龙芯中科
LOONGSON TECHNOLOGY

单周期CPU设计实验

阶段二

CPU的一般设计方法

龙芯中科 LOONGSON TECHNOLOGY

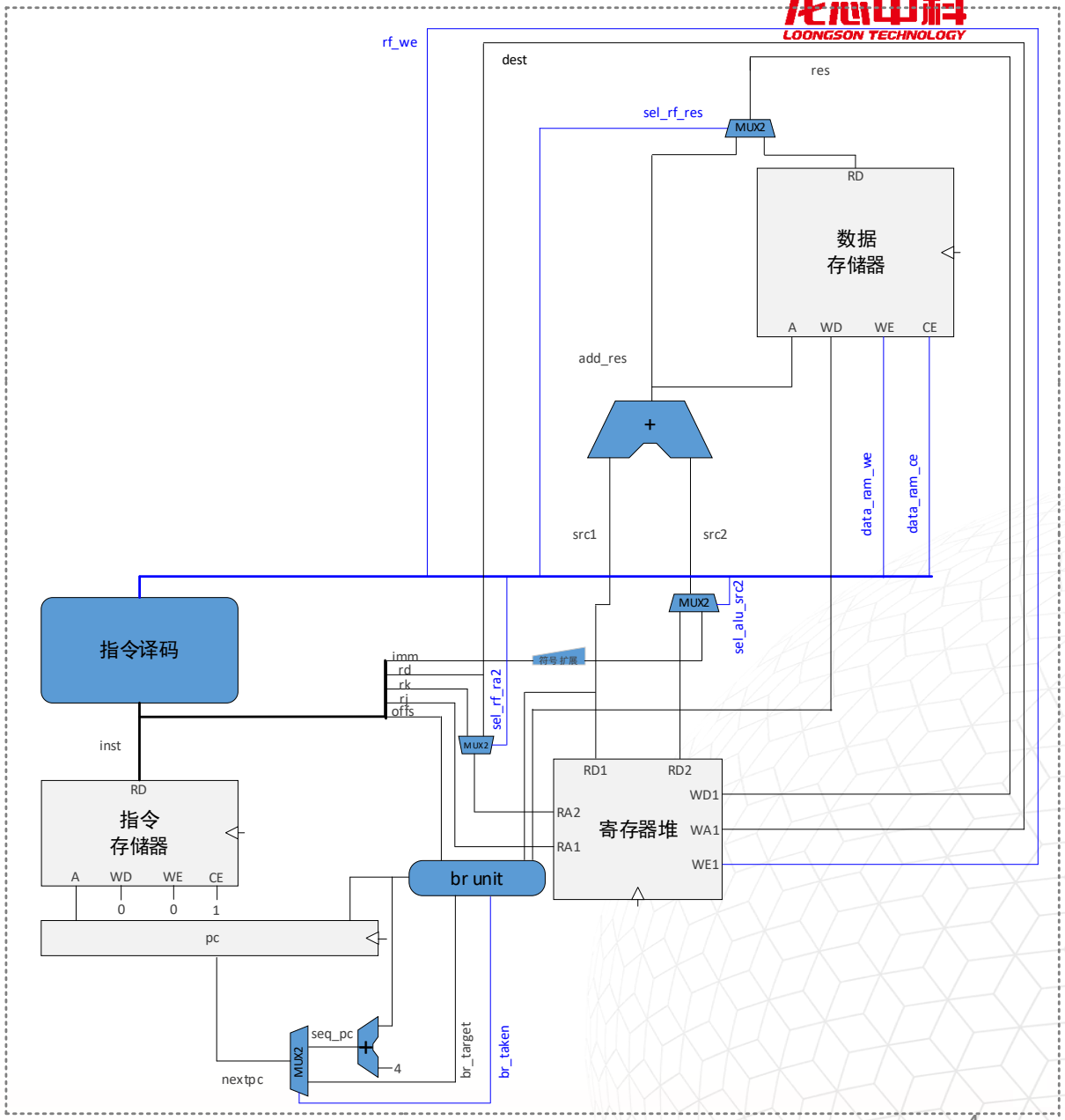
- CPU设计的特点：根据指令系统规范中的定义设计出 “数据通路 + 控制逻辑”
  - 对指令系统中定义的指令逐条进行功能分解，得到一系列操作和操作的对象，这些操作和操作的对象必然对应其各自的数据通路
  - 因为指令间存在一些相同或相近的操作和操作对象，所以我们可以只设计一套数据通路供多个指令公用
  - 对于确实存在差异无法共享数据通路的情况，只能各自设计一套，再用多路选择器从中选择出所需的结果

