MULTIPLE SOURCES FOR INSTRUCTION DECODE

TECHNICAL FIELD

This invention relates to digital signal processors, and more particularly to having multiple sources providing instructions to the decoder.

BACKGROUND

Digital signal processing is concerned with the representation of signals in digital form and the transformation or processing of such signal representation using numerical computation. Digital signal processing is a core technology for many of today's high technology products in fields such as wireless communications, networking, and multimedia. One reason for the prevalence of digital signal processing technology has been the development of low cost, powerful digital signal processors (DSPs) that provide engineers the reliable computing capability to implement these products cheaply and efficiently. Since the development of the first DSPs, DSP architecture and design have evolved to the point where even sophisticated real-time processing of video-rate sequences may be performed.

DSPs are often used for a variety of multimedia applications such as digital video, imaging, and audio. DSPs may manipulate the digital signals to create and open such multimedia files.

MPEG-1 (Motion Picture Expert Group), MPEG-2, MPEG-4 and H.263 are digital video compression standards and file formats. These standards achieve a high compression rate of the digital video signals by storing mostly changes from one video frame to another, instead of storing each entire frame. The video information may then be further compressed using a number of different techniques.

The DSP may be used to perform various operations on the video information during compression. These operations may include motion search and spatial interpolation algorithms. The primary intention is to measure distortion between blocks within adjacent frames. These operations are computationally intensive and may require high data throughput.

The MPEG family of standards is evolving to keep pace with the increasing bandwidth requirements of multimedia applications and files. Each new version of the standard presents more sophisticated algorithms that place

even greater processing requirements on the DSPs used in MPEG compliant video processing equipment.

Video processing equipment manufacturers often rely on application-specific integrated circuits (ASICs) customized for video encoding under the MPEG and H.263 standards. However, ASICs are complex to design, costly to produce and less flexible in their application than general-purpose DSPs.

SUMMARY OF THE INVENTION

According to the invention there is provided a method of providing instructions to a decoder within a processor comprising: providing a selected instruction to a decoder from a plurality of sources.

The method may further comprise collecting a plurality of instructions from the plurality of sources.

The method may further comprise collecting the instructions in a multiplexer.

The method may further comprise transmitting width bits to the decoder corresponding to a size of the selected instruction.

The method may further comprise collecting a plurality of width bits corresponding to the size of the instructions from the plurality of sources.

The method may further comprise collecting the plurality of width bits in a second multiplexer.

The method may further comprise determining the selected instruction based on a priority schedule.

The method may further comprise providing the instructions in a digital signal processor.

According to the invention there is also provided a method of obtaining instructions at a decoder within a processor comprising: collecting a plurality of instructions from a plurality of instruction sources; collecting a plurality of width bits corresponding to the plurality of instructions; determining one of the plurality of instructions to be processed; and transferring the selected one of the plurality of instructions and the corresponding width bit to the decoder.

The method may further comprise collecting instructions from an Icache/Alignment unit.

The method may further comprise collecting instructions from a loop buffer.

The method may further comprise collecting instructions from an emulation instruction register.

The method may further comprise obtaining instructions in a digital signal processor.

The method may further comprise collecting the plurality of instructions having the same format.

The method may further comprise determining the instruction to be processed based on a priority schedule.

According to the invention, there is also provided a processor comprising: a plurality of instruction sources which provide a plurality of instructions; a first multiplexer which receives the plurality of instructions from the plurality of instruction sources, wherein the first multiplexer selects a first instruction to be executed from the plurality of instructions; and a decoder which receives the selected first instruction from the first multiplexer.

The processor may further comprise a second multiplexer which receives a plurality of signals from the instruction sources indicative of the size of the instructions, wherein the second multiplexer provides to the decoder one of the plurality of signals which corresponds to the selected first instruction.

Each of the plurality of instructions may be in the same format.

The decoder may receive the selected first instruction directly from the multiplexer.

The first multiplexer may select a first instruction to be executed from the plurality of instructions based on a priority schedule.

The processor may be a digital signal processor.

According to the invention, there is also provided an apparatus, including instructions residing on a machine-readable storage medium, for use in a machine system to handle a plurality of instructions, the instructions causing the machine to: select an instruction from a plurality of sources; and provide the selected instruction to the decoder.

The machine may collect a plurality of instructions from the plurality of sources.

