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

```
WAR
ADD $t0, $t1, $t2
SUB $t4, $t0, $t6
LD $t0, 0($gp)
ADD $t4, $t4, $t0
```

```
WAW
LD $t1, 8($gp)
ADD $t0, $t1, $t2
SW $t0, 12($gp)
SUB $t0, $t3, $t4
SW $t0, 8($fp)
```
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
# Scoreboard Example

**Instruction status**

<table>
<thead>
<tr>
<th>Instruction</th>
<th>k</th>
<th>Issue</th>
<th>Op</th>
<th>comple</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>LD</td>
<td>F6</td>
<td>34+</td>
<td>R2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LD</td>
<td>F2</td>
<td>45+</td>
<td>R3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MULTIF0</td>
<td>F2</td>
<td>F4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUBD</td>
<td>F8</td>
<td>F6</td>
<td>F2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DIVD</td>
<td>F10</td>
<td>F0</td>
<td>F6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDD</td>
<td>F6</td>
<td>F8</td>
<td>F2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Functional unit status**

<table>
<thead>
<tr>
<th>Time Name</th>
<th>Busy</th>
<th>Op</th>
<th>Fi</th>
<th>Fj</th>
<th>Fk</th>
<th>Qj</th>
<th>Qk</th>
<th>Rj</th>
<th>Rk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integer</td>
<td>No</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mult1</td>
<td>No</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mult2</td>
<td>No</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Add</td>
<td>No</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Divide</td>
<td>No</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Register result status**

<table>
<thead>
<tr>
<th>Clock</th>
<th>F0</th>
<th>F2</th>
<th>F4</th>
<th>F6</th>
<th>F8</th>
<th>F10</th>
<th>F12</th>
<th>...</th>
<th>F30</th>
</tr>
</thead>
</table>

**FU**

| FU |    |    |    |    |    |     |     |     |     |
Scoreboarding w/infinite hardware

Tomasulo's w/infinite hardware

cycle 1

cycle 2

cycle 3

cycle 4