

ARBITER DIAGNOSTIC APPARATUS AND METHOD

BACKGROUND

The present invention generally relates to the arbitration of shared system resources, such as a bus or memory, and particularly relates to arbitration diagnostics.

Contemporary electronic devices incorporate increasingly sophisticated processing systems to handle various operational functions. For example, mobile communication handsets, handheld gaming systems, portable digital assistants, etc., provide an increasing wealth of multimedia functions supported by graphics, audio, communication, and display processing subsystems. The different subsystems generally include dedicated hardware and software processing elements, but they commonly share selected system resources, such as memory, buses, etc.

In such systems, "arbiters" coordinate shared resource accesses by competing entities, based on granting, denying, and deferring access requests made by individual ones of the competing entities. For example, a bus arbiter manages access to a shared system bus by two or more processing subsystems, each needing independent access to the bus. In this role, the arbiter grants bus access temporarily to one subsystem, while blocking bus accesses by the other subsystems.

Resource arbitration thus provides a mechanism for coordinated sharing of resources, but it also complicates system diagnostics. For example, in a distributed processing system using resource arbitration, arbitration processing may block a given subsystem from shared resource access for longer than can be tolerated, leading to an overall system fault. Diagnosing arbitration-induced faults presents significant challenges because of the complex processing states involved with the various subsystems, the "inter-relatedness" of these subsystem states, and the general unavailability of comprehensive arbitration state information at the point and time of failure.

SUMMARY

In one embodiment, a method of resource arbitration diagnostic processing comprises detecting arbitration events for two or more entities having arbitrated access to a shared resource, and maintaining a chronological memory trace of the arbitration events. For example, an arbitration diagnostic circuit can be configured to interface with an arbitration controller, to detect resource request and grant events managed by the controller. Optionally, event tracking includes resource release events, denoting resource release times associated with the resource requesting and granting operations.

In one or more embodiments, the chronological memory trace of the arbitration events comprises a running list of time-stamped arbitration events. Time-stamp information may be locally generated by an associated arbitration circuit, by the arbitration diagnostic circuit, or may be generated elsewhere in the system and provided as a reference signal to the arbitration diagnostic circuit. In any case, time-stamping provides one basis for determining resource grant delays, i.e., the delays between individual resource requests and the corresponding resource grants. These times may be important, for example, in systems where excess resource grant delays can cause the requesting entity to fail, stall, or otherwise operate erroneously.

Thus, the arbitration diagnostic method further comprises, in one or more embodiments, detecting excessive resource

grant delays. Such detection can be based on comparing resource grant delays for given resource request events against corresponding delay limits. The delay limits, which can be stored in one or more memories accessible to an arbitration diagnostic circuit, may be defined individually for the different entities for which arbitration event tracking is active.

Effectively, then, the arbitration diagnostic circuit can be configured to calculate the resource grant delays by tracking elapsed times after detecting resource request events, and comparing the elapsed times to maximum grant delay limits defined for the two or more entities. Elapsed time tracking uses, for example, the time stamp information captured for resource request events, and one or more hardware or software timing functions, which may be driven by a system clock, for example. In one or more embodiments, the arbitration diagnostic method comprises performing one or more actions responsive to detecting an excessive resource grant delay. Such actions include one or more of storing resource grant delay violation information in the running list, freezing the running list, asserting a system halt signal, asserting a delay violation alert signal, capturing arbitration state information, and capturing entity state information for one or more of the two or more entities having arbitrated access to the shared resource.

Thus, in at least one embodiment, an arbitration diagnostic method comprises tracking delay times between resource requests and corresponding resource grants for respective ones of two or more entities having arbitrated access to a shared resource, and detecting resource grant delay violations by comparing the delay times to one or more defined delay limits. The method further comprises asserting a resource grant delay violation signal responsive to detecting a resource grant delay violation, and may include capturing arbitration state information responsive to detecting a resource grant delay violation. Of course, such processing may include maintaining a chronological memory trace of arbitration events for the two or more entities, including resource request events and corresponding resource grant events.

Of course, the present invention is not limited to the above features and advantages. Indeed, those skilled in the art will recognize additional features and advantages upon reading the following detailed description, and upon viewing the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of one embodiment of an arbitration diagnostic circuit, illustrated in the context of one or more entities having arbitrated access to a shared resource.

FIG. 2 is a block diagram of one embodiment of an arbitration diagnostic circuit.

FIG. 3 is a logic flow diagram of one embodiment of arbitration diagnostic processing.

FIG. 4 is a table illustrating one embodiment of a chronological memory trace containing arbitration diagnostic information.

FIG. 5 is a logic flow diagram of another embodiment of arbitration diagnostic processing.

FIG. 6 is a block diagram of another embodiment of an arbitration diagnostic circuit.

FIG. 7 is a table illustrating another embodiment of a chronological memory trace containing arbitration diagnostic information.

FIG. 8 is a block diagram of circuit details for one embodiment of the arbitration diagnostic circuit of FIG. 6.