The instructions may be collected in a multiplexer.

A plurality of width bits corresponding to the size of the instructions may be collected from the plurality of sources.

According to the invention, there is also provided a method of providing instructions to a decoder within a processor comprising: providing a selected instruction and a second instruction to a decoder from a plurality of sources, wherein the selected instruction does not pass through an instruction cache.

The method may further comprise collecting a plurality of instructions from the plurality of sources.

The method may further comprise collecting the instructions in a multiplexer.

The method may further comprise transmitting width bits to the decoder corresponding to a size of the selected instruction.

The method may further comprise collecting a plurality of width bits corresponding to the size of the instructions from the plurality of sources.

The method may further comprise collecting the plurality of width bits in a second multiplexer.

The method may further comprise determining the selected instruction based on a priority schedule.

The method may further comprise providing the instructions in a digital signal processor.

According to the invention, there is also provided a method of obtaining instructions at a decoder within a processor comprising: collecting a plurality of instructions from a plurality of instruction sources, wherein said collecting a plurality of instructions includes collecting a first instruction, wherein the first instruction does not pass through an instruction cache; collecting a plurality of width bits corresponding to the plurality of instructions; determining one of the plurality of instructions to be processed; and transferring the selected one of the plurality of instructions and the corresponding width bit to the decoder.

The method may further comprise collecting instructions from an Icache/Alignment unit.

The method may further comprise collecting instructions from a loop buffer.

The method may further comprise collecting instructions from an emulation instruction register.

The method may further comprise obtaining instructions in a digital signal processor.

The method may further comprise collecting the plurality of instructions having the same format.

The method may further comprise determining the instruction to be processed based on a priority schedule.

According to the invention, there is also provided a processor comprising: a plurality of instruction sources which provide a plurality of instructions; a first multiplexer which receives the plurality of instructions from the plurality of instruction sources, wherein the first multiplexer selects a first instruction to be executed from the plurality of instructions, and wherein the first instruction has not passed through an instruction cache; and a decoder which receives the selected first instruction from the first multiplexer.

The processor may further comprise a second multiplexer which receives a plurality of signals from the instruction sources indicative of the size of the instructions, wherein the second multiplexer provides to the decoder one of the

plurality of signals which corresponds to the selected first instruction.

Each of the plurality of instructions may be in the same format.

The decoder may receive the selected first instruction directly from the multiplexer.

The first multiplexer may select a first instruction to be executed from the plurality of instructions based on a priority schedule.

The processor may be a digital signal processor.

According to the invention, there is also provided an apparatus, including instructions residing on a machine-readable storage medium, for use in a machine system to handle a plurality of instructions, the instructions causing the machine to: select an instruction from a plurality of sources; and provide the selected instruction to the decoder, wherein the selected instruction has not passed through an instruction cache.

The machine may collect a plurality of instructions from the plurality of sources.

The instructions may be collected in a multiplexer.

A plurality of width bits corresponding to the size of the instructions may be collected from the plurality of sources.

According to the invention, there is also provided a method of providing instructions to a decoder within a processor comprising: collecting in a first multiplexer a selected instruction from a plurality of sources; collecting in a second multiplexer a width signal from the one of the plurality of sources, the width signal specifying the size of the selected instruction; and providing the selected instruction and the width signal to the decoder.

The method may further comprise collecting a plurality of instructions from the plurality of sources.

The method may further comprise collecting a plurality of width bits corresponding to the size of the plurality of instructions from the plurality of sources.

The method may further comprise transmitting width bits to the decoder corresponding to a size of the selected instruction.

The method may further comprise collecting the plurality of width bits in the second multiplexer.

The method may further comprise determining the selected instruction based on a priority schedule.

The method may further comprise providing the instructions in a digital signal processor.

The second multiplexer may not collect instructions.

According to the invention, there is also provided a method of obtaining instructions at a decoder within a processor

comprising: collecting in a first multiplexer a plurality of instructions from a plurality of instruction sources; collecting in a second multiplexer a plurality of width bits corresponding to the plurality of instructions; determining one of the plurality of instructions to be processed; and transferring the selected one of the plurality of instructions and the corresponding width bit to the decoder.

The method may further comprise collecting instructions from an Icache/Alignment unit.

The method may further comprise collecting instructions from a loop buffer.

The method may further comprise collecting instructions from an emulation instruction register.