待实现指令定义

龙芯中科 LOONGSON TECHNOLOGY

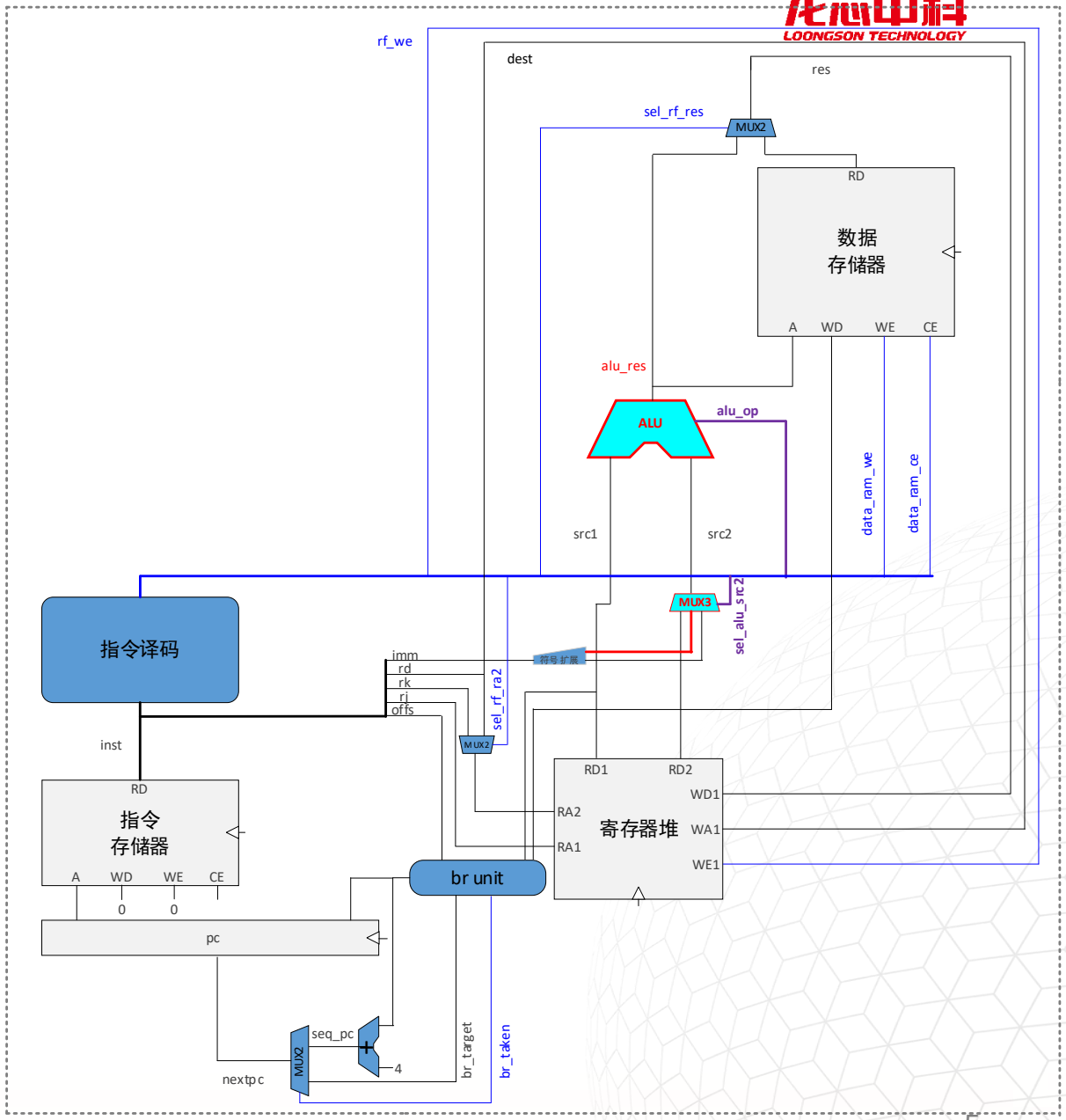
• **sub.w (subtract word)**	sub.w rd, rj, rk	GR[rd] = GR[rj] - GR[rk]
• **slt (set less than)**	slt rd, rj, rk	GR[rd] = GR[rj] $<_{signed}$ GR[rk]
• **sltu (set less than unsigned)**	sltu rd, rj, rk	GR[rd] = GR[rj] $<_{unsigned}$ GR[rk]
• **slli.w (shift left logic immediate word)**	slli.w rd, rj, ui5	GR[rd] = GR[rj] << ui5
• **srli.w (shift right logic immediate word)**	srli.w rd, rj, ui5	GR[rd] = GR[rj] $>>_{logic}$ ui5
• **srai.w (shift right arithmetic immediate word)**	srai.w rd, rj, ui5	GR[rd] = GR[rj] $>>_{arith}$ ui5
• **and (and)**	and rd, rj, rk	GR[rd] = GR[rj] & GR[rk]
• **or (or)**	or rd, rj, rk	GR[rd] = GR[rj] | GR[rk]
• **nor (not or)**	nor rd, rj, rk	GR[rd] = ~(GR[rj] | GR[rk])
• **xor (exclusive or)**	xor rd, rj, rk	GR[rd] = GR[rj] ^ GR[rk]
• **lu12i (load upper from bit12 immediate)**	lu12i rd, si20	GR[rd] = {si20, 12'b0}
• **beq (branch on equal)**	beq rj, rd, off16	if (GR[rj]==GR[rd]) PC = PC + sext32({off16, 2'b0})
• **b (branch)**	b offs26	PC = PC + sext32({off26, 2'b0})
• **bl (branch and link)**	bl offs26	GR[1] = PC+4; PC = PC + sext32({off26, 2'b0})
• **jirl (jump indirect register link)**	jirl rd, rj, offs16	GR[rd] = PC+4; PC = GR[rj]+sext32({offs16, 2'b0})

