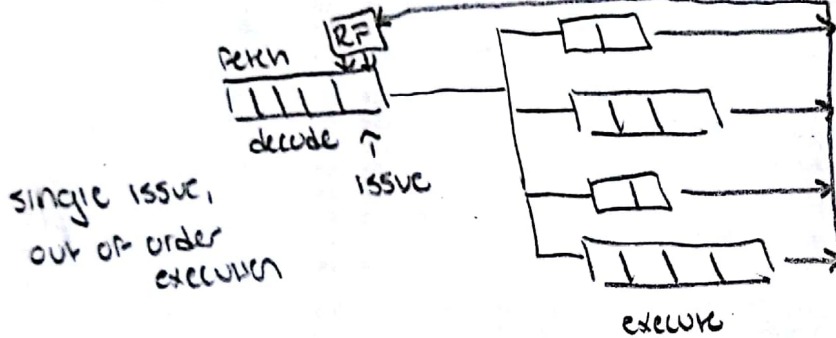


3. Tell me what you know about **out-of-order execution** ... explain it in terms of the mechanisms we studied: **Thornton's Scoreboard** in the CDC 6000, **Tomasulo's Algorithm** in the IBM 360/91, and the **register-renaming** mechanism of the MIPS R10000.

In the late 60's, race to 1 CPI



limitations to this design:

1. there is a stall if the operands are not ready in RF
2. stall if there is potential conflict

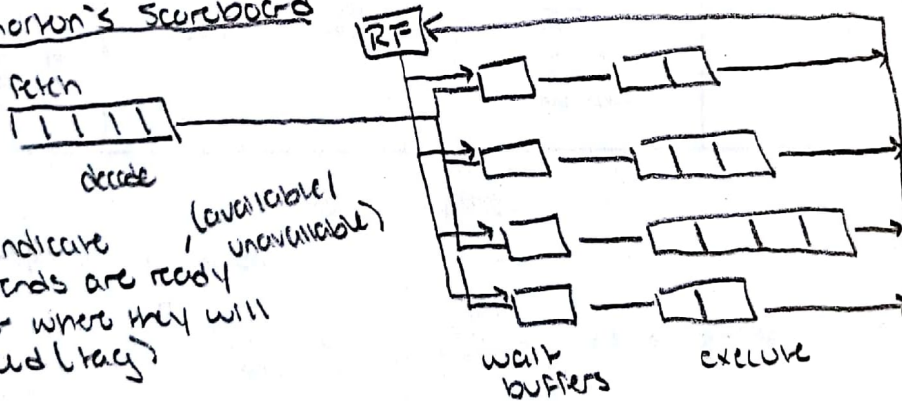
also data hazards thru RF
 RAW (read after write)
 WAR (write after read)
 WAW (write after write)

Excellent

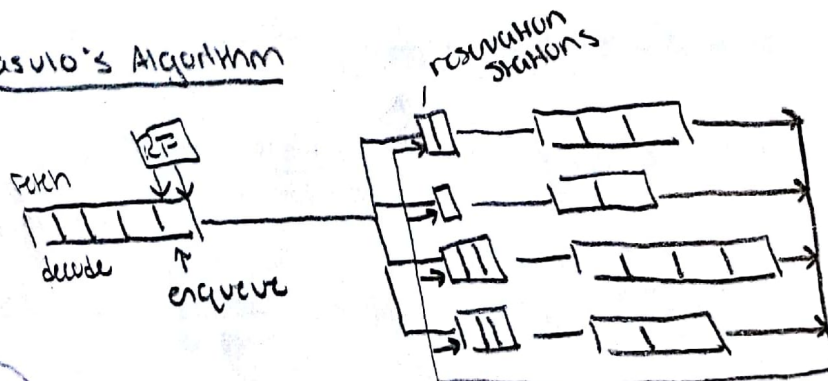
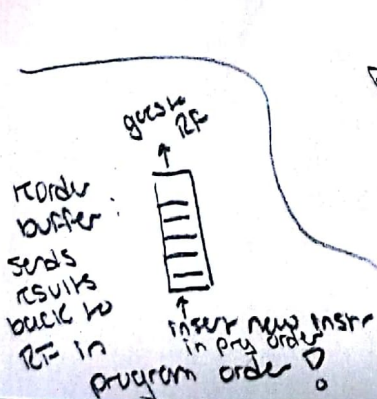
To address limitations, Thornton's scoreboard & Tomasulo's alg were developed.

Thornton's Scoreboard

wait buffers indicate whether operands are ready and if not where they will be produced (tag)



Tomasulo's Algorithm



Unlike Thornton's, Tomasulo's address the WAW register renaming (when two regs w/ same name don't actually have dependency)

Today tomasulo's + reorder buffer is common