The method may further comprise obtaining instructions in a digital signal processor.

The method may further comprise collecting the plurality of instructions having the same format.

The method may further comprise determining the instruction to be processed based on a priority schedule.

The second multiplexer may not collect instructions.

According to the invention, there is also provided a processor comprising: a plurality of instruction sources which provide a plurality of instructions; a first multiplexer which receives the plurality of instructions

from the plurality of instruction sources, wherein the first multiplexer selects a first instruction to be executed from the plurality of instructions; a decoder which receives the selected first instruction from the first multiplexer; and a second multiplexer which receives a plurality of signals from the instruction sources indicative of the size of the instructions, wherein the second multiplexer provides to the decoder one of the plurality of signals which corresponds to the selected first instruction.

The second multiplexer may not receive instructions.

Each of the plurality of instructions may be in the same format.

The decoder may receive the selected first instruction directly from the multiplexer.

The first multiplexer may select a first instruction to be executed from the plurality of instructions based on a priority schedule.

The processor may be a digital signal processor.

The first multiplexer may be coupled to each of the plurality of instruction sources, wherein the second multiplexer is separately coupled to each of the plurality of instruction sources, and wherein the first multiplexer and the second multiplexer are coupled in parallel to the decoder.

According to the invention, there is also provided an apparatus, including instructions residing on a machine-readable storage medium, for use in a machine system to

handle a plurality of instructions, the instructions causing the machine to: collect in a first multiplexer a selected instruction from a plurality of sources; collect in a second multiplexer a width signal from the one of the plurality of sources, the width signal specifying the size of the selected instruction; and provide the selected instruction and the width signal to a decoder.

The machine may collect in the first multiplexer a plurality of instructions from the plurality of sources.

A plurality of width bits corresponding to the size of the instructions may be collected in the second multiplexer from the plurality of sources.

The second multiplexer may not collect instructions.

DESCRIPTION OF DRAWINGS

These and other features and advantages of the invention will become more apparent upon reading the following detailed description and upon reference to the accompanying drawings.

Figure 1 is a block diagram of a mobile video device utilizing a processor according to one embodiment of the present invention.

Figure 2 is a block diagram of a signal processing system according to an embodiment of the present invention.

Figure 3 is a block diagram of an alternative signal processing system according to an embodiment of the present invention.

Figure 4 illustrates exemplary pipeline stages of the processor in Figure 1 according to an embodiment of the present invention.

Figure 5 is a block diagram of a multiple source decoder feed system according to one embodiment of the present invention.

Figure 6 illustrates the process of providing a selected instruction from multiple sources to the decoder according to one embodiment of the present invention.

DETAILED DESCRIPTION

Figure 1 illustrates a mobile video device 100 including a processor according to an embodiment of the invention. The mobile video device 100 may be a hand-held device which displays video images produced from an encoded video signal received from an antenna 105 or a digital video storage medium 120, e.g., a digital video disc (DVD) or a memory card. A processor 110 communicates with a cache memory 115 which may store instructions and data for the processor operations. The processor 110 may be a microprocessor, a digital signal processor (DSP), a microprocessor controlling a slave DSP, or a processor with an hybrid microprocessor/DSP architecture. For the

purposes of this application, the processor 110 will be referred to hereinafter as a DSP 110.

The DSP 110 may perform various operations on the encoded video signal, including, for example, analog-to-digital conversion, demodulation, filtering, data recovery, and decoding. The DSP 110 may decode the compressed digital video signal according to one of various digital video compression standards such as the MPEG-family of standards and the H.263 standard. The decoded video signal may then be input to a display driver 130 to produce the video image on a display 125.

Hand-held devices generally have limited power supplies. Also, video decoding operations are computationally intensive. Accordingly, a processor for use in such a device is advantageously a relatively high speed, low power device.

The DSP 110 may have a deeply pipelined, load/store architecture. By employing pipelining, the performance of the DSP may be enhanced relative to a non-pipelined DSP. Instead of fetching a first instruction, executing the first instruction, and then fetching a second instruction, a pipelined DSP 110 fetches the second instruction concurrently with execution of the first

instruction, thereby improving instruction throughput. Further, the clock cycle of a pipelined DSP may be shorter than that of a non-pipelined DSP, in which the instruction must be fetched and executed in the same clock cycle.

