CLAIMS:

1. A method for accessing data in a memory coupled to a processor, comprising:

receiving a memory reference instruction for accessing data of a first size at an address in the memory, the memory reference instructions specifying the first size;

determining, by the processor, an alignment size of the address, wherein the determining the alignment size comprises counting a number of trailing zeros in the address in binary representation;

determining, by the processor, a greatest common divisor of the first size and

the determined alignment size; and

accessing the data of the first size in one or more groups of data by accessing each group of data block concurrently, the groups of data having sizes that are multiples of a size of the greatest common divisor.

2. The method of claim 1, wherein the first size is larger than the determined alignment size.

3. The method of claim 1, wherein each group of data is not larger than a predefined maximum block concurrency size.

4. The method of claim 1, further comprising determining the first size based on an opcode of the memory reference instruction.

5. The method of claim 1, wherein counting the number of trailing zeros in the address in binary representation further comprises raising two to a power of the number of trailing zeros in binary representation.

6. A system for accessing data, comprising:

a memory; and

processor configured to perform a method comprising:

receiving a memory reference instruction for accessing data of a first size at an address in the memory;

determining an alignment size of the address, wherein the determining the alignment size comprises counting a number of trailing zeros in the address in binary representation;

determining a greatest common divisor of the first size and

the alignment size; and

accessing the data of the first size in one or more groups of data by accessing each group of data block concurrently, the groups of data having sizes that are multiples of the greatest common divisor.

7. The system of claim 6, wherein the first size and the determined alignment size are in number of bytes, wherein the first size is 2.sup.A and the determined alignment size is 2.sup.B, wherein A is an integer greater than B, which is a nonnegative integer.

8. The system of claim 6, wherein each group of data is not larger than a predefined maximum block concurrency size.

9. The system of claim 6, wherein the processor comprises a vector processor, and the memory reference instruction comprises a vector instruction.

10. The system of claim 6, wherein counting the number of trailing zeros in the address in binary representation further comprises raising two to a power of the number of trailing zeros in binary representation.

11. A computer program product for accessing data, the computer program product comprising:

a computer readable storage medium having machine instructions embodied therewith, the machine instructions readable by a processor to cause the processor to perform a method comprising:

receiving a memory reference instruction for accessing data of a first size at an address in the memory;

determining an alignment size of the address, wherein the determining the alignment size comprises counting a number of trailing zeros in the address in binary representation;

determining a greatest common divisor of the first size and

the alignment size; and

accessing the data of the first size in one or more groups of data by accessing each group of data block concurrently, the groups of data having sizes that are multiples of the greatest common divisor.

12. The computer program product of claim 11, wherein the first size is larger than the determined alignment size.

13. The computer program product of claim

11, wherein each group of data is not larger than a predefined maximum block concurrency size.

14. The computer program product of claim 11, wherein the method further comprises determining the first size based on an opcode of the memory reference instruction.

15. A method for accessing data in a memory coupled to a processor, comprising:

receiving a memory reference instruction for accessing data of a first size at an address in the memory, the memory reference instruction specifying the first size;

determining, by the processor, an alignment size of the address, wherein the determining the alignment size comprises counting a number of trailing zeros in the address in binary representation;

determining a greatest common divisor of the first size and the determined alignment size; and

accessing the data of the first size in one or more groups of data, the groups of data having sizes that are multiples of a size of the greatest common divisor.

16. The method of claim 15, wherein the first size is larger than the determined alignment size.

17. The method of claim 15, wherein the first size and the determined alignment size are in number of bytes, wherein the first size is not a power of two size and the determined alignment size is a power of two size.

18. The method of claim 15, further comprising determining the first size based on an operand of the memory reference instruction.

19. The method of claim 15, wherein the accessing the data of the first size is performed when the greatest common divisor is not larger than a predefined maximum block concurrency size.

20. A system for accessing data, comprising:

a memory; and

a processor configured to perform a method comprising:

receiving a memory reference instruction for accessing a data space with a first size at an address in the memory, the memory reference instruction specifying the first size;

determining an alignment size of the address;

determining a greatest common divisor of the first size and the determined alignment size; and

accessing the data of the first size in one or more groups of data, the groups of data having sizes that are multiples of a size of the greatest common divisor.

21. The system of claim 20, wherein the first size is larger than the determined alignment size.

22. The system of claim 20, wherein the first size and the determined alignment size are in number of bytes, wherein the first size is not a power of two size and the determined alignment size is a power of two size.

23. The system of claim 20, further comprising determining the first size based on an operand of the memory reference instruction.

24. The system of claim 20, wherein the processor comprises a vector processor, and the memory reference instruction comprises a vector instruction.

25. A computer program product for accessing data, the computer program product comprising:

a computer readable storage medium having machine instructions embodied therewith, the machine instructions readable by a processor to cause the processor to perform a method comprising:

receiving a memory reference instruction for accessing a data space with a first size at an address in the memory, the memory reference instruction specifying the first size;

determining an alignment size of the address;

determining a greatest common divisor of the first size and the determined alignment size; and

accessing the data of the first size in one or more groups of data, the groups of data having sizes that are multiples of a size of the greatest common divisor.