to execute and then update the response and the next expected access. It would simplify the interface between the microsequencer and the BIU to not have an automatic read facility, but having it both speeds up the interface and reduces the amount of microcode since more than one coprocessor access can be handled by one microinstruction.

Queueing responses. Again, both for reducing the amount of microcode required and for speeding up the reads the MC68020 makes to the response register, a facility exists that allows a microinstruction to queue more than one response to be returned when the response bus register is read.

Recall that there is an interface register between the microsequencer and the BIU that contains an encoded version of the response to be returned (see Figure 11 again). Queueing is made possible by a supercheck bit which, when set by the microsequencer, indicates to the BIU that it should temporarily ignore the interface register and instead return a hardwired supervisor check response. After this hardwired response has been read by the MC68020, the BIU clears the supercheck bit, and the response encoded in the interface register is returned when the next MC68020 response read occurs. (The MC68020 always reads the response register again after a supervisor check response passes.) A performance gain results since both response reads can occur without requiring the microsequencer to be started up, run, and shut down.

W
We have described the MC68851's architecture, operation, and implementation, emphasizing the techniques we used to increase performance. These techniques included pipelining to minimize descriptor fetching overhead, using autonomous BIU state machines, microsequencer biasing, microsequencer start-up in parallel with bus arbitration, and implementation of a fully associative translation cache. Other features, such as support for external data/instruction caches, also contributed to enhanced system performance. Through these enhancements and features, we achieved the goal of our PMMU design—efficient support of the MC68020 in a demand-paged virtual memory environment.

References

5. MC68851 Paged Memory Management Unit User's Manual (manuscript). (To be available from Motorola Literature Distribution Dept., Phoenix, AZ.)

Brad Cohen was system project leader for the MC68851 PMMU and was a member of the design team for the MC68000, MC68451, and MC68010. His primary technical responsibility for the PMMU was the chip's bus interface unit. He received a BSEE from the Georgia Institute of Technology.

Ralph C. McGarity's primary responsibilities while at Motorola have been the design and implementation of the MC68851 PMMU, particularly its microcode, microsequencer, and execution unit. His current interests include high-performance VLSI implementations of computer architectures and CAD. A member of IEEE, ACM, Phi Beta Kappa, and Tau Beta Pi, he received a BSEE from Rice University in 1980 and an MSEE from Carnegie-Mellon University in 1982.

Questions about this article can be directed to Cohen or McGarity at the High-End Systems Design Group, Motorola Microprocessor Products Division, Mail Stop OE-37, 6501 William Cannon Drive West, Austin, TX 78735-8598.

Reader Interest Survey

Indicate your interest in this article by circling the appropriate number on the Reader Interest Card.

High 162 Medium 163 Low 164