
EECS 452 – Lecture 5

Register Renaming

Gokhan Memik

EECS Dept., Northwestern University

Register Renaming



- Avoid stalling for WAR and WAW depends.
- Register version
 - Every write create a new version
 - Uses read last version
 - Need to keep a version until all uses have read it
- Register Renaming
 - Architectural vs. Physical Registers
 - More phys. than arch.
 - Maintain a map of arch. To phys. regs
 - Use in-order decoding to properly identify dependences
 - Instructions wait only for input op. availability

Register Renaming-Example



A: DIVF F3, F1, F0 r1,-,-
 B: SUBF F2, F1, F0 r2,-,-
 C: MULF F0, F2, F4 r3,r2,-
 D: SUBF F6, F2, F3 r4,r2,r1
 E: ADDF F2, F5, F4 r5,-,-
 F: ADDF F0, F0, F2 r6,r3,r5

	F0	F1	F2	F3	F4	F5	F6	F7	...	F30
A				R1						
B			R2	R1						
C	R3		R2	R1						
D	R3		R2	R1			R4			
E	R3		R5	R1			R4			
F	R6		R5	R1			R4			

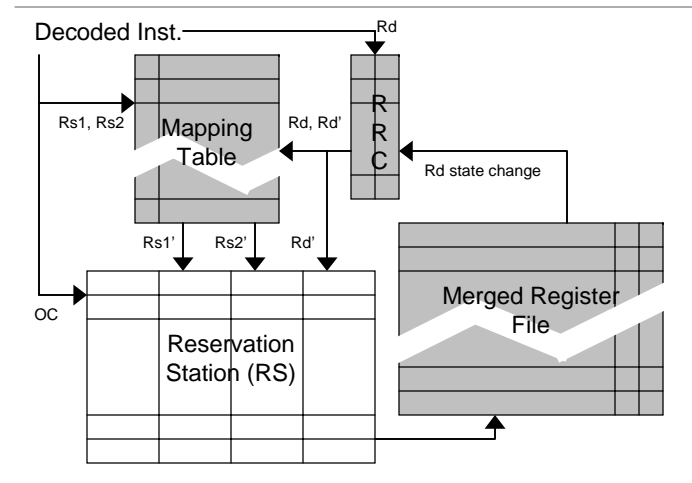
Need more phy regs than architectural

Dynamic Scheduling Example



- Values and names flow together
- Writeback specifies both value and name
- A waiting instruction inspects all results
- It is allowed to execute when all inputs are available
- When to release a phy reg?

Overview of renaming



- Mapping table
 - Vector of logical \rightarrow physical
- Register file
 - Merged vs. “architectural and physical”