

DATA STORAGE MANAGEMENT APPARATUS AND PROCESSING CORE

TECHNICAL FIELD

The disclosure relates to the technical field of processing cores, in particular,
5 relates to a data storage management apparatus and a processing core.

BACKGROUND

With the development of science and technology, human society is rapidly entering
an era of intelligence. The important feature of the era of intelligence is that people may
acquire more and more types of data, the amount of data that people may acquire is
10 becoming larger and larger, and the demand for the data processing speed is becoming
higher and higher.

Chips are the cornerstone of data processing, which essentially determine the
ability to process data. From the perspective of application fields, the chips are mainly
developed in two routes. One is general-purpose chips, for example, CPU and the like,
15 which can provide great flexibility but have low effective computing power when
processing algorithms in a specific field; the other is special-purpose chips, for example,
TPU and the like, which have high effective computing power in certain specific fields
but are poor in processing capacity or even have no processing capacity in flexible and
universal fields.

20 Due to various data types and numerous data volumes in the era of intelligence,
the chips are required to not only process changing algorithms in different fields with
extremely high flexibility, but also rapidly process a great number of dramatically
increasing data volumes with extremely strong processing capacities.

SUMMARY

25 The present disclosure provides a data storage management apparatus and a
processing core, which can address the decrease in computational efficiency caused by
cache access failure, and improve the controllability of program efficiency.

A first aspect of the present disclosure provides a data storage management
apparatus, including: at least two random access memories (RAMs); a control unit,
30 configured to receive an instruction, and generate and send a control signal according

to the instruction; and a direct memory access controller (DMAC), configured to access data in the RAMs according to the control signal.

The data storage management apparatus provided by embodiments of the present disclosure receives and responds to an instruction sent from an external processing unit, and reads data from an external storage unit, so that the external processing unit can directly read the data required for executing a program from the data storage management apparatus when executing the program. The external processing unit does not need to fetch, via the cache, the data from the external storage unit, thereby avoiding the decrease in computational efficiency caused by cache access failure, and improving the controllability of the program efficiency.

Optionally, the DMAC configured to access data in the RAMs according to the control signal, comprises: the DMAC is configured to read data from an external storage unit and store the data in the RAM indicated by the control signal, according to the control signal; or the DMAC is configured to read data from the RAM indicated by the control signal and store the data in the external storage unit, according to the control signal.

Optionally, the number of the RAM indicated by the control signal is one or more.

Optionally, all the RAMs and the external storage unit are uniformly addressed; or, all the RAMs are uniformly addressed.

Optionally, an address range accessible by the DMAC is address segments of the RAMs and address segment of the external storage unit.

According to another aspect of the present disclosure, there is provided a processing core, including a processing unit, a storage unit and the data storage management apparatus provided in the first aspect; wherein the processing unit is configured to send an instruction, and the instruction is for instructing the data storage management apparatus to access data in the storage unit; and the processing unit is further configured to read data required for executing a program from any of the RAMs.

Optionally, the instruction includes a fetch instruction and a store instruction; wherein the processing unit is configured to send the fetch instruction, and the fetch instruction is for instructing the data storage management apparatus to fetch data from the storage unit and store the data to a RAM indicated by the fetch instruction.

Optionally, the DMAC is configured to, after completing the fetch instruction, send a storage completion signal; and the processing unit is configured to issue a new

fetch instruction according to the storage completion signal, and read data from the RAM indicated by the fetch instruction.

Optionally, at a same instant, the processing unit and the DMAC access different RAMs.

5 Optionally, at the same instant, the processing unit and the DMAC access different RAMs, including: the at least two RAMs include a first RAM and a second RAM; at a first timing, the processing unit reads first data from the first RAM, and the DMAC writes second data retrieved from the storage unit into the second RAM; and at a second timing, the processing unit reads the second data from the second RAM, and the DMAC
10 writes third data retrieved from the storage unit into the first RAM.

Optionally, the first RAM is a first group of RAMs including a plurality of RAMs, and the second RAM is a second group of RAMs including a plurality of RAMs.

When the first RAM is the first group of RAMs, and the second RAM is the second group of RAMs, at a same instant, the processing unit and the DMAC access different
15 RAMs within one group, respectively; or, at the same instant, the processing unit and the DMAC access RAMs belonging to two groups.

