CLAIMS

1. A computer program product for facilitating processing within a computing environment, the computer program product comprising:

one or more computer readable storage media and program instructions collectively stored on the one or more computer readable storage media to perform a method comprising:

executing an instruction to perform a conversion between decimal formats, wherein one of the decimal formats is a zoned decimal format, and wherein the executing includes:

obtaining a value from at least one register specified using the instruction;

converting at least a portion of the value from one decimal format to another decimal format different from the one decimal format to provide a converted result;

writing a result obtained from the converted result into a single register specified using the instruction;

obtaining, using a field of the instruction, at least one validity control, the at least one validity control to determine whether one or more particular validity checks relating to the instruction are to be performed, the at least one validity control including a digit validation control and a sign validation control; and

performing, as part of executing the instruction, the one or more particular validity checks based on the at least one validity control indicating the one or more particular validity checks are to be performed, wherein the performing the one or more particular validity checks includes checking a sign code of a source value of the instruction for a general operand data exception based on the digit validation control being set to a selected value to indicate validity

checking and the sign validation control being set to a chosen value to indicate validity checking.

2. The computer program product of claim 1, wherein the one decimal format is the zoned decimal format, and the other decimal format is a packed decimal format.

3. The computer program product of claim 2 , wherein the obtaining the value comprises:

obtaining a first value from a first vector register specified using at least one field of the instruction;

obtaining a second value from a second vector register specified using at least one field of the instruction; and

concatenating the first value and the second value to obtain the value.

4. The computer program product of claim 2, wherein the writing comprises:

determining a number of digits of the converted result to be written as the result;

selecting the determined number of digits of the converted result to be written as the result; and

writing the selected determined number of digits into the single register.

5. The computer program product of claim 4, wherein the determining the number of digits comprises checking a control specified using the instruction, and the single register is a single vector register.

6. The computer program product of claim 2, wherein based on the digit validation control being set to a value different from the selected value indicating validity checking for the general operand data exception, the sign code of the source vector is checked for validity for setting a condition code resulting from execution of the instruction.

7. The computer program product of claim 2, wherein based on the sign validation control being set to a value different from the chosen value indicating validity checking for

the general operand data exception, the sign code of the source vector is checked for validity for setting a condition code resulting from execution of the instruction.

8. The computer program product of claim 1, wherein the one decimal format is a packed decimal format and wherein the other decimal format is the zoned decimal format.

9. The computer program product of claim 8, wherein the converting at least a portion of the value comprises converting selected digits of the value to obtain the converted result.

10. The computer program product of claim 9, wherein the selected digits comprise a leftmost fifteen digits of the value.

11. The computer program product of claim 9, wherein the selected digits comprise a rightmost sixteen digits of the value, and wherein the converting further includes converting a sign of the value.

12. The computer program product of claim 8, wherein the writing comprises writing the converted result to a selected portion of the single register, the single register being a vector register.

13. A computer system for facilitating processing within a computing environment, the computer system comprising:

a memory; and

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

executing an instruction to perform a conversion between decimal formats, wherein one of the decimal formats is a zoned decimal format, and wherein the executing includes:

obtaining a value from at least one register specified using the instruction;

converting at least a portion of the value from one decimal format to another decimal format different from the one decimal format to provide a converted result;

writing a result obtained from the converted result into a single register specified using the instruction;

obtaining, using a field of the instruction, at least one validity control, the at least one validity control to determine whether one or more particular validity checks relating to the instruction are to be performed, the at least one validity control including a digit validation control and a sign validation control; and

performing, as part of executing the instruction, the one or more particular validity checks based on the at least one validity control indicating the one or more particular validity checks are to be performed, wherein the performing the one or more particular validity checks includes checking a sign code of a source value of the instruction for a general operand data exception based on the digit validation control being set to a selected value to indicate validity checking and the sign validation control being set to a chosen value to indicate validity checking.

14. The computer system of claim 13, wherein the one decimal format is the zoned decimal format, and the other decimal format is a packed decimal format.

