US 20060066588A1

(19) **United States**

(12) **Patent Application Publication**

Lyon et al.

(10) **Pub. No.: US 2006/0066588 A1**

(43) **Pub. Date: Mar. 30, 2006**

(54) **SYSTEM AND METHOD FOR PROCESSING RAW DATA OF TRACK PAD DEVICE**

(75) Inventors: **Benjamin Lyon**, San Jose, CA (US); **Stephanie Cinereski**, Sunnyvale, CA (US); **Chad Bronstein**, San Francisco, CA (US); **Steven Hotelling**, San Jose, CA (US)

Correspondence Address:
WONG, CABELLO, LUTSCH, RUTHERFORD & BRUCCULERI,
P.C.
20333 SH 249
SUITE 600
HOUSTON, TX 77070 (US)

(73) Assignee: **Apple Computer, Inc.**, Cupertino, CA

(21) Appl. No.: **11/232,299**

(22) Filed: **Sep. 21, 2005**

Related U.S. Application Data

(63) Continuation-in-part of application No. 10/949,060, filed on Sep. 24, 2004.

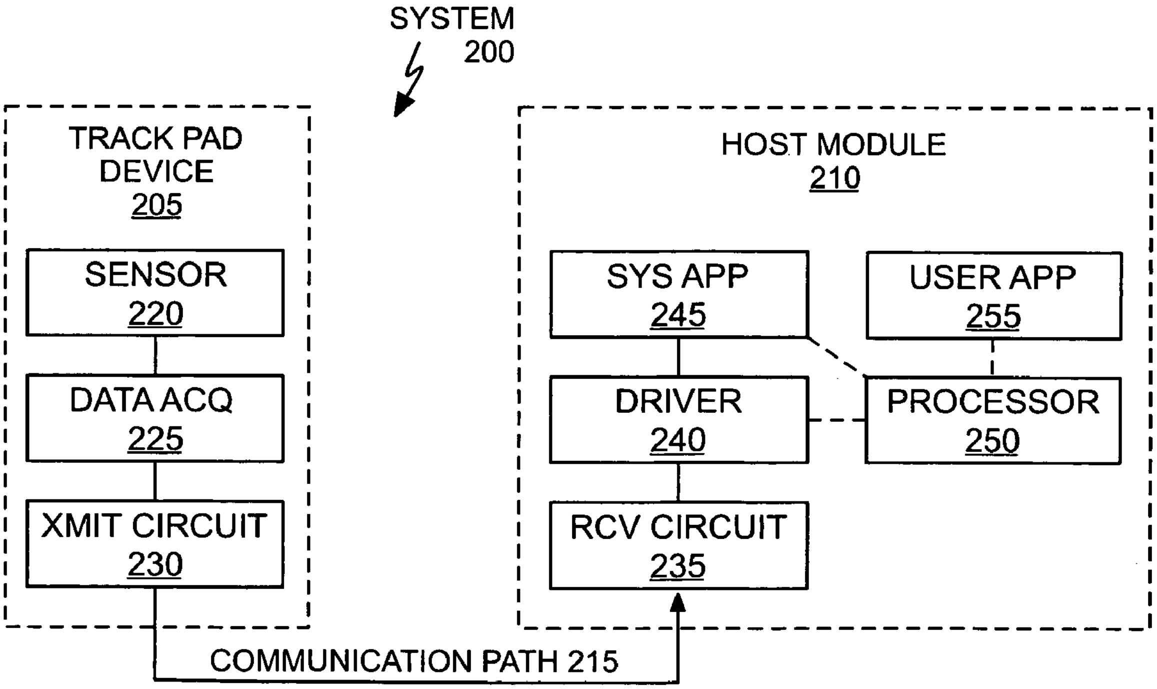
Publication Classification

(51) **Int. Cl.**
G09G 5/00 (2006.01)

(52) **U.S. Cl.** **345/173**

(57) **ABSTRACT**

An input device and system are described that acquires (measures) raw track pad sensor data and transmits this data to a host computer where it is analyzed by an application executing on one or more host computer central processing units. The resulting input processing architecture provides a track pad input device that is both lower in cost to manufacture and more flexible than prior art track pad input devices. Lower costs may be realized by eliminating the prior art's dedicated track pad hardware for processing sensor data (e.g., a processor and associated firmware memory). Increased flexibility may be realized by providing feature set functionality via software that executes on the host computer. In this architecture, track pad functionality may be modified, updated, and enhanced through software upgrade procedures.



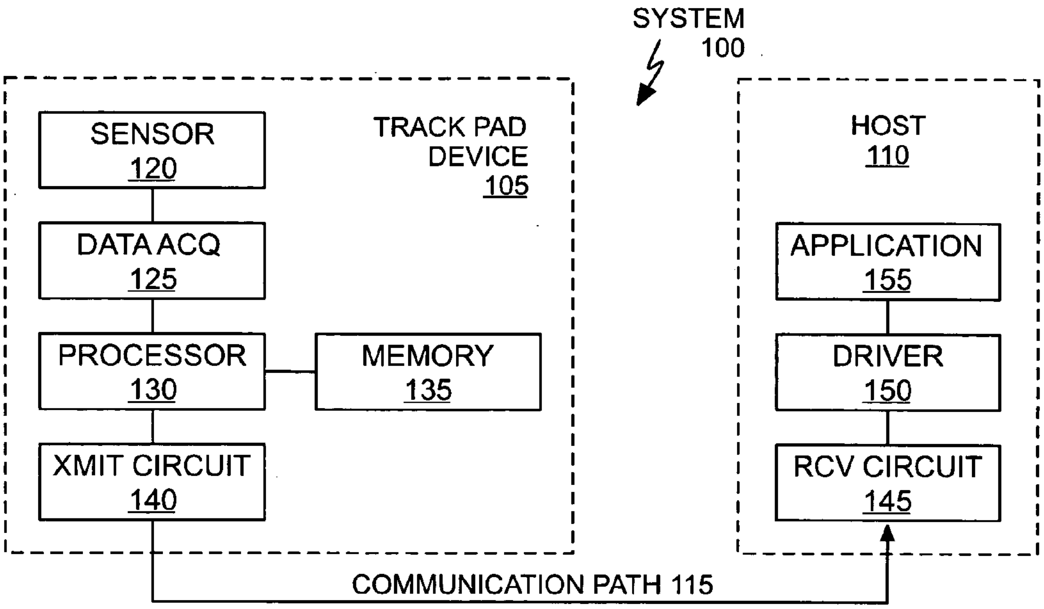


FIG. 1
(Prior Art)