Such a DSP 110 may be used with video camcorders, teleconferencing, PC video cards, and High-Definition Television (HDTV). In addition, the DSP 110 may also be used in connection with other technologies utilizing digital signal processing such as voice processing used in mobile telephony, speech recognition, and other applications.

Turning now to Figure 2, a block diagram of a signal processing system 200 including DSP 110 according to an embodiment is shown. One or more analog signals may be provided by an external source, e.g., antenna 105, to a signal conditioner 202. Signal conditioner 202 may perform certain preprocessing functions upon the analog signals. Exemplary preprocessing functions may include mixing several of the analog signals together, filtering, amplifying, etc. An analog-to-digital converter (ADC) 204 may be coupled to receive the preprocessed analog signals from signal conditioner 202 and to convert the preprocessed analog signals to digital signals consisting of samples, as

described above. The samples may be taken according to a sampling rate determined by the nature of the analog signals received by signal conditioner 202. The DSP 110 may be coupled to receive digital signals at the output of the ADC 204. The DSP 110 may perform the desired signal transformation upon the received digital signals, producing one or more output digital signals. A digital-to-analog converter (DAC) 206 may be coupled to receive the output digital signals from the DSP 110. The DAC 206 converts the output digital signals into output analog signals. The output analog signals may be then conveyed to another signal conditioner 208. The signal conditioner 208 performs post-processing functions upon the output analog signals. Exemplary post-processing functions are similar to the preprocessing functions listed above. It is noted that various alternatives of the signal conditioners 202 and 208, the ADC 204, and the DAC 206 are well known. Any suitable arrangement of these devices may be coupled into a signal processing system 200 with the DSP 110.

Turning next to Figure 3, a signal processing system 300 according to another embodiment is shown. In this embodiment, a digital receiver 302 may be arranged to receive one or more digital signals and to convey the

received digital signals to the DSP 110. As with the embodiment shown in Figure 2, DSP 110 may perform the desired signal transformation upon the received digital signals to produce one or more output digital signals. Coupled to receive the output digital signals is a digital signal transmitter 304. In one exemplary application, the signal processing system 300 is a digital audio device in which the digital receiver 302 conveys to the DSP 110 digital signals indicative of data stored on the digital storage device 120. The DSP 110 then processes the digital signals and conveys the resulting output digital signals to the digital transmitter 304. The digital transmitter 304 then causes values of the output digital signals to be transmitted to the display driver 130 to produce a video image on the display 125.

The pipeline illustrated in Figure 4 includes eight stages, which may include instruction fetch 402-403, decode 404, address calculation 405, execution 406-408, and write-back 409 stages. An instruction i may be fetched in one clock cycle and then operated on and executed in the pipeline in subsequent clock cycles concurrently with the fetching of new instructions, e.g., i+1 and i+2.

Pipelining may introduce additional coordination problems and hazards to processor performance. Jumps in the program flow may create empty slots, or "bubbles," in the pipeline. Situations which cause a conditional branch to be taken or an exception or interrupt to be generated may alter the sequential flow of instructions. After such an occurrence, a new instruction may be fetched outside of the sequential program flow, making the remaining instructions in the pipeline irrelevant. Methods such as data forwarding, branch prediction, and associating valid bits with instruction addresses in the pipeline may be employed to deal with these complexities.

Figure 5 is a block diagram of a multiple source decoder feed system 500 according to one embodiment of the present invention. The decoder feed system 500 may include a plurality of sources such as an Icache/alignment Unit 505 having an instruction register 507, a loop buffer 510, an emulation instruction register 515, and other sources 520, a 64-bit multiplexer (MUX) 525, a 2-bit multiplexer (MUX) 530, and a decoder 535. The decoder feed system 500 may allow the decoder 535 to be fed directly by one of the plurality of sources without having to transfer data to the instruction register 507. Because the data does not have

to be transferred to the instruction register 507 in this particular embodiment, the instruction latency may be reduced and the performance of the DSP 110 is increased. Further, the plurality of sources may provide instructions having the same format, including width bits. The design of the decoder 535 may be simplified by ensuring the plurality of sources provides similarly formatted instructions, thereby improving on cycle time.