According to a third aspect of the present disclosure, there is provided a chip, including one or more processing cores provided in the second aspect.

According to a fourth aspect of the present disclosure, there is provided a circuit
20 board, including one or more chips provided in the third aspect.

According to a fifth aspect of the present disclosure, there is provided an electronic device, including one or more circuit boards provided in the fourth aspect.

According to a sixth aspect of the present disclosure, there is provided a data storage management method, including: receiving an instruction, and generating and
25 sending a control signal according to the instruction, by a control unit; and accessing, by a direct memory access controller (DMAC), data in RAMs according to the control signal.

According to a seventh aspect of the present disclosure, there is provided an electronic device, including: a memory for storing computer-readable instructions; and
30 one or more processors for executing the computer-readable instructions, which, upon execution, cause the processors to implement any one of the data storage management methods in the sixth aspect.

According to an eighth aspect of the present disclosure, there is provided a non-transitory computer-readable storage medium, the non-transitory computer-readable storage medium storing computer instructions for causing a computer to execute any one of the data storage management methods in the sixth aspect.

5 According to a ninth aspect of the present disclosure, there is provided a computer program product, including computer instructions, wherein, when the computer instructions are executed by a computing device, the computing device execute any one of the data storage management methods in the sixth aspect.

10 The data storage management apparatus provided by the embodiments of the present disclosure receives and responds to an instruction sent from an external processing unit, and reads data from an external storage unit, so that the external processing unit can directly read data required for executing a program from the data storage management apparatus when executing the program. The external processing unit does not need to fetch, via the cache, the data from the external storage unit, thereby
15 avoiding the decrease in computational efficiency caused by cache access failure, and improving the controllability of the program efficiency.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a schematic diagram of reading data in a processing core in the prior art.

20 FIG. 2 is a schematic structural diagram of a data storage management apparatus according to an embodiment of the present disclosure.

FIG. 3 is a schematic structural diagram of a processing core according to an embodiment of the present disclosure.

25 FIG. 4 is a schematic structural diagram of a neural network according to an embodiment of the present disclosure.

FIG. 5 is a schematic structural diagram of a processing core according to an embodiment of the present disclosure.

FIG. 6 is a timing diagram of neural network computation performed by a processing core according to an embodiment of the disclosure.

30 FIG. 7 is a schematic flowchart of a data storage management method according to an embodiment of the present disclosure.

DETAILED DESCRIPTION

In order to make the objective, technical solutions and advantages of the present disclosure clearer, the present disclosure is further described in detail below in conjunction with specific embodiments and with reference to the accompanying drawings. It should be understood that these descriptions are exemplary only and are not intended to limit the scope of the present disclosure. In addition, in the following description, descriptions of well-known structures and techniques are omitted to avoid unnecessary confusion about the concepts of the present disclosure.

In neural network computing, multi-core or many-core chips are often used. The cores in multi-core or many-core architecture all have a certain ability to process data independently, and also have a relatively large internal storage space, for storing their own programs, data and weights.

The performance of the basic computing power of each single core determines the capability of the entire chip in neural network computation. The performance of the basic computing power of each single core is determined by the ideal computing power and memory access efficiency of the computing unit of the single core.

Different storage units have different access speeds. Generally speaking, accessing registers is the fastest, with one access usually taking several hundred ps (picoseconds); followed by Static Random Access Memory (SRAM), with one access generally taking a few hundred ps to a few ns (nanoseconds); next, the memory unit, that is, the Double Data Rate Synchronous Dynamic Random Access Memory (DDR SDRAM), with one access generally taking tens to hundreds of ns; and lastly, other storage accessed by the IO port, such as hard disks, which have slow access speeds, generally at the ms (millisecond) level.

In the case of neural network processing, the general concern is the access of the memory unit by the processing unit. It is known that the speed of the processing unit is very fast, and its basic frequency is generally several hundred MHz (megahertz) to several GHz (gigahertz), that is, at ps to ns level, while the access speed of the memory unit is at tens of ns level. There is a huge difference in speed. How to solve the speed difference between the processing unit and the memory access, to effectively unleash the computing power of the processing unit, is a problem in the design of modern CPUs.