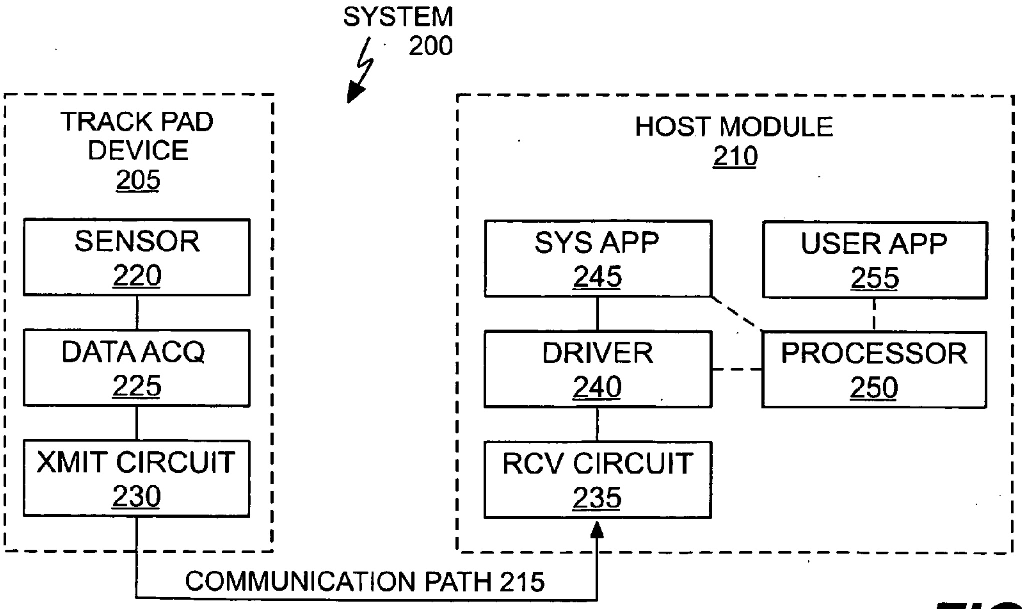


FIG. 2

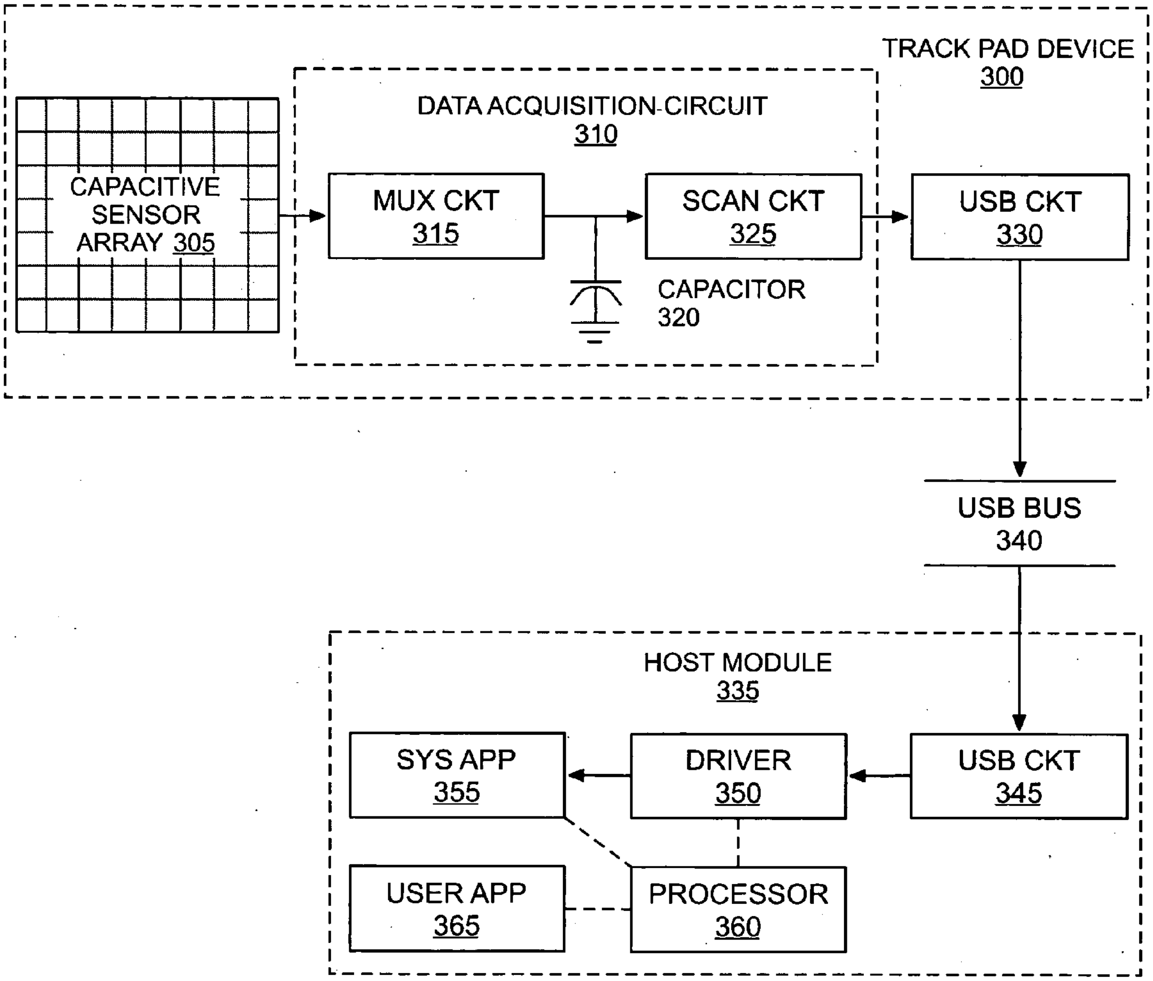

TRACK PAD DEVICE
300
CAPACITIVE
SENSOR
ARRAY 305
DATA ACQUISITION CIRCUIT
310
MUX CKT
315
SCAN CKT
325
CAPACITOR
320
USB CKT
330
USB BUS
340
HOST MODULE
335
SYS APP
355
DRIVER
350
USB CKT
345
USER APP
365
PROCESSOR
360

