Fairness Budget Register Specification

7/9/96

Dave LaFollette
dlafolle@mipos2.sc.intel.com

9.x FAIRNESS_BUDGET register (cable environment)

Not implemented, backplane environment.

Optional, cable environment. This register shall be implemented on nodes capable of sending priority asynchronous request packets.

The FAIRNESS_BUDGET register permits the bus manager to configure a node's fair arbitration behavior in order to optimize Serial Bus utilization. This register provides a mechanism for nodes to send more than one asynchronous request* packet per fairness interval. The definition is given by figure 9-x below.

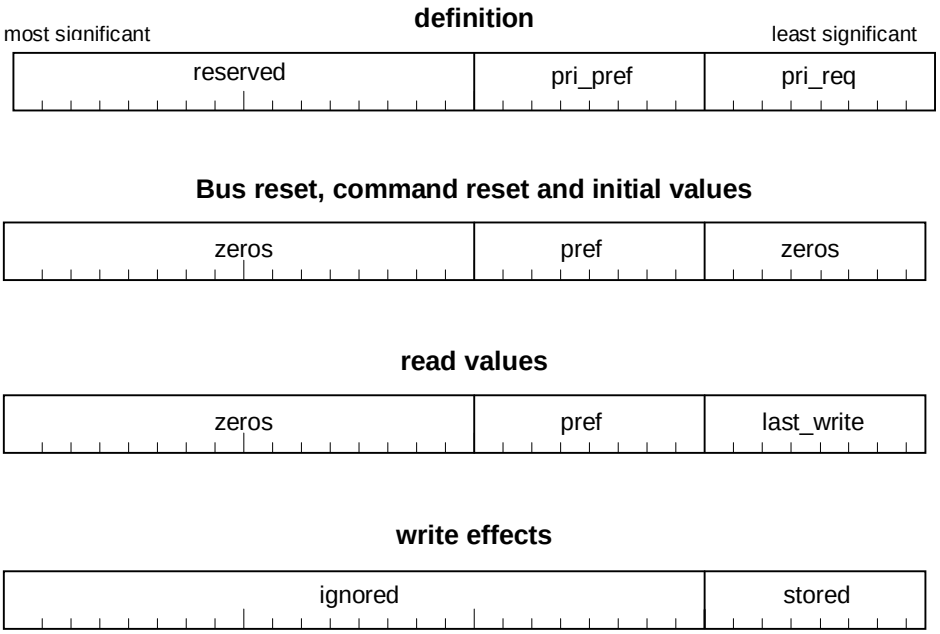

Figure 9-x – FAIRNESS_BUDGET format

The ***pri_req*** field shall specify the maximum number of priority arbitration requests for asynchronous request packets that the link is permitted to make of the PHY during a fairness interval. A Serial Bus fairness interval exists between the occurrence of an arbitration reset gap and the first subsequent arbitration reset gap. The *pri_req* default value of zero is equivalent to the fair arbitration behavior specified by IEEE Std 1394-1995.

The ***pri_pref*** field shall specify the nodes preferred number of priority arbitration requests for asynchronous request packets, *pref*. The bus manager should read this value before writing to the *pri_req* field. The bus manager shall not write a value greater than the nodes *pref* value to the *pri_req* field.

Each time a link receives PHY status of ARB_RESET_GAP, it shall reset an internal variable, *priority_request_count*, to the value of *pri_req* from the FAIRNESS_BUDGET register. When the link wishes the PHY to arbitrate for the bus in order to send a packet with one of the transaction codes listed in table 9-x, it may make a priority arbitration request if the value of *priority_request_count* is nonzero. Each time the link makes a priority arbitration request for one of these transaction codes it shall decrement *priority_request_count*.

Table 9-x – Transaction codes for priority arbitration counting against *priority_request_count*

Code	Name
0	Write request for data quadlet
1	Write request for data block
4	Read request for data quadlet
5	Read request for data block
9	Lock request
A	Asynchronous stream

* – This section does not alter the use of priority requests for cycle start or asynchronous response packets and they shall not count against the *priority_request_count* maintained by the link.