CLAIMS

1. A computer program product for executing a machine instruction in a central processing unit, the computer program product comprising:

a computer readable storage medium readable by a processing circuit and storing instructions for execution by the processing circuit for performing a method comprising:

obtaining, by a processor, a machine instruction for execution, the machine instruction being defined for computer execution according to a computer architecture, the machine instruction comprising:

at least one opcode field to provide an opcode, the opcode identifying a convert to zoned from decimal floating point function;

a first register field designating a first register, the first register comprising a first operand;

a second register field and a displacement field, wherein contents of a second register designated by the second register field are combined with contents of the displacement field to form an address of a second operand; and

a mask field, the mask field comprising one or more controls used during execution of the machine instruction; and

executing the machine instruction, the executing comprising:

converting at least a portion of the first operand in a decimal floating point format to a zoned format; and

placing a result of the converting at a location designated by the address of the second operand.

2. The computer program product of claim 1, wherein the opcode field, when a first value, indicates a long decimal floating point source and a zoned result.

3. The computer program product of claim 1, wherein the opcode, when a second value, indicates an extended decimal floating point source and a zoned result.

4. The computer program product of claim 1, wherein the mask field comprises a sign control used to indicate whether the second operand has a sign field.

5. The computer program product of claim 1, wherein the mask field comprises a zone control used to determine a value for a zone field of the second operand.

6. The computer program product of claim 1, wherein the mask field comprises a plus-sign-code control used to encode a plus sign.

7. The computer program product of claim 1, wherein the mask field comprises a force-plus-zero control used in determining a sign of a result placed in the second operand.

8. The computer program product of claim 1, wherein the mask field comprises a zone field and a sign field, and wherein the method further comprises using at least one of the zone field and the sign field to determine a value for at least one of the sign field and a field code of the result stored in the second operand.

9. The computer program product of claim 1, wherein the machine instruction comprises a length field specifying at least one of a number of rightmost significand digits of the first operand to be converted and a length of the second operand.

10. The computer program product of claim 1, wherein the executing comprises reading the at least a portion of the first operand from at least one floating point register, and wherein the second operand to include the result is in memory.

11. A computer system for executing a machine instruction in a central processing unit, the computer system comprising:

a memory; and

a processor in communications with the memory, wherein the computer system is configured to perform a method, said method comprising:

obtaining a machine instruction for execution, the machine instruction being defined for computer execution according to a computer architecture, the machine instruction comprising:

at least one opcode field to provide an opcode, the opcode identifying a convert to zoned from decimal floating point function;

a first register field designating a first register, the first register comprising a first operand;

a second register field and a displacement field, wherein contents of a second register designated by the second register field are combined with contents of the displacement field to form an address of a second operand; and

a mask field, the mask field comprising one or more controls used during execution of the machine instruction; and

executing the machine instruction, the executing comprising:

converting at least a portion of the first operand in a decimal floating point format to a zoned format; and

placing a result of the converting at a location designated by the address of the second operand.

12. The computer system of claim 11, wherein the mask field comprises a sign control used to indicate whether the second operand has a sign field.

13. The computer system of claim 11, wherein the mask field comprises a zone control used to determine a value for a zone field of the second operand.

14. The computer system of claim 11, wherein the mask field comprises a plus-sign-code control used to encode a plus sign.

15. The computer system of claim 11, wherein the mask field comprises a force-plus-zero control used in determining a sign of a result placed in the second operand.

16. The computer system of claim 11, wherein the mask field comprises a zone field and a sign field, and wherein the method further comprises using at least one of the zone field and the sign field to determine a value for at least one of the sign field and a field code of the result stored in the second operand.

17. The computer system of claim 11, wherein the machine instruction comprises a length field specifying at least one of a number of rightmost significand digits of the first operand to be converted and a length of the second operand.

18. A method for executing a machine instruction in a central processing unit, the method comprising:

obtaining, by a processor, a machine instruction for execution, the machine instruction being defined for computer execution according to a computer architecture, the machine instruction comprising:

at least one opcode field to provide an opcode, the opcode identifying a convert to zoned from decimal floating point function;

a first register field designating a first register, the first register comprising a first operand;

a second register field and a displacement field, wherein contents of a second register designated by the second register field are combined with contents of the displacement field to form an address of a second operand; and

a mask field, the mask field comprising one or more controls used during execution of the machine instruction; and

executing the machine instruction, the executing comprising:

converting at least a portion of the first operand in a decimal floating point format to a zoned format; and

placing a result of the converting at a location designated by the address of the second operand.