已有CPU结构设计


rf_we
dest
res
sel_rf_res
MUX2
RD
数据
存储器
A
WD
WE
CE
add_res
+
src1
src2
data_ram_we
data_ram_ce
MUX2
sel_alu_src2
指令译码
imm
rd
rk
rj
offs
符号扩展
sel_rf_ra2
MUX2
inst
RD
指令
存储器
A
WD
WE
CE
0
0
1
RD1
RD2
WD1
RA2
寄存器堆
WA1
RA1
WE1
br unit
pc
seq_pc
+
4
MUX2
nextpc
br_target
br_taken

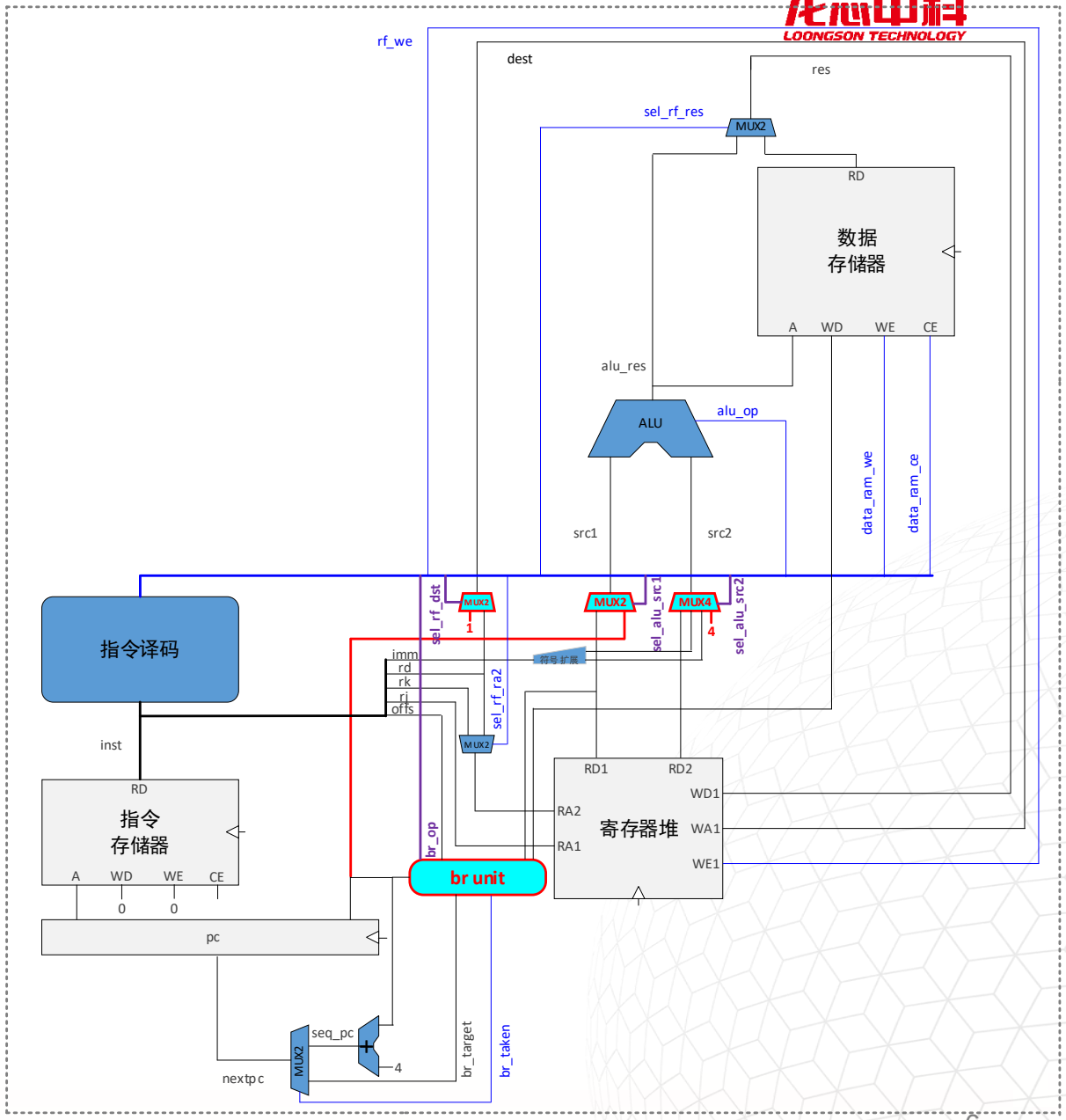
添加ALU类指令

- 将加法器升级为ALU部件，以支持新增ALU类指令的算术逻辑功能，同时需新增ALU部件的控制信号alu_op
- slli.w、srli.w、srai.w指令的立即数位宽小于addi.w的立即数位宽，可以高位补0后复用已有通路；lu12i.w指令的立即数位宽大于addi.w的，需增加新的数据通路