FIG. 1 is a schematic diagram of reading data in a processing core.

As shown in FIG. 1, in the processing core, a cache (Cache) is inserted between the Processing Unit (PU) and the storage unit (Memory). The PU accesses the Memory in a layered and indirect way, that is, the PU directly accesses the Cache, and the PU indirectly accesses the Memory through the Cache. The Cache is a mapping of the Memory, and its contents are a subset of memory contents. The Cache does not have an independent programming space, and the address of the Cache is the same as the address of the memory accessed.

For example, when the PU executes a program, it reads some data from the Memory through the Cache, that is, the Cache saved this part of the data. When the PU needs to use this part of the data again in a short time, the PU will directly fetch it from the Cache.

However, for the program executed by the PU, the Cache is transparent and has no functional significance, that is, the program cannot access the Cache independently, or the program deems that the PU fetches the data from the Memory, while in fact the PU fetches the data from the Cache.

The above scheme has the following defects:

(1) Due to the huge amount of parameters and data used in neural network computing, which usually far exceeds the capacity of the Cache, the measures taken by the Cache to reduce the access failure rate based on the temporal and spatial locality characteristics of the data will not be implemented, thus greatly reducing the computing power of the processing unit.

(2) The Cache circuit is complex, which leads to difficulty in chip design and high cost of the chip.

The data storage management apparatus provided by an embodiment of the present disclosure will be described in detail below. In the description of the present disclosure, it should be noted that the terms "first", "second", "third" and "fourth" are only used for descriptive purposes and cannot be construed as indicating or implying relative importance. In addition, the technical features involved in the different embodiments of the present disclosure described below can be combined with each other as long as they do not conflict with each other.

FIG. 2 is a schematic structural diagram of a data storage management apparatus according to an embodiment of the present disclosure.

As shown in FIG. 2, the data storage management apparatus includes: at least two Random Access Memories (RAMs), a control unit and a Direct Memory Access Controller (DMAC).

5 Optionally, the data storage management apparatus may be provided in a processing core.

The at least two RAMs include RAM_0, RAM_1, ..., RAM_N. The data storage management apparatus is provided with the at least two RAMs, and each RAM can be accessed independently and in parallel.

Optionally, the storage capacities of all RAMs may be the same or different.

10 The control unit is configured to receive an instruction, and generate and send a control signal C_DMAC according to the instruction. The instruction is sent by the processing unit PU located outside the data storage management apparatus.

The DMAC is configured to access data in the RAMs according to the control signal.

15 In an embodiment, the DMAC configured to access data in the RAMs according to the control signal, includes: the DMAC is configured to read data from an external storage unit, Memory, and store the data in the RAM indicated by the control signal, according to the control signal.

In an embodiment, the DMAC configured to access data in the RAMs according to the control signal, includes: the DMAC is configured to read data from the RAM indicated by the control signal, and store the data in the external Memory, according to the control signal.

Optionally, the number of the RAM indicated by the control signal is one or more.

25 Optionally, all the RAMs are uniformly addressed. Optionally, the access addresses of the RAMs are programmed consecutively to reduce the complexity of program control.

In an optional embodiment, all the RAMs and the external storage unit are uniformly addressed.

30 For example, the data storage management apparatus contains two RAMs, namely RAM_0 and RAM_1. The addresses of RAM_0 are 0000H-0FFFH, the addresses of RAM_1 are 1000H-1FFFH, and the addresses of the external memory are 2000H-FFFFH.

Optionally, the addresses accessible by the DMAC are the full address range, specifically, the address segments of all the RAMs and the address segment of the external Memory.

The addresses accessible by the PU are the address segments of all the RAMs.

5 In an embodiment, after the DMAC reads the data from the external memory according to the control signal and stores the data in the RAM indicated by the control signal, the DMAC sends a storage completion signal to the external processing unit. The storage completion signal is for indicating that the external processing unit can read data from the RAM that has just finished storage.

10 FIG. 3 is a schematic structural diagram of a processing core according to an embodiment of the present disclosure.

As shown in FIG. 3, the processing core includes a processing unit PU, a storage unit Memory, and the data storage management apparatus provided in the above embodiments.

