What is claimed is:

1. A method including:

receiving information representative of a graph-based program specification having a plurality of components, each of which corresponds to a task, and directed links between ports of said components, and

executing a program corresponding to said graph-based program specification,

wherein a first component of the graph-based program specification includes a first data port, a first control port, and a second control port,

wherein said first data port is configured to receive data to be processed by a first task corresponding to said first component, or configured to provide data that was processed by said first task corresponding to said first component,

wherein said first control port is configured to receive first control information that includes information that triggers invoking said first task or triggers suppression of processing by said first task,

wherein said second control port is configured to provide second control information and is connected to a first control port of a second component, and

wherein executing a program corresponding to said graph-based program specification includes:

receiving said first control information at said first control port,

in response to receiving said first control information, determining whether or not to invoke said first task,

causing said first component to, (1) if said first control information includes information that triggers invoking said first task, trigger processing by a second task corresponding to said second component, wherein causing said first component to trigger processing by said second task includes said first component placing a positive control message on said second control port of said first component, and said second component receiving said positive control message on said first control port of said second component, and, (2) if said first control information includes information that triggers suppression of processing by said first task, suppress processing by said second task corresponding to said second component, wherein causing said first component to suppress processing by said

second task includes said first component placing a negative control message on said second control port of said first component, and said second component receiving said negative control message on said first control port of said second component, and

after receiving said first control information, providing said second control information from said second control port.

2. The method of claim 1, wherein said first control information includes information that triggers invoking said first task.

3. The method of claim 2, wherein invocation of said first task causes processing of data received at said first data port.

4. The method of claim 2, wherein invocation of said first task causes processing of data provided from said first data port.

5. The method of claim 2, wherein providing said second control information from said second control port includes computing said second control information based at least in part on a result of said first task.

6. The method of claim 2, wherein invocation of said first task causes allocation of computational resources to said task.

7. The method of claim 2, wherein invocation of said first task causes said task to transition from consuming no computational resources to consuming computational resources.

8. The method of claim 1, wherein said first control information includes information that triggers suppression of processing by said first task.

9. The method of claim 8, wherein providing said second control information from said second control port includes propagating said information that triggers suppression of processing by said first task to a component having a control port connected to said second control port.

10. The method of claim 8, including causing said first component to suppress processing by a second task corresponding to said second component.

11. The method of claim 10, wherein causing said first component to suppress processing by said second task includes said first component placing a negative control message on said second control port of said first component, and said second component receiving said negative control message on said first control port of said second component.

12. The method of claim 2, including causing said first component to trigger processing by a second task corresponding to said second component.

13. The method of claim 12, wherein causing said first component to trigger processing by said second task includes said first component placing a positive control message on said second control port of said first component, and said second component receiving said positive control message on said first control port of said second component.

14. The method of claim 1, further including providing logic including a first input, a second input, and an output, and connecting said output to said first control port of said second component.

15. The method of claim 14, further including connecting said first input to said second control port of said first component.

16. The method of claim 15, further including connecting said second input to a control port of a third component.

17. Software stored in a non-transitory form on a computer-readable medium, the software including instructions for causing a computing system to:

receive information representative of a graph-based program specification having a plurality of components, each of which corresponds to a task, and directed links between ports of said components, and

execute a program corresponding to said graph-based program specification,

wherein a first component of the graph-based program specification includes a first data port, a first control port, and a second control port,

wherein said first data port is configured to receive data to be processed by a first task corresponding to said first component, or configured to provide data that was processed by said first task corresponding to said first component,

wherein said first control port is configured to receive first control information that includes information that triggers invoking said first task or triggers suppression of processing by said first task,

wherein said second control port is configured to provide second control information and is connected to a first control port of a second component, and

wherein executing a program corresponding to said graph-based program specification includes:

receiving said first control information at said first control port,

in response to receiving said first control information, determining whether or not to invoke said first task,

causing said first component to, (1) if said first control information includes information that triggers invoking said first task, trigger processing by a second task corresponding to said second component, wherein causing said first component to trigger processing by said second task includes said first component placing a positive control message on said second control port of said first component, and said second component receiving said positive control message on said first control port of said second component, and, (2) if said first control information includes information that triggers suppression of processing by said first task, suppress processing by said second task corresponding to said second component, wherein causing said first component to suppress processing by said second task includes said first component placing a negative control message on said second control port of said first component, and said second component receiving said negative control message on said first control port of said second component, and

after receiving said first control information, providing said second control information from said second control port.