FIG. 3

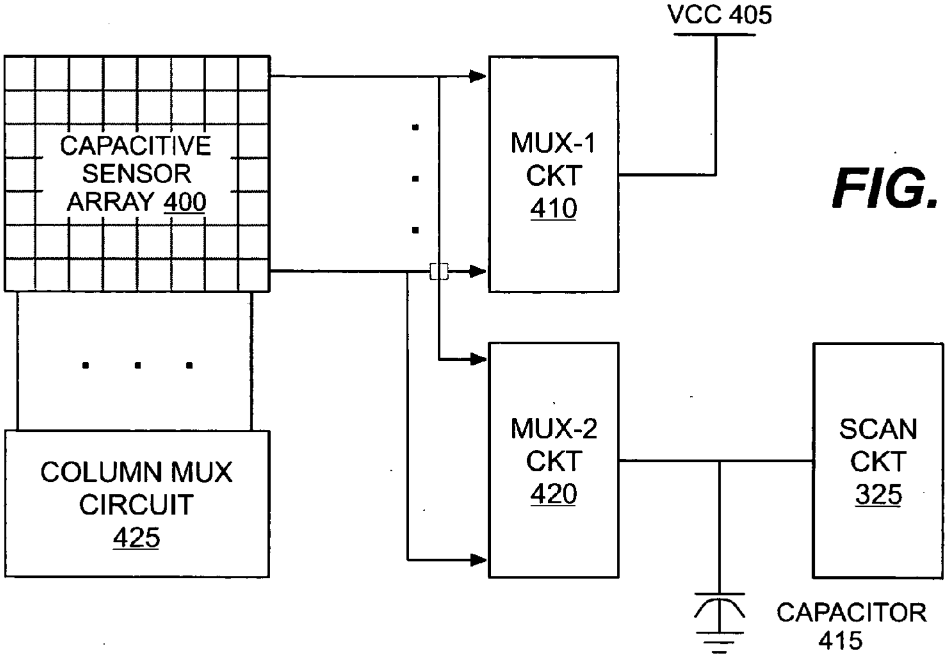

VCC 405
CAPACITIVE SENSOR ARRAY 400
MUX-1 CKT 410
MUX-2 CKT 420
SCAN CKT 325
COLUMN MUX CIRCUIT 425
CAPACITOR 415

FIG. 4

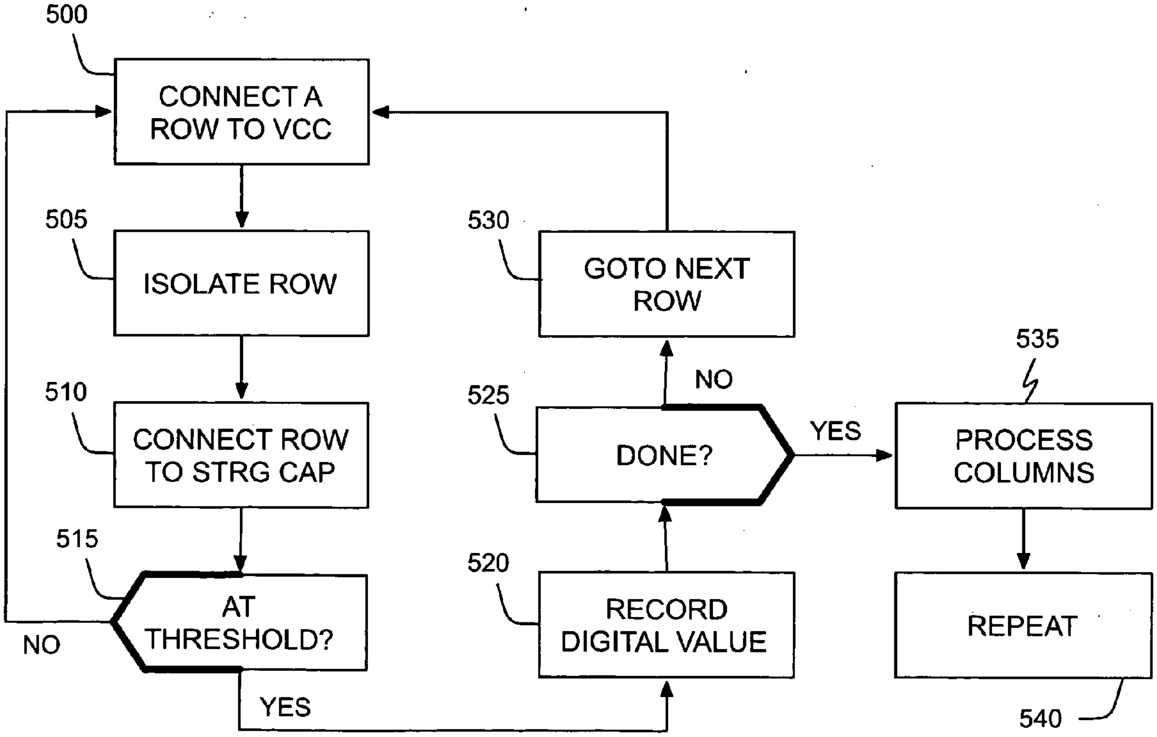

500
CONNECT A ROW TO VCC
505
ISOLATE ROW
510
CONNECT ROW TO STRG CAP
515
AT THRESHOLD?
NO
YES
520
RECORD DIGITAL VALUE
525
DONE?
NO
YES
530
GOTO NEXT ROW
535
PROCESS COLUMNS
REPEAT
540