15 The PU is configured to send an instruction, wherein the instruction is for instructing the data storage management apparatus to access data in the memory.

The instruction includes a fetch instruction and a store instruction.

The PU is configured to send the instruction, wherein the instruction is for instructing the data storage management apparatus to access data in the memory, including:

20

the PU is configured to send the fetch instruction, and the fetch instruction is for instructing the data storage management apparatus to fetch data from the memory and store the data to a RAM indicated by the fetch instruction. Optionally, after the data is stored in the RAM indicated by the instruction, the DAMC sends a storage completion

25 signal to the PU.

In an embodiment, the PU is further configured to read, from any of the RAMs, data required for executing a program.

Optionally, the PU is configured to issue a new fetch instruction each time after receiving a storage completion signal sent by the DMAC, and read the data from the

30 RAM that has just finished storage.

Specifically, when the PU receives the storage completion signal sent by the DMAC, it issues a new fetch instruction, and then reads data from the RAM that has just finished storage. In this way, the DMAC reads data from the memory according to

the new fetch instruction and stores it in the corresponding RAM, while, in parallel, the PU fetches data from the RAM that has just finished storage to execute the program, improving the efficiency of computation.

Of course, it is also possible that, after receiving the storage completion signal sent
5 by the DMAC, the PU first reads data from the RAM that has just finished storage, and then issues the new fetch instruction.

In an embodiment, the PU and the DMAC access different RAMs at a same point in time.

Specifically, the at least two RAMs include a first RAM and a second RAM. At a
10 first timing, the PU reads first data from the first RAM, and the DMAC writes second data retrieved from the memory into the second RAM. At a second timing, the PU reads the second data from the second RAM, and the DMAC writes third data retrieved from the memory into the first RAM.

Optionally, the PU and the DMAC may access the same RAM at the same time,
15 and the RAM responds to the PU and the DMAC in serial.

Optionally, if the RAM is a dual-port RAM, the PU and the DMAC may also access the same RAM at the same time, and the dual-port RAM responds to the PU and the DMAC in parallel.

It should be noted that, the first RAM may be a first group of RAMs including a
20 plurality of RAMs, and the second RAM may be a second group of RAMs including a plurality of RAMs.

Optionally, the number of RAMs in the first group of RAMs may be the same as or different from the number of RAMs in the second group of RAMs.

When the first RAM is the first group of RAMs, and the second RAM is the second
25 group of RAMs, at a same instant, the PU and the DMAC may access different RAMs within one group, respectively; or, at a same instant, the PU and the DMAC access RAMs that belong to two groups.

Specifically, at a first timing, the PU reads the first data from the first group of RAMs, and the DMAC writes the second data retrieved from the memory into the
30 second group of RAMs; at a second timing, the PU reads the second data from the second group of RAMs, and the DMAC writes the third data retrieved from the memory into the first group of RAMs.

Optionally, when the first RAM is the first group of RAMs, and the second RAM is the second group of RAMs, the processing unit or the DMAC may also access each RAM in the same group in a time-sharing manner.

In the processing core provided by embodiments of the present disclosure, the PU
5 fetching data from the RAM and the DMAC storing data stored in the memory into the
RAM may be processed in parallel, thereby further improving the computing power of
the processing core, and more suitable for neural network computation. In addition,
there is no need to design a complex Cache circuit in the processing core, which saves
the cost of the processing core and reduces the difficulty in chip design. Since there is
10 no need to design a Cache circuit in the processing core, the processing unit does not
need to fetch data from the external memory through the cache. The decrease in
computational efficiency due to cache access failure is thus avoided, and the processing
core may directly fetch data from the RAM of the data storage management apparatus,
thereby improving the controllability of program efficiency.

15 FIG. 4 is a schematic structural diagram of a neural network according to an
embodiment of the present disclosure.

As shown in FIG. 4, the neural network has two layers, the output of the first layer
is used as the input of the second layer, and the output of the second layer is the output
of the entire neural network.

20 FIG. 5 is a schematic structural diagram of a processing core according to an
embodiment of the present disclosure. The processing core shown in FIG. 5 is for
implementing the computation of the neural network shown in FIG. 4.

As shown in FIG. 5, the processing core includes a data storage management
apparatus, a processing unit and a storage unit.

