（一）

只需按上次作业修改pipe-full.hcl中对应部分即可：

将该两条指令设为合法。

需要寄存器、常数值中添加IADDL指令。

为IADDL, LEAVE指令添加A、B源寄存器及E、M目的寄存器位置。

为该两条指令设置ALU输入A、B。

IADDL需要修改condition code.

选择内存地址。

设置内存读写控制信号。

修改完成后，将Makefile 文件中的VERSION设为full。

make clean, make。

运行检测指令 make SIM=../pipe/psim TFLAGS=-il.

通过。

（二）

增加 RMEXCHANGE指令。

该指令将一个寄存器中的内容与一个内存地址中的内容互换。

由于其与rmmvol及mrmovl的相似性，比对其两条指令进行修改。

修改isa.h, isa.c 及 grammar.lex将其合法定义。

修改 pipe-full.hcl类似上题。修改流水线寄存器控制逻辑。

make clean 以及 make.

编写小程序测试该指令正确运行。代码及注释见附件。

../mics/yis rmtest.yo及../pipe/psim rmtest.ys

编写通讯程序。当signal=0时，A运行；signal=1时，B运行。

每次运行都将内存地址附近处的值增加1.

代码及功能注释见附件。

修改isa.c, isa.h重新定义init\_mem2使分配共享内存。

由于使用了linux核函数记得包含相关头文件。

修改pipe-psim.c 使得调用新的内存分配函数。

make clean, make.

在同一个终端下运行两个线程

./pipe/psim –v 1 –l 20000 sharetest.yo & ./pipe/psim –v 1 –l 20000 sharetest.yo

可以看到计数器正确交替上升。

致谢：

由于对linux环境不甚熟悉，在配置文件、修改bug过程中，得到了来自金迪、夷安、张起、卓亮的许多帮助，以此致谢。