19. The method of claim 18, wherein the mask field comprises a sign control used to indicate whether the second operand has a sign field, a zone control used to determine a value for a zone field of the second operand, a plus-sign-code control used to encode a plus sign, and a force-plus-zero control used in determining a sign of a result placed in the second operand.

20. The method of claim 18, wherein the machine instruction comprises a length field specifying at least one of a number of rightmost significand digits of the first operand to be converted and a length of the second operand.

CLAIMS

1. A computer program product for executing a machine instruction in a central processing unit, the computer program product comprising:

a non-transitory computer readable storage medium readable by a processing circuit and storing instructions for execution by the processing circuit for performing a method comprising:

obtaining, by a processor, a machine instruction for execution, the machine instruction being defined for computer execution according to a computer architecture, the machine instruction comprising:

at least one opcode field to provide an opcode, the opcode identifying a convert to zoned from decimal floating point function;

a first register field designating a first register, the first register comprising a first operand;

a second register field and a displacement field, wherein contents of a second register designated by the second register field are combined with contents of the displacement field to form an address of a second operand; and

a mask field, the mask field comprising one or more controls used during execution of the machine instruction, the mask field comprising a zone control to control a zone field of the second operand; and

executing the machine instruction, the executing comprising:

converting at least a portion of the first operand in a decimal floating point format to a zoned format, wherein the converting comprises:

converting a first plurality of bits of the first operand to corresponding triplets of zoned digits, the triplets of zoned digits being provided as a first portion of a result;

converting a second plurality of bits of the first operand to a most significant zoned digit of the result; and

providing the zone field for each zoned digit of the result; and

placing the result at a location designated by the address of the second operand, wherein the result is stored to the location in the zoned format absent issuing one or more additional machine instructions to convert the stored result to the zoned format.

2. The computer program product of claim 1, wherein the opcode field, when a first value, indicates a long decimal floating point source and a zoned result.

3. The computer program product of claim 1, wherein the opcode, when a second value, indicates an extended decimal floating point source and a zoned result.

4. The computer program product of claim 1, wherein the mask field comprises a sign control used to indicate whether the second operand has a sign field.

5. The computer program product of claim 1, wherein the mask field comprises a plus-sign-code control used to encode a plus sign.

6. The computer program product of claim 1, wherein the mask field comprises a force-plus-zero control used in determining a sign of a result placed in the second operand.

7. The computer program product of claim 1, wherein the mask field comprises a sign field, and wherein the method further comprises using the sign field to determine a value for a sign field of the result stored in the second operand.

8. The computer program product of claim 1, wherein the machine instruction comprises a length field specifying at least one of a number of rightmost significand digits of the first operand to be converted and a length of the second operand.

9. The computer program product of claim 1, wherein the executing comprises reading the at least a portion of the first operand from at least one floating point register, and wherein the second operand to include the result is in memory.

10. A computer system for executing a machine instruction in a central processing unit, the computer system comprising:

a memory; and

a processor in communications with the memory, wherein the computer system is configured to perform a method, said method comprising:

obtaining a machine instruction for execution, the machine instruction being defined for computer execution according to a computer architecture, the machine instruction comprising:

at least one opcode field to provide an opcode, the opcode identifying a convert to zoned from decimal floating point function;

a first register field designating a first register, the first register comprising a first operand;

a second register field and a displacement field, wherein contents of a second register designated by the second register field are combined with contents of the displacement field to form an address of a second operand; and

a mask field, the mask field comprising one or more controls used during execution of the machine instruction, the mask field comprising a zone control to control a zone field of the second operand; and

executing the machine instruction, the executing comprising:

converting at least a portion of the first operand in a decimal floating point format to a zoned format, wherein the converting comprises:

converting a first plurality of bits of the first operand to corresponding triplets of zoned digits, the triplets of zoned digits being provided as a first portion of a result;

converting a second plurality of bits of the first operand to a most significant zoned digit of the result; and

providing the zone field for each zoned digit of the result; and

placing the result at a location designated by the address of the second operand, wherein the result is stored to the location in the zoned format absent issuing one or more additional machine instructions to convert the stored result to the zoned format.

11. The computer system of claim 10, wherein the mask field comprises a sign control used to indicate whether the second operand has a sign field.

12. The computer system of claim 10, wherein the mask field comprises a plus-sign-code control used to encode a plus sign.

13. The computer system of claim 10, wherein the mask field comprises a force-plus-zero control used in determining a sign of a result placed in the second operand.

14. The computer system of claim 10, wherein the mask field comprises a sign field, and wherein the method further comprises using the sign field to determine a value for a sign field of the result stored in the second operand.

15. The computer system of claim 10, wherein the machine instruction comprises a length field specifying at least one of a number of rightmost significand digits of the first operand to be converted and a length of the second operand.