The Icache/alignment unit 505, the loop buffer 510, the emulation instruction register 515, or any other source 520 may be connected to both the 64-bit MUX 525 and the 2-bit MUX 530. Each of these sources may be capable of providing instructions of multiple widths, such as 16-bit, 32-bit, or 64-bit instructions. These instructions are provided to the 64-bit MUX 525. Of course, other size MUXs capable of handling other size instructions may be used without departing from the spirit of the invention. Each of the sources also provides a signal to the 2-bit MUX 530 indicative of the width of the instruction provided to the 64-bit MUX 525. With a 2-bit signal, there are 4 possible values for the 2-bit width signal. For example, width bits of 00 indicates the instruction is invalid, width bits of 01 indicates a 16-bit instruction, width bits of 10

indicates a 32-bit instruction, and width bits of 11 indicates a 64-bit instruction. Once a particular instruction source is selected, both the instruction from the 64-bit MUX 525 and the width bits from the 2-bit MUX 530 from that source may be transferred to the decoder 535 for processing.

Although two multiplexers 525 and 530 are shown, it can be appreciated that any number of multiplexers may be used to permit selection of additional information. For example, an additional multiplexer may receive pre-decode information from each of the sources 505-520 and send the appropriate pre-decode information to the decoder.

The process 600 for providing instructions to the decoder 535 is shown in Figure 6. The process 600 begins at a start block 605. Proceeding to block 610, one or more of the sources provides instructions and corresponding width bits to the MUXs 525, 530. As stated above, the instructions may be a variety of sizes, including 16-bit, 32-bit, or 64-bit. Instructions may be provided by only one of the sources, but instructions and the corresponding width bits may also be provided by two or more of the sources.

Proceeding to block 615, the source to provide the instruction is selected. The DSP 110 may determine that the next instruction be provided by the Icache/alignment unit 505, the loop buffer 510, the emulation instruction register 515, or another source 520. After the DSP 110 determines the instruction to send to the decoder 535, the MUXs 525 and 530 provides the proper instruction and width bits to the decoder 535.

Proceeding to block 620, the selected instruction and width bits are transferred directly to the decoder 535 without being stored in the instruction register 507. By directly transferring the instructions to the decoder 535, the instruction latency may be lowered and the performance may be increased. The decoder 535 may then execute the instruction. The process then terminates in an end block 630.

The multiplexers 525, 530 of the present invention provide the proper information to the decoder 535 based on the selected instruction source 505-520. However, if multiple sources 505-520 are selected, the multiplexers 525, 530 may include priority logic to control the distribution of information to the decoder 535. For example, the multiplexers 525, 530 may include priority

logic stating that information from the emulation instruction register 515 has the highest priority, while information from the Icache/Alignment unit 505 is to be processed prior to information from the loop buffer 510. The priority schedule may be pre-determined or updated throughout processing.

Numerous variations and modifications of the invention will become readily apparent to those skilled in the art. Accordingly, the invention may be embodied in other specific forms without departing from its spirit or essential characteristics.

WHAT IS CLAIMED IS:

1. A method of providing instructions to a decoder within a processor comprising:

providing a selected instruction to a decoder from a plurality of sources.

2. The method of Claim 1, further comprising collecting a plurality of instructions from the plurality of sources.

3. The method of Claim 2, further comprising collecting the instructions in a multiplexer.

4. The method of Claim 1, further comprising transmitting width bits to the decoder corresponding to a size of the selected instruction.

5. The method of Claim 1, further comprising collecting a plurality of width bits corresponding to the size of the instructions from the plurality of sources.

6. The method of Claim 5, further comprising collecting the plurality of width bits in a second multiplexer.

7. The method of Claim 1, further comprising determining the selected instruction based on a priority schedule.

8. The method of Claim 1, further comprising providing the instructions in a digital signal processor.

9. A method of obtaining instructions at a decoder within a processor comprising:

collecting a plurality of instructions from a plurality of instruction sources;

collecting a plurality of width bits corresponding to the plurality of instructions

determining one of the plurality of instructions to be processed; and

transferring the selected one of the plurality of instructions and the corresponding width bit to the decoder.

10. The method of Claim 9, further comprising collecting instructions from an Icache/Alignment unit.

11. The method of Claim 9, further comprising collecting instructions from a loop buffer.

12. The method of Claim 9, further comprising collecting instructions from an emulation instruction register.

13. The method of Claim 9, further comprising obtaining instructions in a digital signal processor.

14. The method of Claim 9, further comprising collecting the plurality of instructions having the same format.

15. The method of Claim 9, further comprising determining the instruction to be processed based on a priority schedule.

16. A processor comprising:

a plurality of instruction sources which provide a plurality of instructions;

a first multiplexer which receives the plurality of instructions from the plurality of instruction sources,

wherein the first multiplexer selects a first instruction to be executed from the plurality of instructions; and

a decoder which receives the selected first instruction from the first multiplexer.

