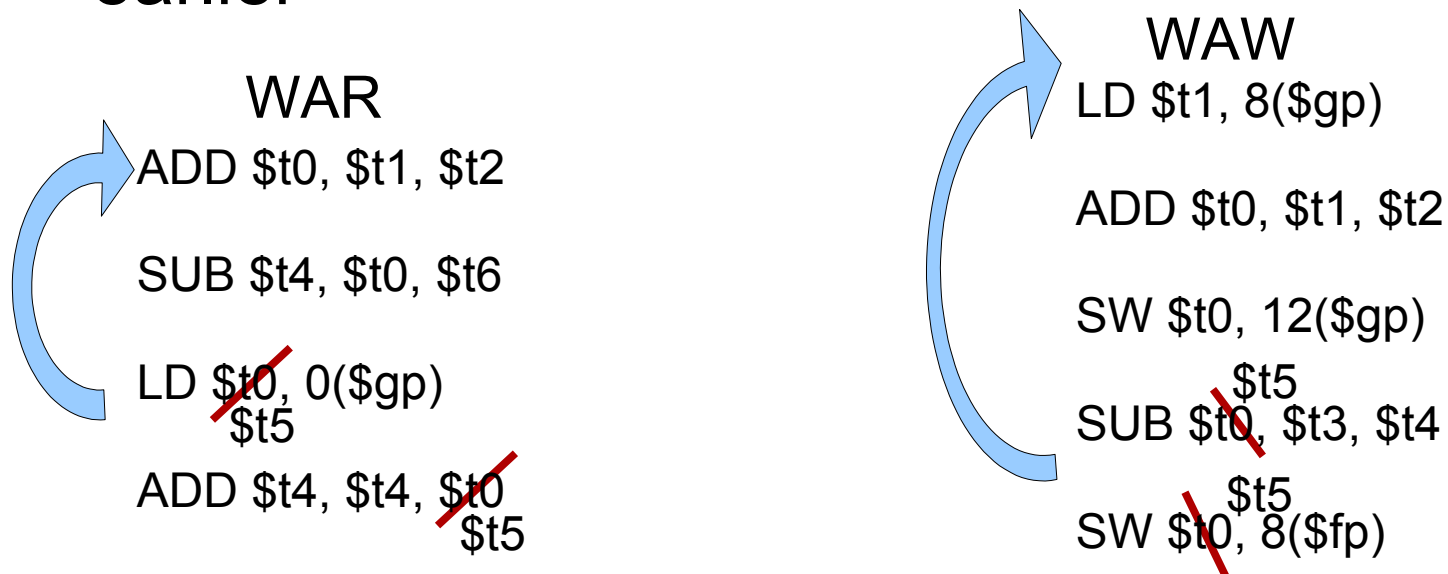


Maximum ILP

- Assume infinite hardware
- Only limited by dependencies in the code
 - RAW – Real Dependence
 - WAR – False(anti-) Dependence
 - WAW – False(anti-) Dependence
- Scoreboarding is stuck with anti-dependencies
- Tomasulo's can work around anti-dependencies

High Level Tomasulo's

- Eliminates name-dependencies through renaming
 - WAR, WAW – If both of these “Write After”'s were writing to a different place, there would be no dependency, and we could execute instructions earlier



