

lw \$t0,0(base) J: add \$1,\$t0,\$t0 检测到应该为跳转后,废弃 指令J+1应该刚解完码 lw解码: 需要在Fetch和第二缓冲区加 uins=lw 一条数据通路进行通知,下 一次执行结果应是nop? rd1=base rd2=不重要 imm32=signext 从instruction获得 rt 数据相关和load延 addi解码: 迟在这个设计中应 uins=add 是一样处理。因为 uins=lw

回连Control的result在内部 会连一个寄存器,或者是 reg_rd

1 - src1使用result_reg值 2 - src2使用result_reg值 lw执行: 3 - src1和scr2都用result_reg reg_wd=ra reg_wa=t0

访存也变为了执行 rd1=t0xrd2=t0xconflict=3

设计哲学: 尽量

降低控制开销

wa=t0 addi执行: conflict=3所以使

result_reg=ra

用result_reg里面的 值做src1和src2 reg_wd=result

reg_wa=\$1