17. The processor of Claim 16, further comprising a second multiplexer which receives a plurality of signals from the instruction sources indicative of the size of the instructions, wherein the second multiplexer provides to the decoder one of the plurality of signals which corresponds to the selected first instruction.

18. The processor of Claim 16, wherein each of the plurality of instructions is in the same format.

19. The processor of Claim 16, wherein the decoder receives the selected first instruction directly from the multiplexer.

20. The processor of Claim 16, wherein the first multiplexer selects a first instruction to be executed from the plurality of instructions based on a priority schedule.

21. The processor of Claim 16, wherein the processor is a digital signal processor.

22. An apparatus, including instructions residing on a machine-readable storage medium, for use in a machine system to handle a plurality of instructions, the instructions causing the machine to:

select an instruction from a plurality of sources; and

provide the selected instruction to the decoder.

23. The apparatus of Claim 22, wherein the machine collects a plurality of instructions from the plurality of sources.

24. The apparatus of Claim 23, wherein the instructions are collected in a multiplexer.

25. The apparatus of Claim 22, wherein a plurality of width bits corresponding to the size of the instructions are collected from the plurality of sources.

26. A method of providing instructions to a decoder within a processor comprising:

providing a selected instruction and a second instruction to a decoder from a plurality of sources, wherein the selected instruction does not pass through an instruction cache.

27. The method of Claim 26, further comprising collecting a plurality of instructions from the plurality of sources.

28. The method of Claim 27, further comprising collecting the instructions in a multiplexer.

29. The method of Claim 26, further comprising transmitting width bits to the decoder corresponding to a size of the selected instruction.

30. The method of Claim 26, further comprising collecting a plurality of width bits corresponding to the size of the instructions from the plurality of sources.

31. The method of Claim 30, further comprising collecting the plurality of width bits in a second multiplexer.

32. The method of Claim 26, further comprising determining the selected instruction based on a priority schedule.

33. The method of Claim 26, further comprising providing the instructions in a digital signal processor.

34. A method of obtaining instructions at a decoder within a processor comprising:

collecting a plurality of instructions from a plurality of instruction sources, wherein said collecting a plurality of instructions includes collecting a first instruction, wherein the first instruction does not pass through an instruction cache;

collecting a plurality of width bits corresponding to the plurality of instructions

determining one of the plurality of instructions to be processed; and

transferring the selected one of the plurality of instructions and the corresponding width bit to the decoder.

35. The method of Claim 34, further comprising collecting instructions from an Icache/Alignment unit.

36. The method of Claim 34, further comprising collecting instructions from a loop buffer.

37. The method of Claim 34, further comprising collecting instructions from an emulation instruction register.

38. The method of Claim 34, further comprising obtaining instructions in a digital signal processor.

39. The method of Claim 34, further comprising collecting the plurality of instructions having the same format.

40. The method of Claim 34, further comprising determining the instruction to be processed based on a priority schedule.

41. A processor comprising:

a plurality of instruction sources which provide a plurality of instructions;

a first multiplexer which receives the plurality of instructions from the plurality of instruction sources, wherein the first multiplexer selects a first instruction to be executed from the plurality of instructions, and wherein the first instruction has not passed through an instruction cache; and

a decoder which receives the selected first instruction from the first multiplexer.

42. The processor of Claim 41, further comprising a second multiplexer which receives a plurality of signals from the instruction sources indicative of the size of the instructions, wherein the second multiplexer provides to the decoder one of the plurality of signals which corresponds to the selected first instruction.

43. The processor of Claim 41, wherein each of the plurality of instructions is in the same format.

44. The processor of Claim 41, wherein the decoder receives the selected first instruction directly from the multiplexer.

45. The processor of Claim 41, wherein the first multiplexer selects a first instruction to be executed from the plurality of instructions based on a priority schedule.