25 The data storage management apparatus includes RAM_0, RAM_1, DAMC, and
a control unit.

It is assumed that the parameters and data for each layer of the neural network are
less than the capacity of a single RAM, that is, both RAM_0 and RAM_1 can
accommodate the parameters and data for computation of a layer of the neural network.

30 It is configured that, at a same instant, the PU and the DMAC access different
RAMs, so that program execution by the PU and data storing by the DMAC may be
processed in parallel, thereby optimizing computation and storage efficiency. For
example, at a first instant, the PU accesses RAM_0, and the DAMC accesses RAM_1.

At a second instant, the PU accesses RAM_1, and the DAMC accesses RAM_0, and so on.

In an embodiment, the PU sends an instruction `lls_dis`. The control unit receives the instruction, generates a control signal `C_DMAC` and sends it to the DMAC. The DMAC reads the data indicated by the instruction from the memory according to the instruction, and stores the data into RAM_0 indicated by the instruction. When the DMAC completes the data storage, it sends a storage completion signal to the PU; the PU receives the storage completion signal, and issues a new instruction to instruct the DMAC to read new data from the memory and store it in RAM_1, and then the PU reads data from RAM_0. After the DMAC stores the data in RAM_1, it sends a storage completion signal. The PU issues an instruction again, and then reads the data from RAM_1. The instruction issued again instructs the DMAC to read new data from memory and store it in RAM_0. In this way, the DMAC stores data and the PU reads data in parallel, so that the efficiency of both computation and storage may be maximized.

FIG. 6 is a timing diagram of neural network computation performed by a processing core according to an embodiment of the disclosure.

As shown in FIG. 6, at t_0 , when the PU reads data from RAM_0, that is, RAM_0 is occupied by the PU executing the program of the first layer of the neural network, and simultaneously, the DMAC writes the data read from the memory into RAM_1. At t_1 , RAM_1 is occupied by the PU executing the program of the second layer of the neural network, and simultaneously, the DMAC writes the data read from the memory into RAM_0. By setting different access addresses for the PU and the DMAC, it is not necessary for the PU to fetch data from the memory, which can reduce the complexity of the program and improve the computing power of the processing core.

It should be understood that the program executed by the PU at t_2 may be set to be the same as the program executed at t_1 , that is, the same computation of the first layer of the neural network may be performed at t_2 ; or the program executed by the PU at t_2 may be set to be different from the program executed at t_1 . The present disclosure is not limited to this.

According to an embodiment of the present disclosure, there is provided a chip, including one or more processing cores provided in the above embodiments.

According to an embodiment of the present disclosure, there is provided a circuit board, including one or more chips provided by the above embodiments.

According to an embodiment of the present disclosure, there is provided an electronic device, including one or more circuit boards provided by the above
5 embodiments.

FIG. 7 is a schematic flowchart of a data storage management method according to an embodiment of the present disclosure. The method includes step S101 to step S102.

Step S101, a control unit receives an instruction, and generates and sends a control
10 signal according to the instruction; and

Step S102, a direct memory access controller (DMAC) accesses data in the random access memories (RAMs) according to the control signal.

The DMAC accesses data in the RAMs according to the control signal, including: the DMAC reads data from an external memory, and stores the data in the RAM
15 indicated by the control signal, according to the control signal.

The DMAC accesses data in the RAMs according to the control signal, including: the DMAC reads data from the RAM indicated by the control signal, and stores the data in the external memory, according to the control signal.

It can be understood that, the data storage management method provided by the
20 embodiments may be executed by the data storage management apparatus provided by the above embodiments of the present disclosure. Therefore, the same features are not described here again.

An embodiment of the present disclosure provides a schematic flowchart of a method of processing data by a processing core.

25 The method includes step S201 to step S202.

Step S201, the processing unit sends a fetch instruction;

Step S202, the data storage management apparatus reads data from a storage unit according to the fetch instruction, and stores the data in the RAM of the data storage management apparatus indicated by the fetch instruction.

30 In an embodiment, after the data storage management apparatus stores the data in the RAM of the data storage management apparatus indicated by the fetch instruction, it sends a storage completion signal.