FIG. 5

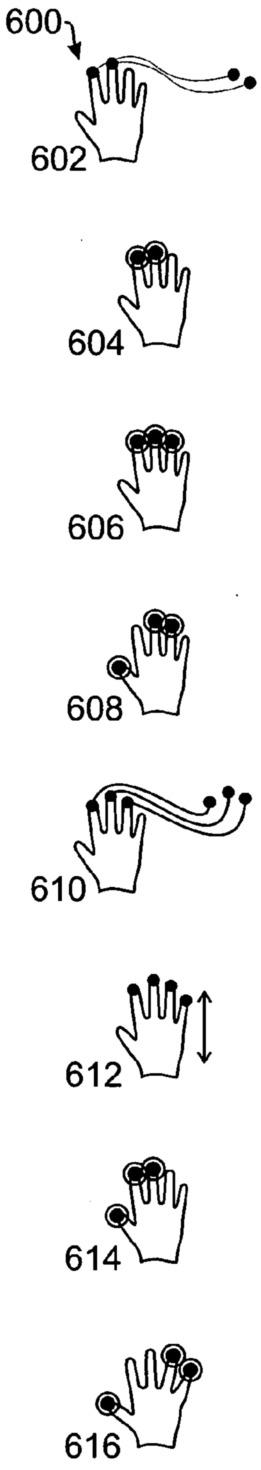


FIG. 6

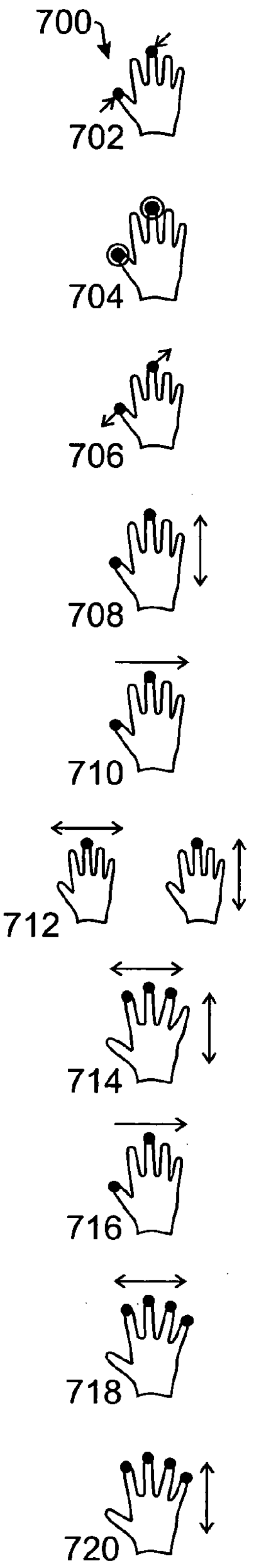


FIG. 7

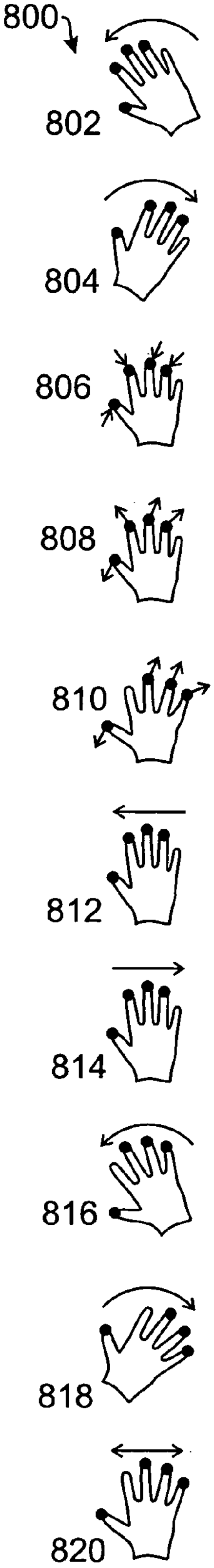


FIG. 8

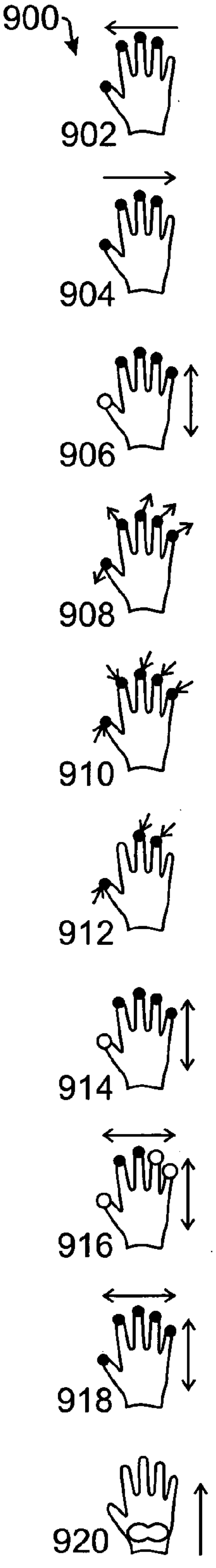


FIG. 9

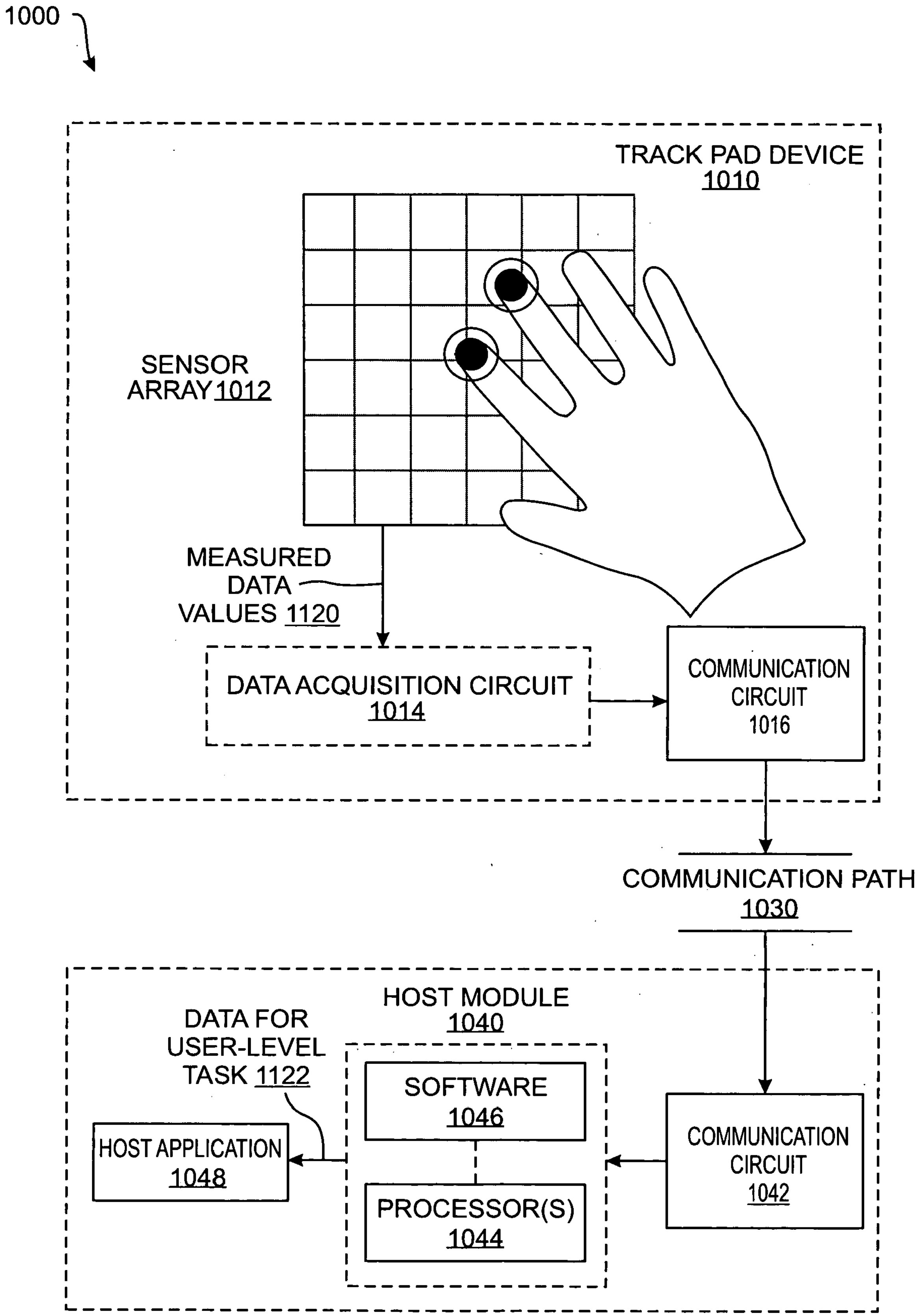

1000
TRACK PAD DEVICE
1010
SENSOR
ARRAY 1012
MEASURED
DATA
VALUES 1120
DATA ACQUISITION CIRCUIT
1014
COMMUNICATION
CIRCUIT
1016
COMMUNICATION PATH
1030
HOST MODULE
1040
DATA FOR
USER-LEVEL
TASK 1122
SOFTWARE
1046
PROCESSOR(S)
1044
HOST APPLICATION
1048
COMMUNICATION
CIRCUIT
1042

FIG. 10

SYSTEM AND METHOD FOR PROCESSING RAW DATA OF TRACK PAD DEVICE

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation-in-part of co-pending U.S. patent application Ser. No. 10/949,060 filed Sep. 24, 2004, which is incorporated herein by reference in its entirety and to which priority is claimed.

BACKGROUND

[0002] The invention relates generally to computer input devices and more particularly to a track pad input device that generates and transmits measured (raw) sensor data to a host computer system. Software executing on the host computer system analyzes the raw sensor data to determine the user's action.

[0003] A track pad is a touch-sensing planar digitizer input device used instead of, or in conjunction with, a mouse or trackball. During use, an operator places a finger on the track pad and moves the finger along the touch-sensing planar surface. The track pad detects the movement of the finger and in response provides location and/or motion signals to a computer. There are two common types of track pad sensor devices: resistive and capacitive. A resistive track pad sensor is a mechanical sensor that uses two layers of material that are typically separated by air. Pressure from a finger pushes the top layer (generally a thin, clear polyester film) so that it touches the bottom layer (generally glass). The voltage at the contact point is measured and the finger's location and/or motion is computed and transmitted to a host computer system. After the finger is removed, the top layer "bounces back" to its original configuration. A capacitive track or touch pad sensor, in contrast, is a solid-state sensor made using printed circuit board ("PCB") or flex circuit technology. A finger on, or in close proximity to, a top grid of conductive traces changes the capacitive coupling between adjacent traces or the self-capacitance of each trace. This change in capacitance is measured and the finger's location and/or motion is computed and transmitted to a host computer system.