46. The processor of Claim 41, wherein the processor is a digital signal processor.

47. An apparatus, including instructions residing on a machine-readable storage medium, for use in a machine

system to handle a plurality of instructions, the instructions causing the machine to:

select an instruction from a plurality of sources; and

provide the selected instruction to the decoder,

wherein the selected instruction has not passed through an instruction cache.

48. The apparatus of Claim 47, wherein the machine collects a plurality of instructions from the plurality of sources.

49. The apparatus of Claim 48, wherein the instructions are collected in a multiplexer.

50. The apparatus of Claim 47, wherein a plurality of width bits corresponding to the size of the instructions are collected from the plurality of sources.

51. A method of providing instructions to a decoder within a processor comprising:

collecting in a first multiplexer a selected instruction from a plurality of sources;

collecting in a second multiplexer a width signal from the one of the plurality of sources, the width signal specifying the size of the selected instruction; and

providing the selected instruction and the width signal to the decoder.

52. The method of Claim 51, further comprising collecting a plurality of instructions from the plurality of sources.

53. The method of Claim 52, further comprising collecting a plurality of width bits corresponding to the size of the plurality of instructions from the plurality of sources.

54. The method of Claim 51, further comprising transmitting width bits to the decoder corresponding to a size of the selected instruction.

55. The method of Claim 51, further comprising collecting the plurality of width bits in the second multiplexer.

56. The method of Claim 51, further comprising determining the selected instruction based on a priority schedule.

57. The method of Claim 51, further comprising providing the instructions in a digital signal processor.

58. The method of claim 51, wherein the second multiplexer does not collect instructions.

59. A method of obtaining instructions at a decoder within a processor comprising:

collecting in a first multiplexer a plurality of instructions from a plurality of instruction sources;

collecting in a second multiplexer a plurality of width bits corresponding to the plurality of instructions;

determining one of the plurality of instructions to be processed; and

transferring the selected one of the plurality of instructions and the corresponding width bit to the decoder.

60. The method of Claim 59, further comprising collecting instructions from an Icache/Alignment unit.

61. The method of Claim 59, further comprising collecting instructions from a loop buffer.

62. The method of Claim 59, further comprising collecting instructions from an emulation instruction register.

63. The method of Claim 59, further comprising obtaining instructions in a digital signal processor.

64. The method of Claim 59, further comprising collecting the plurality of instructions having the same format.

65. The method of Claim 59, further comprising determining the instruction to be processed based on a priority schedule.

66. The method of claim 59, wherein the second multiplexer does not collect instructions.

67. A processor comprising:

a plurality of instruction sources which provide a plurality of instructions;

a first multiplexer which receives the plurality of instructions from the plurality of instruction sources, wherein the first multiplexer selects a first instruction to be executed from the plurality of instructions;

a decoder which receives the selected first instruction from the first multiplexer; and

a second multiplexer which receives a plurality of signals from the instruction sources indicative of the size of the instructions, wherein the second multiplexer provides to the decoder one of the plurality of signals which corresponds to the selected first instruction.

68. The processor of Claim 67, wherein the second multiplexer does not receive instructions.

69. The processor of Claim 67, wherein each of the plurality of instructions is in the same format.

70. The processor of Claim 67, wherein the decoder receives the selected first instruction directly from the multiplexer.

71. The processor of Claim 67, wherein the first multiplexer selects a first instruction to be executed from the plurality of instructions based on a priority schedule.

72. The processor of Claim 67, wherein the processor is a digital signal processor.

73. The processor of claim 67, wherein the first multiplexer is coupled to each of the plurality of instruction sources, wherein the second multiplexer is separately coupled to each of the plurality of instruction sources, and wherein the first multiplexer and the second multiplexer are coupled in parallel to the decoder.

74. An apparatus, including instructions residing on a machine-readable storage medium, for use in a machine system to handle a plurality of instructions, the instructions causing the machine to:

collect in a first multiplexer a selected instruction from a plurality of sources;

collect in a second multiplexer a width signal from the one of the plurality of sources, the width signal specifying the size of the selected instruction; and

provide the selected instruction and the width signal to a decoder.

75. The apparatus of Claim 74, wherein the machine collects in the first multiplexer a plurality of instructions from the plurality of sources.

76. The apparatus of Claim 75, wherein a plurality of width bits corresponding to the size of the instructions are collected in the second multiplexer from the plurality of sources.

77. The apparatus of claim 74, wherein the second multiplexer does not collect instructions.