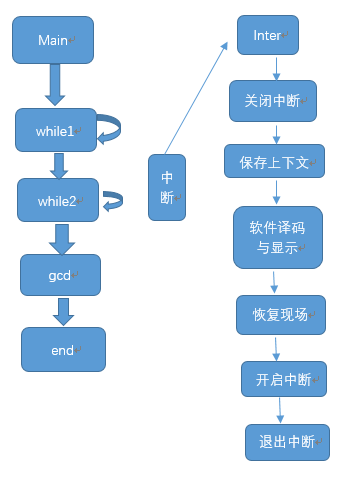
1. 分工

在小组中，于晨昕主要负责单周期处理器的代码以及流水线处理器中寄存器部分的代码；胡钰彬主要负责ALU模块以及流水线处理器中的其他部分；王振宇主要负责汇编代码、编译器以及外设模块的代码。

1. 实验设计
   1. 总体方案设计
   2. 汇编程序和编译器

3.2.1汇编程序

汇编程序的基本结构如下图所示：



汇编程序完成的功能为通过 UART 输入两个 8bits 的操作数并计算它们的最大公约数。采用软件译码方式通过七段数码管显示十六进制的操作数，通过 8 个 LEDs 显示最大公约数结果并通过 UART 输出。

具体来讲，首先，进入 Main，在while1与while2两个部分通过轮询的方式利用UART读取两个操作数，读取完毕后设置定时器周期并启动定时器。之后跳转到gcd处进行最大公约数的计算。计算结束后进入end部分，将结果通过UART返回，并通过二极管显示。

当定时器发送中断信号时，程序会进入Inter部分，开始执行中断处理代码。依次完成以下流程：

I.定时器中断禁止，同时中断状态清零，TCON的 1-2bit 清零，TCON&=0xfffffff9

II.保存现场

III.软件译码并显示

IV. 恢复现场

V.使能中断，TCON的 1bit 置 1，TCON |= 0x00000002

VI.退出中断服务程序，跳转到中断发生时保存 的断点地址处继续执行（$26）

3.2.2编译器

我们的编译器使用 C++语言，对 txt 格式的文 本进行处理，最终生成 txt 文件，文件中为汇编代 码的对应的机器码。

编译的过程是对汇编指令所在文件的两次扫描。第一次扫描，主要读取汇编代码中的所有标签，并将标签对应的地址计算出来，存入一个vector变量中。第二次扫描，则逐条处理汇编代码，将每一条汇编代码都转换为相应的机器码。

在逐条处理的过程中，考虑到十进制立即数、寄存器、OpCode和Func的转换，设计了dec2bin函数，实现十进制数转换为二进制比特串的功能，输入为需要转换的十进制数和输出二进制比特串的位数，输出为对应的二进制比特串。具体转换时，首先使用函数ins2Opcode识别OpCode对应的十进制数，之后根据识别到的指令种类不同，进行不同的处理。根据本实验的要求，在编译过程中，我们将指令分为了一般R型指令、内存读取型指令（lw和sw）、lui指令、一般I型指令、左移右移型指令、分支跳转型（其中beq和bne为一类、blez, bgtz和bltz为一类）、j和jal、jr、jalr、nop这几类，根据每一类中指令的特点，进行相应的转换。

* 1. 外设

外设模块块其实与数据存储是统一的， 根据地址范围对二者进行区分。0x00000000～0x3FFFFFFF（字节地址）为数据 RAM，可以提供数据存储功能；0x40000000～ 0x7FFFFFFF（字节地址）为外设地址空间，对其地址的读写对应到相应的外设资源。

外设中定时器模块已经由老师完成，因此这里主要介绍一下 UART 外设。UART外设模块根据春季学期设计的UART 进行改造。基本结构包括UART时钟产生模块（产生9600波特率串口对应的合适时钟频率）、接收器和发送器三个模块。与春季学期不同的是由于我们使用的CPU周期大于UART使用的时钟周期，因此接收器中的rx\_status需要适当地加长从而使得CPU能够检测到。

* 1. ALU
  2. 单周期
     1. 指令存储器
     2. 控制器
     3. 寄存器
     4. 数据存储器

3.6流水线

3.6.1冒险和转发

3.6.2ID

3.6.3IF

3.6.4EX

3.6.5MEM

3.6.6WB

1. 仿真情况
   1. 编译器&MIPS
   2. ALU
   3. 单周期
   4. 流水线
2. 综合情况
   1. 单周期
      1. 时序分析
      2. 面积分析
   2. 流水线
      1. 时序分析
      2. 面积分析
3. 总结
   1. 调试问题
      1. ALU
      2. 单周期
      3. 流水线
   2. 收获&致谢

参考文献

附录