[0004] Referring to **FIG. 1**, prior art computer system **100** includes track pad device **105** coupled to host computer module **110** via communication path **115**. Track pad device **105** comprises sensor **120**, data acquisition circuit **125**, processor **130**, memory **135** and transmit circuit **140**. In the case of a capacitive track pad device, as a user's finger(s) is (are) moved over the surface of sensor **120**, data acquisition circuit **125** measures changes in the capacitive coupling between adjacent sensor elements (or the self-capacitance of a given sensor element). Processor **130**, in conjunction with memory **135**, processes the acquired capacitance signals to compute a signal indicating the user's finger position on sensor **120** (e.g., a Δx and Δy signal). In some prior art track pad devices, processor **130** may also determine if multiple fingers are activating sensor **120** and whether certain pre-determined finger motions (often referred to as "gestures") are being made—e.g., "select,""drag,""file open" and "file close" operations. At specified intervals (e.g., 50 times per second), the user's finger location and/or motion as determined by processor **130** is transmitted to host computer module **110** via communication path **115**. At host computer module **110**, receive circuit **145** receives the transmitted track pad signal and passes its information to driver application **150**. Driver application **150**, in turn, makes the computed sensor information available to other applications such as, for example, window display subsystem application **155**. Thus, prior art system **100** utilizes a dedicated processor for measuring and analyzing raw track pad sensor data to generate a signal that indicates a user's action.

[0005] One of ordinary skill in the art will recognize that processor **130** may be embodied in a general-purpose processor (e.g., a microprocessor), a microcontroller, or a special purpose or custom designed processor or state machine (e.g., an application specific integrated circuit or a custom designed gate array device). Further, memory **135** is typically used to provide permanent storage for instructions (i.e., firmware) to drive processor **130** and may, optionally, include random access memory and/or register storage. A benefit of the architecture of **FIG. 1** is that host computer module **110** does not need to know about or understand the type of data generated by sensor **120**. A corollary of this feature is that host computer module **110** does not process track pad sensor data.

[0006] It will also be recognized by one of ordinary skill that a drawback to the architecture of **FIG. 1** is that the feature set (i.e., what motions are detectable) provided by track pad device **105** is essentially fixed by its dedicated hardware—processor **130** and associated firmware (memory **135**). Another drawback to the architecture of **FIG. 1** is that each manufactured device **105** includes the cost of processor **130** and associated firmware memory **135**. Thus, it would be beneficial to provide a track pad device that overcomes these inherent drawbacks.

SUMMARY

[0007] In one embodiment the invention provides a track pad input device that includes a track pad sensor element that generates output signals representing a track pad sensor characteristic (i.e., capacitance or resistance), a data acquisition circuit that measures a (digital) value encoding the track pad sensor's characteristic and a communication circuit that transmits the measured track pad sensor values to a general purpose processor for analysis, the general purpose processor is also responsible for executing user and other system level tasks or applications. In one specific embodiment, the track pad sensor is a capacitive track pad sensor so that measured values comprise raw track pad sensor values and the general-purpose processor corresponds to a host computer system's central processing unit.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] **FIG. 1** shows, in block diagram form, a track pad-computer system architecture in accordance with the prior art.

[0009] **FIG. 2** shows, in block diagram form, a track pad-computer system architecture in accordance with one embodiment of the invention.

[0010] **FIG. 3** shows, in block diagram form, a track pad device and host computer system in accordance with one embodiment of the invention.

[0011] **FIG. 4** shows, in block diagram form, a track pad sensor data acquisition system in accordance with one embodiment of the invention.

[0012] **FIG. 5** shows, in flowchart form, a data acquisition method in accordance with one embodiment of the invention.

[0013] **FIGS. 6 through 9** show various gestures that can be made on a touch pad and interpreted as a user-level task using the disclosed system according to the present disclosure.

[0014] **FIG. 10** illustrates the disclosed touch pad device and a host module for interpreting measured data values from gestures as user-level tasks for a host application.

DETAILED DESCRIPTION

[0015] Referring first to **FIG. 2**, the general architecture of a system incorporating a track pad device in accordance with the invention is illustrated. As shown, system **200** includes track pad device **205** coupled to host module **210** through communication path **215**. Track pad device **205** comprises track pad sensor **220** that generates signals based on user manipulation thereof, data acquisition circuit **225** for capturing or measuring the sensors and transmit circuit **230** for aggregating and periodically transmitting the measured sensor data values to host module **210** via communication path **215**. At host module **210**, receive circuit **235** receives the measured sensor data and passes them to driver application **240**. Driver application **240**, in turn, processes or analyzes the measured data to determine the user's conduct (e.g., a "single click,""double click,""scroll" or "drag" operation), passing the calculated location and/or movement information to other applications such as, for example, window display subsystem application **245**. In accordance with the invention, driver application **240** is executed by host processor **250** which, as shown, is also responsible for executing (at least in part) one or more user applications or processes **255**. It is significant to note that track pad device **205** has no capability to process or analyze data signals (values) acquired from sensor **220**. In accordance with the invention, sensor data is analyzed by a host computer system's general-purpose processor or central processing unit ("CPU").

[0016] The architecture of **FIG. 2** recognizes and takes unique advantage of the processing power of modern CPUs incorporated in host computer systems (e.g., notebook or other personal computers, workstations and servers). This recognition and the architecture of **FIG. 2** permits a computer system **200** that is both lower in cost to manufacture and more flexible than the systems provided by the prior art. Lower costs may be realized by eliminating the prior art's dedicated hardware for processing track pad sensor data (i.e., a processor and associated firmware memory—see components **130** and **135** in **FIG. 1**). Increased flexibility may be realized by providing feature set functionality via software that executes on the host computer's CPU—that is, processing/analyzing measured track pad sensor data on one or more of the host computer's CPUs. In this architecture, track pad functionality may be modified, updated and enhanced through conventional software upgrade procedures.

[0017] The following description is presented to enable any person skilled in the art to make and use the invention as claimed and is provided in the context of the particular examples discussed below, variations of which will be readily apparent to those skilled in the art. Accordingly, the claims appended hereto are not intended to be limited by the disclosed embodiments, but are to be accorded their widest scope consistent with the principles and features disclosed herein.

[0018] Referring to **FIG. 3**, track pad device **300** in accordance with one embodiment of the invention comprises m-row by n-column capacitive sensor array **305**, data acquisition circuit **310** (itself comprising multiplexer ("MUX") circuit **315**, storage capacitor **320** and scan circuit **325**) and Universal Serial Bus ("USB") transmit circuit **330**. During operation, MUX circuit **315** is responsible for coupling and stimulating successive sensor array elements (e.g., rows, columns, or individual pixels—that is, an element at the intersection of a row and column) to storage capacitor **320** in a controlled/sequenced manner and indicating that a measurement cycle has begun to scan circuit **325**. When the charge on storage capacitor **320** reaches a specified value or threshold, scan circuit **325** records the time required to charge storage capacitor **320** to the specified threshold. Accordingly, scan circuit **325** provides a digital value that is a direct indication of the selected sensor array element's capacitance. USB transmit circuit **330** is responsible for aggregating the measured capacitance values into packets and transmitting them in accordance with the USB protocol to host module **335** via USB bus **340**. One of ordinary skill in the art will understand that depending upon the version of USB-used and the bandwidth of bus **340**, USB transmit circuit **330** may transfer each frame of data to host module **335** in more than one, one or more than one packet. When the host module's USB receive circuit **345** receives the measured sensor data from track pad device **300** via USB bus **340**, it unpacks and passes the measured capacitance data to driver application **350**. Driver application **350**, in turn, accepts and processes the raw (measured) capacitance data to provide meaningful cursor movement input to operating system application **355**. (One of ordinary skill in the art will recognize that scan circuit **325** measures capacitance values from sensor array **305** in a predetermined order or sequence and that this sequence must be known by driver application **350** a priori or conveyed to driver application **350** along with the measured sensor data.) In one embodiment, driver application **350** implements track pad algorithms traditionally provided by a dedicated track pad processor such as, for example, processor **130** and firmware memory **135** of **FIG. 1**.