18. A computing system including:

an input device or port configured to receive information representative of a graph-based program specification having a plurality of components, each of which corresponds to a task, and directed links between ports of said components, and

at least one processor configured to execute a program corresponding to said graph-based program specification,

wherein a first component of the graph-based program specification includes a first data port, a first control port, and a second control port,

wherein said first data port is configured to receive data to be processed by a first task corresponding to said first component, or configured to provide data that was processed by said first task corresponding to said first component,

wherein said first control port is configured to receive first control information that includes information that triggers invoking said first task or triggers suppression of processing by said first task,

wherein said second control port is configured to provide second control information and is connected to a first control port of a second component, and

wherein executing a program corresponding to said graph-based program specification includes:

receiving said first control information at said first control port,

in response to receiving said first control information, determining whether or not to invoke said first task,

causing said first component to, (1) if said first control information includes information that triggers invoking said first task, trigger processing by a second task corresponding to said second component, wherein causing said first component to trigger processing by said second task includes said first component placing a positive control message on said second control port of said first component, and said second component receiving said positive control message on said first control port of said second component, and, (2) if said first control information includes information that triggers suppression of processing by said first task, suppress processing by said second task corresponding to said second component, wherein causing said first component to suppress processing by said second task includes said first component placing a negative control message on

said second control port of said first component, and said second component receiving said negative control message on said first control port of said second component, and

after receiving said first control information, providing said second control information from said second control port.

19. A computing system including:

means for receiving information representative of a graph-based program specification having a plurality of components, each of which corresponds to a task, and directed links between ports of said components, and

means for executing a program corresponding to said graph-based program specification,

wherein a first component of the graph-based program specification includes a first data port, a first control port, and a second control port,

wherein said first data port is configured to receive data to be processed by a first task corresponding to said first component, or configured to provide data that was processed by said first task corresponding to said first component,

wherein said first control port is configured to receive first control information that includes information that triggers invoking said first task or triggers suppression of processing by said first task,

wherein said second control port is configured to provide second control information and is connected to a first control port of a second component, and

wherein executing a program corresponding to said graph-based program specification includes:

receiving said first control information at said first control port,

in response to receiving said first control information, determining whether or not to invoke said first task,

causing said first component to, (1) if said first control information includes information that triggers invoking said first task, trigger processing by a second task corresponding to said second component, wherein causing said first component to trigger processing by said second task includes said first component placing a positive control message on said second control port of said first component, and

said second component receiving said positive control message on said first control port of said second component, and, (2) if said first control information includes information that triggers suppression of processing by said first task, suppress processing by said second task corresponding to said second component, wherein causing said first component to suppress processing by said second task includes said first component placing a negative control message on said second control port of said first component, and said second component receiving said negative control message on said first control port of said second component, and

after receiving said first control information, providing said second control information from said second control port.

20. The method of claim 1, wherein said second control port of said first component is connected to an input port of said second component, such that said second control information is provided to the input port of the second component; and said first control port of said first component is connected to an output port of a third component of the graph based program specification, such that said first control information is provided to said first control port of said first component from the output port of the third component.

21. The method of claim 20, wherein said second control information is determined by said first component.

22. The method of claim 21, wherein said second control information is determined based at least in part on said first control information.

23. The software of claim 17, wherein said first control information includes information that triggers invoking said first task.

24. The software of claim 23, wherein invocation of said first task causes processing of data received at said first data port.

25. The software of claim 23, wherein invocation of said first task causes processing of data provided from said first data port.

26. The software of claim 23, wherein providing said second control information from said second control port includes computing said second control information based at least in part on a result of said first task.

27. The software of claim 23, wherein invocation of said first task causes allocation of computational resources to said task.

28. The software of claim 23, wherein invocation of said first task causes said task to transition from consuming no computational resources to consuming computational resources.

29. The software of claim 17, wherein said first control information includes information that triggers suppression of processing by said first task.

30. The software of claim 29, wherein providing said second control information from said second control port includes propagating said information that triggers suppression of processing by said first task to a component having a control port connected to said second control port.

31. The software of claim 17, wherein said second control port of said first component is connected to an input port of said second component, such that said second control information is provided to the input port of the second component; and said first control port of said first component is connected to an output port of a third component of the graph based program specification, such that said first control information is provided to said first control port of said first component from the output port of the third component.

32. The software of claim 31, wherein said second control information is determined by said first component.

33. The software of claim 32, wherein said second control information is determined based at least in part on said first control information.

34. The computing system of claim 18, wherein said first control information includes information that triggers invoking said first task.

35. The computing system of claim 34, wherein invocation of said first task causes processing of data received at said first data port.

36. The computing system of claim 34, wherein invocation of said first task causes processing of data provided from said first data port.

37. The computing system of claim 34, wherein providing said second control information from said second control port includes computing said second control information based at least in part on a result of said first task.

38. The computing system of claim 34, wherein invocation of said first task causes allocation of computational resources to said task.

39. The computing system of claim 34, wherein invocation of said first task causes said task to transition from consuming no computational resources to consuming computational resources.

40. The computing system of claim 18, wherein said first control information includes information that triggers suppression of processing by said first task.

41. The computing system of claim 40, wherein providing said second control information from said second control port includes propagating said information that triggers suppression of processing by said first task to a component having a control port connected to said second control port.

42. The computing system of claim 18, wherein said second control port of said first component is connected to an input port of said second component, such that said second control information is provided to the input port of the second component; and said first control port of said first component is connected to an output port of a third component of the graph based program specification, such that said first control information is provided to said first control port of said first component from the output port of the third component.

43. The computing system of claim 42, wherein said second control information is determined by said first component.

44. The computing system of claim 43, wherein said second control information is determined based at least in part on said first control information.