The processing unit sends a new fetch instruction each time after receiving the storage completion signal sent by the DMAC, and reads the data from the RAM indicated by the just completed fetch instruction.

5 In an embodiment, the processing unit and the DMAC access different RAMs at a same instant.

Specifically, at a first timing, the processing unit reads the first data from the first RAM, and the DMAC writes the second data retrieved from the memory into the second RAM. At a second timing, the PU reads the second data from the second RAM, and the DMAC writes the third data retrieved from the memory into the first RAM.

10 It can be understood that, the method of processing data by a processing core provided by the embodiments may be executed by the processing core provided by the above embodiments of the present disclosure. Therefore, the same features are not described here again.

15 According to an embodiment of the present disclosure, there is provided an electronic device, including: a memory for storing computer-readable instructions; and one or more processors for executing the computer-readable instructions, which, upon execution, cause the processors to implement the data storage management methods in the above embodiments.

20 According to an embodiment of the present disclosure, there is provided a non-transitory computer-readable storage medium, the non-transitory computer-readable storage medium storing computer instructions for causing a computer to execute the data storage management methods in the above embodiments.

25 According to an embodiment of the present disclosure, there is provided a computer program product, including computer instructions, wherein, when the computer instructions are executed by a computing device, the computing device executes the data storage management methods in the above embodiments.

30 It should be understood that the above specific embodiments of the present disclosure are used only for exemplary illustration or explanation of the principle of the present disclosure, but not to limit the present disclosure. Therefore, any modifications, equivalent replacements, improvements, etc. made without departing from the spirit and scope of the present disclosure should be included within the protection scope of the present disclosure. Furthermore, the appended claims of this disclosure are intended

to cover all variations and modifications that fall within the scope and boundaries of the appended claims, or the equivalents of such scope and boundaries.

CLAIMS

1. A data storage management apparatus, comprising:
at least two random access memories (RAMs);
a control unit, configured to receive an instruction, and generate and send a control
5 signal according to the instruction; and
a direct memory access controller (DMAC), configured to access data in the
RAMs according to the control signal.
2. The apparatus according to claim 1, wherein the DMAC configured to access
data in the RAMs according to the control signal, comprising:
10 the DMAC is configured to read data from an external storage unit and store the
data in the RAM indicated by the control signal, according to the control signal; or
the DMAC is configured to read data from the RAM indicated by the control signal
and store the data in the external storage unit, according to the control signal.
3. The apparatus according to claim 1 or 2, wherein the number of the RAM
15 indicated by the control signal is one or more.
4. The apparatus according to any one of claims 1-3, wherein all the RAMs and an
external storage unit are uniformly addressed; or, all the RAMs are uniformly
addressed.
5. The apparatus according to any one of claims 1-4, wherein an address range
20 accessible by the DMAC is address segments of all the RAMs and address segment of
the external storage unit.
6. A processing core, comprising a processing unit, a storage unit and the data
storage management apparatus according to any one of claims 1-5; wherein
the processing unit is configured to send an instruction, and the instruction is for
25 instructing the data storage management apparatus to access data in the storage unit;
and
the processing unit is further configured to read data required for executing a
program from any of the RAMs.
7. The processing core according to claim 6, wherein
30 the instruction includes a fetch instruction and a store instruction; and
the processing unit is configured to send the fetch instruction, and the fetch
instruction is for instructing the data storage management apparatus to fetch data from
the storage unit and store the data to a RAM indicated by the fetch instruction.

8. The processing core according to claim 7, wherein the DMAC is configured to, after completing the fetch instruction, send a storage completion signal; and

the processing unit is configured to issue a new fetch instruction according to the storage completion signal, and read data from the RAM indicated by the fetch
5 instruction.

9. The processing core according to any one of claims 6-8, wherein, at a same instant, the processing unit and the DMAC access different RAMs.

10. The processing core according to claim 7, wherein, at the same instant, the processing unit and the DMAC access different RAMs, comprising:

10 the at least two RAMs include a first RAM and a second RAM;

at a first timing, the processing unit reads first data from the first RAM, and the DMAC writes second data retrieved from the storage unit into the second RAM; and

at a second timing, the processing unit reads the second data from the second RAM, and the DMAC writes third data retrieved from the storage unit into the first RAM.

15