下面由我来为大家展示我们小组的例外与中断处理部分。我将从我将首先介绍例外与中断处理的硬件基础CP0协处理器，然后从整体上介绍我们的例外与中断的处理思路，最后介绍结合具体代码，实现特权指令与仿真测试。

换页

协处理器实际上是由若干寄存器和其控制逻辑组合而成的寄存器堆，其中的寄存器反映着处理器的运行状态。异常发生时的检测和处理都由CP0中的寄存器来定义和控制。

我们实现的CP0协处理器中有6个寄存器，分别是：1、记录最近一次地址相关异常地址的BadVAddr寄存器；2、负责处理器周期计数的Count寄存器；3、负责定时中断控制的Compare寄存器；4、记录处理器状态的Status寄存器；5、记录上一次异常发生原因的Cause寄存器；6、记录上一次异常时pc的EPC寄存器。这6个寄存器具体字段的含义均可以通过A02系统指令规范和A06自己动手写CPU进行查询，这里不再赘述。

CP0协处理器中的寄存器都是程序员可见的，因此程序员可以对其进行读写操作，读写操作通过MTC0、MFC0这样的特权指令实现。

换页

在实现了CP0协处理器中的寄存器之后，需要对其读写过程进行优化，仿照RegReadProxy模块为CP0提供CP0ReadProxy模块，用来解决数据相关冲突问题。

在上述实现的基础上，便可以开始对例外和中断进行处理。

换页

异常是某些事件打断了程序的正常执行，在CPU中一共需要实现6类异常，分别是：1、地址相关异常，如加载数据地址未对齐，存储数据地址未对齐；2、系统调用，如执行了系统调用指令syscall；3、断点异常，如执行了断点指令break；4、溢出异常，如运算指令add，addi，sub运算溢出；5、保留指令异常，即试图执行CPU没有实现的指令；6、中断异常Interrupt

在操作系统课程中我们学习到，当异常发生后，系统会自动调用异常处理程序进行具体的异常处理，待处理完毕后，返回异常发生的位置继续执行。实验指导书为我们提供了异常处理程序的入口地址380H。

基于精确异常的要求，必须要求异常发生的顺序与指令顺序相同，但是对于流水线处理器而言异常会在不同阶段发生。针对这个问题，我们的解决方法是：选定一个特定的流水级统一处理异常，在其他阶段只检测异常并生成相应的异常信号，综合考虑6种异常之后，我们选定mem级作为统一处理异常的阶段。

换页

对于异常的判断，需要分流水级进行：

1、在IF阶段，需要对指令的地址是否对齐进行判断，若未对齐触发地址异常，生成相应的信号：ft\_adel\_flag。

2、在ID阶段，需要对是否是断点指令、系统调用指令、异常返回指令，保留指令异常、溢出异常进行判断，主要在ExpectionGen模块生成相应的信号：break\_flag、syscall\_flag、eret\_flag、ri\_flag、ov\_flag，并将这些信号融合为exc后传递到EX级。此外，还需要将currentpc传递到EX级。

3、在EX阶段，需要对溢出异常进行判断，生成相应信号：ov\_exc，将异常信号融合进exc后传至MEM级，并将currentpc传至MEM级。

4、在MEM阶段，结合CP0寄存器的内容，需要对中断，读写地址异常进行判断，生成相应信号：int\_occured、int\_enabled、adel\_flag、ades\_flag，再根据传过来的exc信号，对需要处理的异常进行判断。

由于在WB级不检测异常，所以在MEM级会将异常信号和相应的pc信号传至CP0，接下来就是对异常的处理。

换页

对异常进行处理时主要涉及CP0寄存器和piplinecontroler模块。

piplinecontroller模块接收来自CP0的epc信号，来自MEM级的exc信号，负责：1、提供例外处理程序的入口地址380H；2、程序返回时生成flush信号冲刷流水线；3、提供程序返回时重新开始执行的指令的pc

CP0寄存器需要接收来自MEM级的exc异常类型信号，来自EXMEM模块的currentpc和delayslot信号，主要负责：1、BadVAddr寄存器记录与地址异常的PC；2、status寄存器根据异常处理情况修改exl位；3、cause寄存器记录上次发生的例外；4、epc寄存器记录上次发生例外的pc；5、根据当前指令是否在延迟操中生成异常处理完毕后开始执行的pc

具体的修改细节通过查阅实验指导书和自己动手写CPU

换页

接下来，将详细介绍特权指令mtc0的实现过程和相应的仿真测试。

查阅实验指导书，指令的要求是将指定寄存器中的内容写进指定CP0寄存器。

实现思路如下：1、在译码阶段根据指令，在RegGen模块读出地址为rt寄存器的值，在CP0Gen模块生成对CP0的写信号和地址；

换页

2、在执行阶段确定要写入的CP0寄存器的值，将这些信息传递到访存阶段；3、在访存阶段将这些信息传递到写回阶段；4、在写回阶段根据这些信息修改CP0中地址为【rd，sel】的寄存器

换页

接下来，我们编写相应的测试程序，对mtc0指令进行仿真测试。这段测试程序是仿照A06实验指导书编写的，首先给$1寄存器赋值，然后将$1中的内容写入CP0中的status寄存器。

换页

根据仿真波形，可以看到CP0\_write\_data为$1中的内容0x10000401，在下一个时钟周期status寄存器的内容得到修改。