13. The touch pad assembly as recited in claim 1 wherein the new value of the logical device units implements a specific control function in the host device.

14. The touch pad assembly as recited in claim 1 wherein the logical device units are angular Polar units distributed around the surface of the touch pad in a clock like manner.

15. The touch pad assembly as recited in claim 1 wherein the ratio of native sensor coordinates to logical device units is between about 1024:1 to about 8:1.

16. The touch pad assembly as recited in claim 1 further comprising one or more touch buttons having one or more sensors, and wherein the controller receives a native value from the sensors, determines a button status from the native value, and reports the button status to a host device, the button status being used by the host device to implement a button function in the host device.

17. The touch pad assembly as recited in claim 16 wherein the controller only reports the button status to the host device when it is determined that there is a change in button status.

18. The touch pad assembly as recited in claim 1 wherein each of the logical device units represent a different movement direction on a display screen of the host device so as to enable joystick implementations, multiple dimensional menu selection or photo image panning.

19. The touch pad assembly as recited in claim 1 wherein the host device is a media player for storing and playing media such as audio, video or images, the media player including a housing that supports the touch pad assembly, a display for displaying text and graphics to a user of the media player and a CPU capable of receiving the new value of the logical device units from the controller and issuing commands based on the new value of logical device units to other components of the media player, the commands being used to at least move an object on the display.

20. A method for a touch pad, comprising: IV

mapping the touch pad into native sensor coordinates;

producing native values of the native sensor coordinates when events occur on the touch pad;

filtering the native values of the native sensor coordinates based on the type of events that occur on the touch pad;

generating a control signal based on the native values of the native sensor coordinates when a desired event occurs on the touch pad.

21. The method as recited in claim 20 wherein the control signal includes the native values of the native sensor coordinates.

22. The method as recited in claim 20 further comprising:

adjusting the native values of the native sensor coordinates into a new value when a desired event occurs on the touch pad, the control signal including the new value.

23. The method as recited in claim 20 wherein the new value has the same units as the native values.

24. The method as recited in claim 20 wherein the new value has different units as the native values.

25. The method as recited in claim 20 wherein the step of filtering comprises:

determining if the native values are caused by noise events or actual events; and

filtering out noise events and passing actual events.

26. The method as recited in claim 25 wherein the step of determining comprises:

comparing a current set of native values with a last set of native values;

classifying the current set of native values as noise events when the current set of native values is substantially similar to the previous set of native values; and

classifying the current set of native values as actual events when the current set of native values is significantly different than the previous set of native values.

27. The method as recited in claim 25 wherein the control signal includes native values of the native sensor coordinates if it is determined that the events are actual events.

28. The method as recited in claim 25 further comprising:

adjusting the native values of the native sensor coordinates into a new value if it is determined that the events are actual events, and including the new value in the control signal.

29. A signal processing method for a controller of a touch pad, comprising:

receiving a current user location;

determining the difference in user location by comparing the current user location to a last user location;

only outputting the current user location when the difference in user location is larger than a threshold value;

converting the outputted current user location into a logical device unit; and

generating a message for a host device, the message including the more logical user location, the more logical user location being used by the host device to move a control object in a specified manner.

30. The method as recited in claim 29 wherein the threshold value is defined as the number of sensor levels that need to change in the touch pad in order to report a change in the user location.

31. The method as recited in claim 30 wherein the threshold is determined by the following equation:

Threshold(T)=C*(native sensor resolution of the touch pad/logical device resolution of the touch pad),

where the native sensor resolution defines the maximum number of different user locations that the sensors of the touch pad are able to detect over the touch pad plane, the logical device resolution defines the number of logical device units that the touch pad reports to the host device, and C defines the width border area between clusters of sensors of the touch pad that define one logical device unit.

32. The method as recited in claim 31 wherein the coefficient C is a value between about 0 and 0.5.

33. The method as recited in claim 31 wherein the native sensor resolution is about 1024 and the logical device resolution is about 128.