[0019] Referring to **FIG. 4**, a more detailed view of MUX circuit **315** as it can be implemented for a row and column addressable capacitive sensor array is illustrated. As shown, each row in sensor array **400** is electrically coupled to voltage source Vcc **405** through MUX-1**410** and to storage capacitor **415** through MUX-2**420**. (While not shown in detail, each column of sensor array **400** is similarly coupled to Vcc **405** and to storage capacitor **415** through other MUX circuits—block **425**.)

[0020] Referring now to **FIG. 5**, in operation MUX-1**410** couples a first sensor array row to Vcc **405** for a specified period of time (block **500**) and then isolates or disconnects that row from Vcc **405** (block **505**). Next, MUX-2**420** couples the same row to storage capacitor **415** for a specified period of time, or until the voltage on storage capacitor **415** reaches a specified threshold (block **510**). If, during the time MUX-2**420** couples the selected sensor row to storage capacitor **415** the storage capacitor's voltage reaches a specified threshold (the "Yes" prong of block **515**), the digital value corresponding to the time it took to charge storage capacitor **415** to the threshold is recorded by scan circuit **325** (block **520**). If, during the time MUX-**2420** couples the selected sensor row to storage capacitor **415** the storage capacitor's voltage does not reach the specified threshold (the "No" prong of block **515**), the acts of block **500-510** are repeated. Once a digital value corresponding to the capacitance of the selected row has been obtained (block **520**), a check is made to see if there are additional rows in sensor array **400** that need to be sampled. If all the rows in sensor array **400** have been sampled in accordance with blocks **500-520** (the "Yes" prong of block **525**), the same process is used to acquire a capacitance value for each column of sensor elements in sensor array **400** (block **535**). Once all rows and all columns have been processed in accordance with blocks **500-535**, the entire process is repeated (block **540**). If, on the other hand, there are rows in sensor array **400** that have not been sampled in accordance with blocks **500-520** (the "No" prong of block **525**), the next row is selected (block **530**) and the acts of blocks **500-525** are performed.

[**0021**] In one illustrative embodiment: sensor array **400** comprises a 16×32 capacitive grid, providing 48 output channels; Vcc is 3.3 volts; storage capacitor **415** is approximately 10,000 picofarads, an average row capacitance value is approximately 12 picofarads; an average column capacitance value is approximately 9 picofarads; the average change in capacitance of a row or column electrode due to a user's finger touching sensor array **400** is approximately 0.2 picofarads; the threshold value at which a digital capacitance value is obtained is 1.6 volts; and the rate at which MUX circuits **410**, **420** and **425** are switched is 6 megahertz. It has been found, for these values, that its takes approximately 580-600 sample cycles to charge storage capacitor **415** to the threshold voltage. In one embodiment, the digital capacitance value is, in fact, a count of the number of sampling cycles required to charge storage capacitor **415** to the threshold value. One of ordinary skill in the art will recognize that this value is directly related to the sensor element's (e.g., row or column) capacitance value. In this embodiment, scan circuit **325** (in conjunction with MUX circuits **410**, **420** and **425** and storage capacitor **415**) measures each of the 48 sensor array outputs 125 times each second, with each measurement comprising a 10-bit value (unsigned integer). Referring to the 48 measurements acquired by scan circuit **325** from sensor array **400** in each of the 125 epochs as a frame, the illustrative track pad sensor device generates:

$$\left(\frac{48 \text{ channels}}{\text{frame}}\right)\left(\frac{10 \text{ bits}}{\text{channel}}\right)\left(\frac{125 \text{ frames}}{\text{second}}\right)\left(\frac{1 \text{ byte}}{8 \text{ bits}}\right) = 7{,}500 \text{ bytes/second.}$$

[**0022**] As noted with respect to **FIG. 2** and as further shown in **FIG. 3**, driver application **350** is executed general purpose processing unit **360** that is also responsible for executing user applications and tasks, e.g., **365**. That is, in accordance with the invention raw track pad sensor data is analyzed by one, or more, general-purpose processing units associated with the host computer system and not by a dedicated processor or processing circuit(s) associated with track pad device **300**. A direct consequence of the architecture of **FIGS. 2 and 3** is that the processing resources (e.g., CPUs) tasked with analyzing track pad sensor data must be shared with other computer system processing needs such as other system level and user level applications.

[**0023**] Various changes in the materials, components, and circuit elements of the described embodiments are possible without departing from the scope of the following claims. Consider, for example, the system of **FIG. 3**. Other embodiments could include a smaller (e.g., 10×16) or larger (e.g., 32×32) sensor array **305**. Further, frame rates other than 125 Hertz ("Hz") and sample resolutions other than 10 bits are possible. It will also be understood that the host computer system may comprise more than one general-purpose processing unit (e.g., processor **250**). In addition, some of the circuitry identified in **FIGS. 2 and 3** as being integral to track pad device **205** or **300** may be embodied in circuitry also used for other functions. For example, transmit circuits **230** and **330** may be shared by other USB input devices such as, for example, a keyboard. In addition, one of ordinary skill in the art will recognize that the invention is also applicable to track pad sensor devices that are pixilated rather that row-column addressable. It will be further recognized that the operational procedure outlined in **FIG. 5** may be modified. For instance, sensor column values may be obtained before sensor row values. Alternatively, sensor row and sensor column data may be interlaced and/or measured at the same time. In any event, it will be recognized that scan circuit **325** measures sensor pad characteristic values (e.g., capacitance or resistance) in a set order and that this order must be known or communicated to driver application **350**. In yet another embodiment, scan circuit **325** may measure sensor characteristic values in any convenient manner and reorder them into a sequence known or expected by driver application **350** prior to transmission by transmit circuit **330**.

[**0024**] Various gestures can be detected by the track pad device **300** and interpreted as user-level tasks or operations. For example, the user gesture made on the sensing elements **305** of the track pad device **300** include a touch of one digit, a touch of two or more digits concurrently, a touch with sliding movement of one or more digits, a touch and sliding movement of one or more digits concurrently with a stationary touch of one or more digits, a touch and contractive movement of two or more digits, a tap of one or more digits, a touch and expansive movement of two or more digits, a touch of one or more digits with a tap of one or more digits, a touch of a portion (e.g., palm) of a hand, and a touch and movement of a portion of a hand. Moreover, the digits or portions of one or two hands can also be used for gestures made on the track pad device.

[**0025**] Some one-finger gestures include a one-finger tap on the track pad device to implement a left click mouse operation and a one-finger down and hold on the track pad device to implement a cursor motion. Some two-finger gestures on the track pad device include: (1) two fingers down together and a vertical movement to implement a vertical scroll operation; (2) two fingers down together and a horizontal movement to implement a horizontal scroll operation; (3) a two-finger tap to implement an operation; (4) two fingers both down and then spread apart and back together to implement a zoom operation; (5) two fingers down together and rotated in a circular motion either clockwise/counter-clockwise to implement a rotate operation; (6) one finger down and then a second finger down to implement an operation; (7) one finger down and then a second finger tap to implement an operation; and (8) two fingers down together and a diagonal movement to implement an operation.