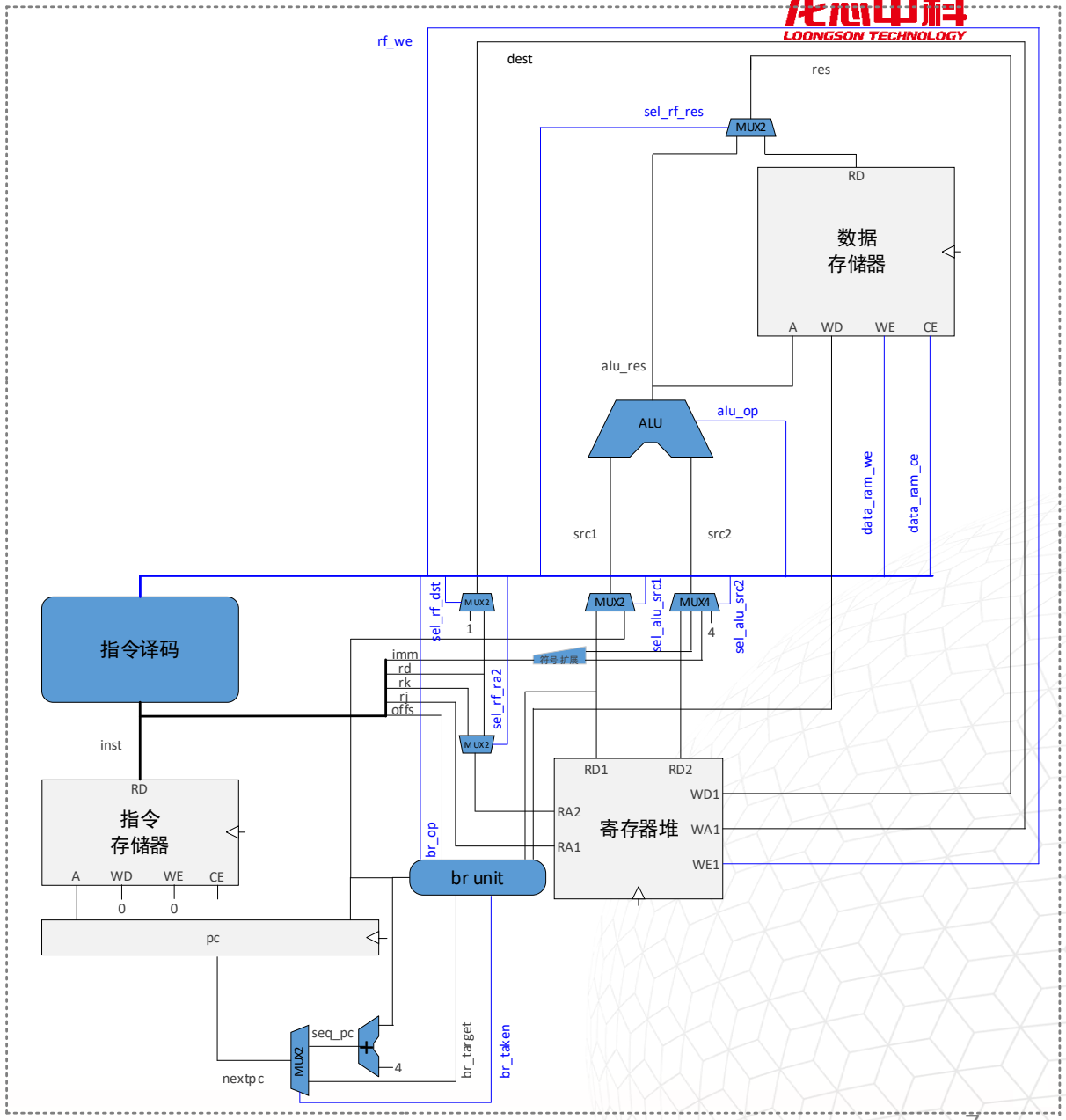
添加转移类指令

- beq与bne很相似，仅判断条件不同
- b指令无需判断一定跳转，不过其用于计算跳转目标的偏移值立即数位宽增至26位
- bl指令在完成b指令所有功能之外，还要将PC+4的值写入$r1寄存器
- jirl指令无需判断一定跳转，不过其跳转目标是rj寄存器加立即数偏移值，其将PC+4的值写入rd寄存器功能可以复用bl指令中的相关数据通路



调整控制信号

- 已有的控制信号
  - br_taken, sel_rf_ra2, sel_alu_src2, data_mem_ce, data_mem_we, sel_rf_res, rf_we
- 新增的控制信号
  - sel_rf_dst, sel_alu_src1, alu_op
- 注意：无论已有还是新增的控制信号都要考虑到所有的指令。



龙芯中科
LOONGSON TECHNOLOGY

为人民做我芯