15. The computer system of claim 14, wherein the obtaining the value comprises:

obtaining a first value from a first vector register specified using at least one field of the instruction;

obtaining a second value from a second vector register specified using at least one field of the instruction; and

concatenating the first value and the second value to obtain the value.

16. The computer system of claim 14, wherein the writing comprises:

determining a number of digits of the converted result to be written as the result;

selecting the determined number of digits of the converted result to be written as the result; and

writing the selected determined number of digits into the single register.

17. The computer system of claim 13, wherein the one decimal format is a packed decimal format and wherein the other decimal format is the zoned decimal format.

18. A computer-implemented method of facilitating processing within a computing environment, the computer-implemented method comprising:

executing an instruction to perform a conversion between decimal formats, wherein one of the decimal formats is a zoned decimal format, and wherein the executing includes:

obtaining a value from at least one register specified using the instruction;

converting at least a portion of the value from one decimal format to another decimal format different from the one decimal format to provide a converted result;

writing a result obtained from the converted result into a single register specified using the instruction;

obtaining, using a field of the instruction, at least one validity control, the at least one validity control to determine whether one or more particular validity checks relating to the instruction are to be performed, the at least one validity control including a digit validation control and a sign validation control; and

performing, as part of executing the instruction, the one or more particular validity checks based on the at least one validity control indicating

the one or more particular validity checks are to be performed, wherein the performing the one or more particular validity checks includes checking a sign code of a source value of the instruction for a general operand data exception based on the digit validation control being set to a selected value to indicate validity checking and the sign validation control being set to a chosen value to indicate validity checking.

19. The computer-implemented method of claim 18, wherein the one decimal format is the zoned decimal format, and the other decimal format is a packed decimal format.

20. The computer-implemented method of claim 19, wherein the obtaining the value comprises:

obtaining a first value from a first vector register specified using at least one field of the instruction;

obtaining a second value from a second vector register specified using at least one field of the instruction; and

concatenating the first value and the second value to obtain the value.

21. The computer-implemented method of claim 19, wherein the writing comprises:

determining a number of digits of the converted result to be written as the result;

selecting the determined number of digits of the converted result to be written as the result; and

writing the selected determined number of digits into the single register.

22. The computer-implemented method of claim 18, wherein the one decimal format is a packed decimal format and wherein the other decimal format is the zoned decimal format.

23. A computer program product for facilitating processing within a computing environment, the computer program product comprising:

one or more computer readable storage media and program instructions collectively stored on the one or more computer readable storage media to perform a method comprising:

executing a vector unpack zoned high instruction to perform an unpack operation, the unpack operation including a conversion between a signed packed decimal format and a zoned decimal format and validity checking, wherein the executing includes:

obtaining a value from at least one register specified using the instruction;

converting selected leftmost digits of the value from the signed packed decimal format to the zoned decimal format to provide a converted result;

writing a result obtained from the converted result into a single register specified using the instruction;

obtaining, using a field of the instruction, a digit validation control and a sign validation control; and

performing, as part of executing the vector unpack zoned high instruction, a sign code validity check of an input value of the instruction for a general operand data exception based on the digit validation control being set to a selected value to indicate validity checking and the sign validation control being set to a chosen value to indicate validity checking.

24. The computer program product of claim 23, wherein the selected digits comprise a leftmost fifteen digits of the value.

25. A computer program product for facilitating processing within a computing environment, the computer program product comprising:

one or more computer readable storage media and program instructions collectively stored on the one or more computer readable storage media to perform a method comprising:

executing a vector unpack zoned low instruction to perform an unpack operation, the unpack operation including a conversion between a signed packed decimal format and a zoned decimal format and validity checking, wherein the executing includes:

obtaining a value from at least one register specified using the instruction;

converting selected rightmost digits of the value from the signed packed decimal format to the zoned decimal format to provide a converted result;

writing a result obtained from the converted result into a single register specified using the instruction;

obtaining, using a field of the instruction, a digit validation control and a sign validation control; and

performing, as part of executing the vector unpack zoned low instruction, a sign code validity check of an input value of the instruction for a general operand data exception based on the digit validation control being set to a selected value to indicate validity checking and the sign validation control being set to a chosen value to indicate validity checking.

* * * * *