[0026] Some three-finger and four-finger gestures on the track pad device to implement user-level tasks or operations are shown in the Table below.

Three-Finger Gestures	Four-Finger Gestures
three-finger tap	four-finger tap
three fingers down together and sweep away or towards from self	four fingers down together and sweep away from or towards self on the track pad
three fingers down together and sweep sideways towards left or right	four fingers down together and sweep towards left or right on the track pad
three fingers down together and a vertical, a horizontal, or a diagonal movement	four fingers down together and a vertical, a horizontal, or a diagonal movement
three fingers down together and rotate in circular motion either clockwise/ counter-clockwise	four fingers down together and rotate in a circular motion either clockwise/counter-clockwise
two fingers down and then third finger down or tap	three fingers down and then a fourth finger down or tap
one finger down and then two fingers down or tap	two fingers down and then two fingers down or tap
	one finger down and then three fingers down or tap

[0027] In **FIGS. 6 through 9**, a number of exemplary gestures **600**, **700**, **800**, and **900** and corresponding user-level tasks or operations are shown and described below. These gestures **600**, **700**, **800**, and **900** and the corresponding user-level tasks are exemplary. One skilled in the art will appreciate that other gestures and user-level tasks are possible beyond those expressly described herein.

[0028] Referring to **FIG. 6**, user gestures **600** on the sensing elements of a track pad device (not shown) can be used to perform various mouse operations or tasks of a host application. The mouse operations can include, but are not limited to, a point, a click, a double click, a right click, a drag/select, and a scroll. For example, a point operation can be implemented with a first gesture **602** involving touching and moving any two adjacent fingers on the track pad device. A click operation can be implemented with a second gesture **604** involving a tap (momentary touch) of any two adjacent fingers on the track pad device. A double click operation can be implemented with a third gesture **606** involving a tap (momentary touch) of any three adjacent fingers on the track pad device. A right click operation can be implemented with a fourth gesture **608** involving a tap of the thumb, middle finger, and ring finger on the track pad device. A drag/select operation can be implemented with a fifth gesture **610** involving a touch and movement of three fingers on the track pad device. A scroll operation can be implemented with a sixth gesture **612** involving the touch and slide up/down of four fingers on the track pad device.

[0029] The system can also be used to emulate operation of a three-button mouse. For example, a click of a middle button can be implemented by a seventh gesture **614** involving a tap of the thumb, index, and middle fingertips on the track pad device. In addition, a right click in a three-button emulation can be implemented by a eighth gesture **616** involving the spreading of the hand and then the tap of the thumb, ring, and pinky fingers on the track pad device.

[0030] Referring to **FIG. 7**, additional user gestures **700** on the track pad device can be used to implement various editing and cursor operations or tasks of a host application. The editing operations can include, but are not limited to, cut, copy, paste, undo previous operation, and redo previous operation. The cursor operations can include, but are not limited to, move cursor, select with cursor, tab cursor, move cursor to home, move cursor to end, page up, and page down.

[0031] For example, a cut operation can be implemented with a first gesture **702** involving a touch and pinch of the thumb and middle finger on the track pad device. A copy operation can be implemented with a second gesture **704** involving a tap of the thumb and middle finger on the track pad device. A paste operation can be implemented with a third gesture **706** involving a touch and expansion of the thumb and middle finger on the track pad device. An Undo/Redo operation can be implemented with a fourth gesture **708** involving a touch and slide of the thumb and middle finger up/down on the track pad device. Undoing just one operational step can be implemented by sliding quickly, or multiple steps can be implemented by sliding gradually. A Tab/BackTab operation can be implemented with a fifth gesture **710** involving a touch and slide of the thumb and middle finger right/left on the track pad device. Just one tab can be implemented by sliding-quickly, and repetitive tabs can be implemented by sliding gradually.

[0032] An arrows operation can be implemented with a sixth gesture **712** involving a touch with a slide of any finger on the track pad device to move the text cursor. A Text Select operation can be implemented with a seventh gesture **714** involving a touch with a slide up/down of three spread fingers on the track pad device. A Tab/Back Tab operation can be implemented with a eighth gesture **716** involving a touch of the thumb and middle finger and a slide left/right on the track pad device. A Home/End operation can be implemented with a ninth gesture **718** involving a touch of four spread fingers with slide left/right on the track pad device. Finally, a Page-Up/Page-Down operation can be implemented with a tenth gesture **720** involving a touch of four spread fingers with a slide up/down on the track pad device.

[0033] Referring to **FIG. 8**, additional user gestures **800** on the sensing elements of the track pad can be used to implement various file and application operations or tasks of a host application. The file operations can include, but are not limited to, open file, close file, save file, new file, print file, next file, and previous file. The application operations can include, but are not limited to, show desktop, exit application, and switch application window.

[0034] For example, an open file operation can be implemented with a first gesture **802** involving a touch with counter-clockwise rotation of the thumb and three fingers on the track pad device. A close file operation can be implemented with a second gesture **804** involving a touch with clockwise rotation of the thumb and three fingers on the track pad device. A save file operation can be implemented with a third gesture **806** involving a touch with contraction of the thumb and three fingers on the track pad device. A new file operation can be implemented with a fourth gesture **808** involving a touch with expansion of the thumb and three inner fingers on the track pad device. A print file operation can be implemented with a fifth gesture **810** involving pre-spreading of the hand and then a touch with further expansion of the thumb and three outer fingers on the track pad device. A next file operation can be implemented with a sixth gesture **812** involving a touch with a slide of the thumb and three fingertips to the left on the track pad device. A previous file operation can be implemented with a seventh gesture **814** involving a touch with a slide of the thumb and three fingertips to the right on the track pad device.

[**0035**] An operation for showing the desktop can be implemented with a eighth gesture **816** involving touching the track pad with spread thumb and three fingers and then sliding to the left on the track pad device. An exit application operation can be implemented with a ninth gesture **818** involving a touch with clockwise rotation of spread thumb and three fingers on the track pad device. A switch application operation can be implemented with a tenth gesture **820** involving touching the track pad with spread three fingers and thumb and then sliding left or right on the track pad device. Sliding crisply can be used to advance just one window, and sliding gradually can be used to scroll through whole list.

[**0036**] Referring to **FIG. 9**, additional user gestures **900** on the track pad device can be used to perform various web-browsing and keyboard operations or tasks of a host application. The browser operations can include, but are not limited to, back, forward, scroll, zoom in, zoom out, and find in page. The keyboard operations can include, but are not limited to, shift, control/command, and select key.

