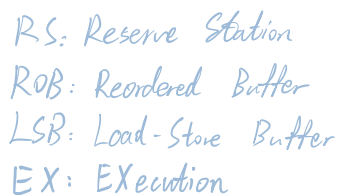


Abbreviation



LSB & EX output:
gol, vol ↑ ↓
 jt to ROB.
 RS, LSB

RS
LSB | exe: EXEcutable

LSB | ls: Load or Store

ROB | rdy: READY to be committed

LSB | rw: Read or Write

LSB
↓
DC | len: LENgth

DC | add: ADDress

DC | dat: DATa

ROB
↓
LSB | cs: Commit Store

IS. $rs \& rd \rightarrow REG$
 $REG[rd].q = qn$
 $REG[rs].vs \& qs$
 $ROB[qn].pc \& rd \leftarrow IS.pc \& rd$
 $\rightarrow ROB.qs = REG.qs$
 $\rightarrow ROB.vs = (ROB[qs].en) ? (ROB[qs].v) : (REG.vs)$

LSB 内依次执行

ROB 内依次 Commit

ROB 内 Branch/Jump 指令 **精确中断**:

1. Store 指令完成后对应 entry 在 ROB 才能 Commit
2. 预测失败则清空

IF, IS, ROB, RS, EX, **LSB**

以及 REG 中的 Q

保留前缀
Committed STORE