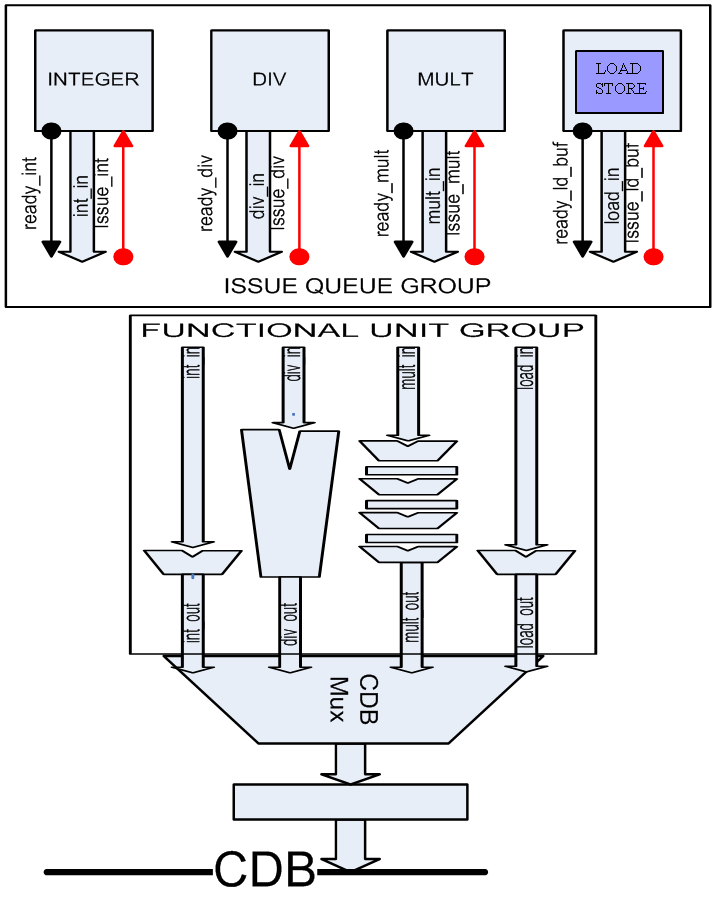
**Issue Unit**

**Introduction:**

Four Issue Queues send instructions to the Functional Units (FUs) as shown in Fig. 1.

The issue queue group represents:

* Integer Queue – (INTEGER)
* Divider Queue – (DIV)
* Multiplier Queue – (MULT)
* Load store Queue - (LOAD STORE) – it is followed by a load buffer where the lw/sw instructions wait.



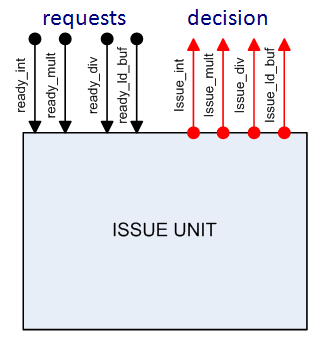
**Fig 1**

**Subtle features of Issue Queue Group:**

* Integer Issue Queue, Div Issue Queue and Mult Issue Queue are almost same, Div Issue Queue and Mult Queue being exactly same.
* Lw/Sw Queue is little different because of the memory disambiguation rules, resulting in the need of Address Buffer and counters.

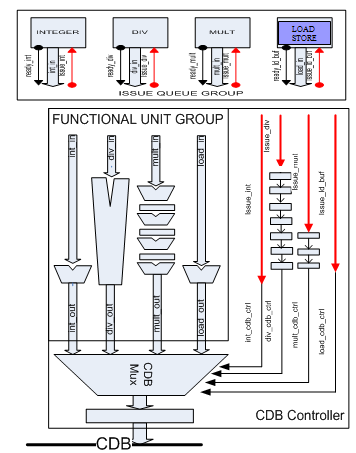
**Functions and operations of Issue Unit (Fig 2):**

* Receive requests from one of the Issue Queues and make Issue Decision.
* Acknowledge the Issue request from the corresponding Issue Queue.
* Control the CDB mux so as to decide which instruction will be coming on the CDB.

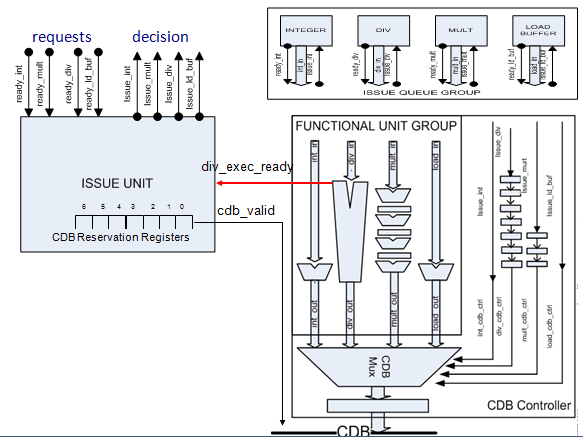


**Fig 2**

* Mux control signals cannot be given directly to CDB, they should travel through stage registers along with the functional unit data, remember the functional units are of different latencies ( div – 7 clocks, mult – 3 clocks). (Fig 3)
* Need of Bus reservation table to avoid collision on Bus when multiple instructions on various queues become ready to declare their result on the Bus. (Fig 4)

****

**Fig 3**

****

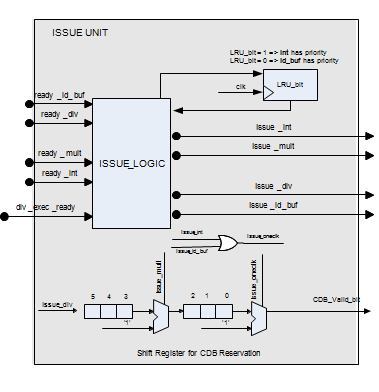
**Fig 4**

**NOTE:** divider is a multi cycle unit (non-pipelined) also we are using a singnal “div\_exec\_ready” to prevent consecutive execution of **div** on subsequent clocks. A ‘1’ on “div\_exec\_ready” indicates that the **div** has finished its execution, so issue the next div.

**Inside the Issue Unit – Implementation specific details (Fig 5):**

* **Issue Policy**
* long-latency instruction first – “*situation does not arise in our design”*
* LRU (Least Recently Used) for instructions of same latency
* **Latencies**
* **div:** 6 clocks + 1 clock
* **mult:** 3 clocks (pipelined) + 1clock
* **int:** 1 clock
* **ld\_buf:** 1 clock
* **LRU\_bit Usage and updating**
* 1: **int** has priority, 0: **ld\_buf** has priority
* **if both requested** - issue according to the LRU\_bit
* **if one of them requested** - issue it

**Up to 3 instructions enter the CDB Reservation Register**

****

**Fig 5**