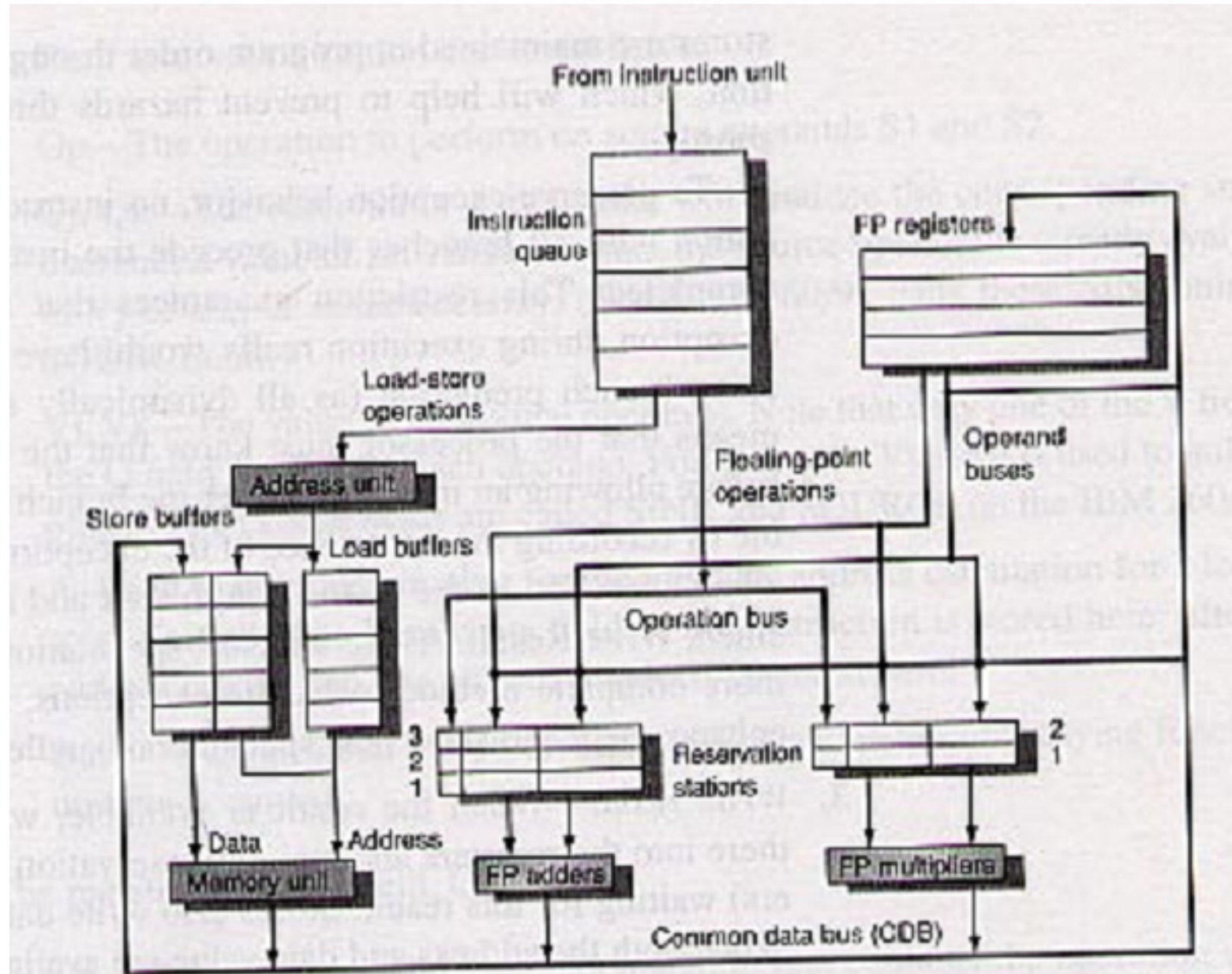
High Level Tomasulo's

- Each written result gets its own name
- Each instruction “in-flight” knows its operands by the unique result name, not the register/memory location from the instruction
- When a result is produced, the result name and value are broadcast on the bus
- Anything waiting for that result picks up the value from the bus and can proceed

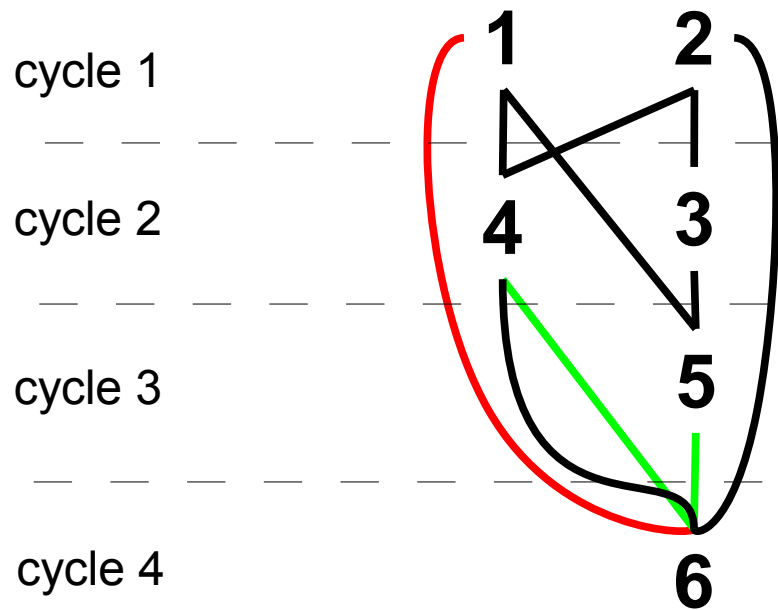
Tomasulo's Alg. Implementation

- Instructions are issued to entries in functional unit buffers called *reservation stations*
- The name of a reservation station that will produce a result becomes the name of that result
- When you can re-use the reservation station, you've already broadcast the result, so you can reuse the name as well

Tomasulo's Alg. Implementation



Scoreboarding w/infinite hardware



Tomasulo's w/infinite hardware

