

MEMORY USABLE IN CACHE MODE OR SCRATCH PAD MODE TO REDUCE THE FREQUENCY OF MEMORY ACCESSES

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority to European Patent Application No. 04291918.3, filed on Jul. 27, 2004 and incorporated herein by reference. This application also contains subject matter that may be related to U.S. patent applications Ser. No. 10/818,584 entitled "Management of Stack-Based Memory Usage in a Processor, Ser. No. 10/632,067 entitled "Memory Management of Local Variables," U.S. Ser. No. 10/632,076 entitled "Memory Management of Local Variables Upon a Change of Context," and Ser. No. 10/632,228 entitled "System and Method to Automatically Stack and Unstack Java Local Variables." This applications also contains subject matter that may be related to concurrently filed applications entitled "Context Save and Restore with a Stack-Based Memory Structure"[Attorney Docket TI-38570 (1962-21100)] And "Cache Memory Usable As Scratch Pad Storage"[Attorney Docket TI-38571 (1962-21200)].

BACKGROUND OF THE INVENTION

[0002] 1. Technical Field of the Invention

[0003] The present disclosure relates generally to processors and more particularly to the use of cache memory as scratch pad storage.

[0004] 2. Background Information

[0005] Many types of electronic devices are battery operated and thus preferably consume as little power as possible. An example is a cellular telephone. Further, it may be desirable to implement various types of multimedia functionality in an electronic device such as a cell phone. Examples of multimedia functionality may include, without limitation, games, audio decoders, digital cameras, etc. It is thus desirable to implement such functionality in an electronic device in a way that, all else being equal, is fast, consumes as little power as possible and requires as little memory as possible. Improvements in this area are desirable.

BRIEF SUMMARY

[0006] In at least some embodiments, a processor is adapted to couple to external memory. The processor comprises a controller and data storage (e.g., cache memory). The data storage is configurable to operate in either a cache policy mode in which a miss results in an access of the external memory or in a scratch pad policy mode in which a miss does not result in an access of the external memory. The data storage comprises a first portion and a second portion, and only one of the portions is active at a time. The non-active portion is unusable to store or retrieve data (e.g., Java local variables). When the active portion does not have sufficient capacity for additional data to be stored therein, the other portion becomes the active portion.

[0007] In another embodiment, a system (e.g., a cellular telephone) comprises a communications transceiver, a first memory, a controller communicatively coupled to the communications transceiver and the memory, and a second

memory operated by the controller. The second memory is configurable to operate in either a cache policy mode in which a miss results in an access of the first memory or in a scratch pad policy mode in which a miss does not result in an access of the first memory. The second memory comprises a first portion and a second portion. Only one of the portions is active at a time, the non-active portion being unusable. When the active portion does not have sufficient capacity for additional data to be stored therein, the other portion becomes the active portion.

[0008] In yet another embodiment, a method comprises using only a first portion of a cache memory data array to store local variables until the first portion has insufficient capacity for storing additional local variables. The cache memory data array comprising the first portion and a second portion. Once the first portion has insufficient capacity for storing additional local variables, the method comprises using only the second portion of the cache memory data to store the additional local variables and not using the first portion. When the second portion has insufficient capacity for storing additional local variables, the method comprises copying the local variables from only the first portion to external memory.

NOTATION AND NOMENCLATURE

[0009] Certain terms are used throughout the following description and claims to refer to particular system components. As one skilled in the art will appreciate, different companies may refer to a component by different names. This document does not intend to distinguish between components that differ in name but not function. In the following discussion and in the claims, the terms "including" and "comprising" are used in an open-ended fashion, and thus should be interpreted to mean "including, but not limited to . . .". Also, the term "couple" or "couples" is intended to mean either an indirect or direct connection. Thus, if a first device couples to a second device, that connection may be through a direct connection, or through an indirect connection via other devices and connections. The terms "first portion" and "second portion" are intended to broadly refer to either portion of the multi-portion RAM-set explained below.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] For a more detailed description of the preferred embodiments of the present invention, reference will now be made to the accompanying drawings, wherein:

[0011] **FIG. 1** shows a diagram of a system in accordance with preferred embodiments of the invention and including a Java Stack Machine ("JSM") and a Main Processor Unit ("MPU");

[0012] **FIG. 2** depicts an exemplary embodiment of the system described herein in the form of a communication device (e.g., cellular telephone);

[0013] **FIG. 3** shows a block diagram of the JSM of **FIG. 1** in accordance with a preferred embodiment of the invention;

[0014] **FIG. 4** shows various registers used in the JSM of **FIGS. 1 and 3**;

[0015] **FIG. 5** illustrates the storage of local variables and pointers in accordance with the preferred embodiments;