[**0037**] For example, a back operation can be implemented with a first gesture **902** involving a touch with a slide of the thumb and three fingertips to the left on the track pad device. A forward operation can be implemented with a second gesture **904** involving a touch with a slide of the thumb and three fingertips to the right on the track pad device. A scroll operation can be implemented with a third gesture **906** involving a touch with a slide of four fingers up/down on the track pad device. If desired, the thumb can be rested on the track pad device after starting the slide on the track pad device. A zoom-in operation can be implemented with a fourth gesture **908** involving a touch with expansion of the thumb and four fingers on the track pad device. A zoom-out operation can be implemented with a fifth gesture **910** involving a touch with a contraction of the thumb and four fingers on the track pad device. A find-in-page operation can be implemented with a sixth gesture **912** involving a touch with a pinch of the thumb and two fingertips on the track pad device. A fine scroll operation can be implemented with a seventh gesture **914** involving the "roll" of fingers instead of sliding them on the track pad device. If desired, the thumb can be rested on the track pad device once the scroll starts. A pointing operation can be implemented with an eighth gesture **916** involving dropping other fingers to point with all five fingers on the track pad device once pointing starts. Additional gestures can include a ninth gesture **918** involving moving all five fingers on the track pad device and a tenth gesture **920** involving a touch and movement of a portion (e.g., palm) of a hand on the track pad device.

[**0038**] Referring to **FIG. 10**, a computer system **1000** according to certain teachings of the present disclosure is illustrated. The system **1000** includes track pad device **1010** having sensor array **1012**, data acquisition circuit **1014**, first communication circuit **1016**, host module **1040** having second communication circuit **1042**, one or more host processors **1044**, software **1046**, and host application **1048**, each of which is similar to previous embodiments discussed above. For example, the second communication circuit **1042** is operatively coupled to the first communication circuit **1016** via communication path **1030**, such as a USB bus. The one or more host processors **1044** are operatively coupled to the second communication circuit **1042**, and at least one of the host processors **1044** is responsible, at least in part, for executing user-level tasks of the host application **1048**.

[**0039**] Sensing elements of the track pad array **1012** measure data values **1120** (e.g., raw data) in response to a user performing a user gesture (e.g., the two-finger gesture shown) on the array **1012**. The touch pad device **1010** does not process the raw, measured data values representative of the user gesture. Instead, the data acquisition circuit **1014** obtains the measured data values **1020** of the array **1012**, and the first communication circuit **1016** transmits the measured data values **1020** to the second communication circuit **1042** of the host module **1040**. The gesture processing software **1046** on the host module **1040** executes on at least one of the host processors **1044**. Upon receiving the raw data values-**1020**, the gesture processing software **1046** interprets the raw data values **1020** communicated from the track pad device **1010** and makes interpreted data **1122** available to the host application **1048** for execution as a user-level task.

[**0040**] To interpret the raw data values **1020**, the software **1046** uses algorithms for processing the data and interpreting it into data for user-level tasks or operations to be implemented at the host module **1040**. The algorithms used can include those algorithms known in the art, such as existing algorithms used with touch-screens, existing algorithms used with touch pads that process gestures at the touch pad, and existing algorithms used for navigation features provided on a prior art Fountain touch pad. For example, the software **1046** can use algorithms, such as those disclosed in U.S. Pat. Nos. 6,570,577 and 6,677,932 and in U.S. Patent Publication Nos. 2005/0104867 and 2002/0015024, which are incorporated herein by reference.

[**0041**] Briefly, an exemplary algorithm for the system **1000** may first involve the track pad device **1010** obtaining raw row and column data from the sensor array **1012** at a first point in time and transmitting that raw data to the host module **1040**. The software **1046** receives that row and column data as a current frame. The software **1046** then filters or smoothes the data and separately counts the number of areas that have created increased signal in the rows and columns of the current frame. These areas of increased signal correspond to images of the user's digits on the track pad device **1010**. (The areas of increased strength are referred to here as the finger count). If the finger count in either the rows or columns of the current frame is zero, then the finger count for the entire track pad device **1010** of the current frame is assumed to be zero. Otherwise, the finger count for the entire track pad device **1010** is set to the maximum of the row finger count or the column finger count. Once the maximum finger count of the current frame has been established, it is compared to the maximum finger count in one or more previous frames of data obtained from the track pad device **1010**. If the maximum finger count has changed between the current and previous frames, the state of this change is examined to determine if the change in finger count represents a gesture or is part of a gesture.

[**0042**] When examining the state between the current and previous finger counts, the amount of time that the previous finger count was held may be considered. In one example, the maximum finger count in the current frame may be one, while the maximum finger count in the previous frame may have been two. If the two-finger count in the previous frame lasted less than 250-ms, then the user has made a second finger tap on the track pad device **1010** corresponding to a gesture for a mouse button operation, for example. On the other hand, if the two-finger count in the previous frame lasted more than 250-ms, then the user has used a two-finger gesture for performing a scroll operation, for example, and no mouse button operation is issued.

[**0043**] When examining the state between the current and previous finger counts, the locations of the areas of increased signal strength (i.e., the rows and columns of the user's digits on the track pad device **1010**) may be considered. In one example, the maximum finger counts in the current and previous frames may both be two. If the two-finger count in the previous frame had first row and column values that are different than second row and column values in the current frame, then the user has moved two fingers on the track pad device **1010**, which may correspond to a gesture for a scroll operation or the like.

[**0044**] The foregoing description of preferred and other embodiments is not intended to limit or restrict the scope or applicability of the inventive concepts conceived of by the Applicants. In exchange for disclosing the inventive concepts contained herein, the Applicants desire all patent rights afforded by the appended claims. Therefore, it is intended that the appended claims include all modifications and alterations to the full extent that they come within the scope of the following claims or the equivalents thereof.

What is claimed is:

1. A computer system, comprising:

a track pad device having a first communication circuit and a plurality of sensing elements, the track pad device obtaining raw data with the sensing elements in response to a user gesture and transmitting raw data with the first communication circuit;

a second communication circuit operatively coupled to the first communication circuit via a communication path;

one or more host processors operatively coupled to the second communication circuit, at least one of the host processors being responsible, at least in part, for executing user-level tasks of a host application; and

software executing on at least one of the host processors, the software interpreting raw data from the track pad device and making interpreted data available to the host application for execution as a user-level task.

2. The computer system of claim 1, wherein the user gesture is selected from the group consisting of a touch of one digit, a touch of two or more digits concurrently, a touch and sliding movement of one or more digits, a touch and sliding movement of one or more digits concurrently with a stationary touch of one or more digits, a touch and contractive movement of two or more digits, a tap of one or more digits, a touch and expansive movement of two or more digits, a touch of one or more digits with a tap of one or more digits, a touch of a portion of a hand, and a touch and movement of a portion of a hand.

3. The computer system of claim 1, wherein the user-level task of the host application is selected from the group consisting of a mouse operation, an editing operation, a cursor operation, a file operation, an application operation, a browser operation, and a keyboard operation.

4. The computer system of claim 3, wherein the mouse operation is selected from the group consisting of a point, a click, a double click, a right click, a drag/select, and a scroll.

5. The computer system of claim 3, wherein the editing operation is selected from the group-consisting of cut, copy, paste, undo previous operation, and redo previous operation.

6. The computer system of claim 3, wherein the cursor operation is selected from the group consisting of move cursor, select with cursor, tab cursor, move cursor to home, move cursor to end, page up, and page down.

7. The computer system of claim 3, wherein the file operation is selected from the group consisting of open file, close file, save file, new file, print file, next file, and previous file.

8. The computer system of claim 3, wherein the application operation is selected from the group consisting of show desktop, exit application, and switch application window.

9. The computer system of claim 3, wherein the browser operation is selected from the group consisting of back, forward, scroll, zoom in, zoom out, and find in page.

10. The computer system of claim 3, wherein the keyboard operation is selected from the group consisting of shift, control/command, and select key.

11. The computer system of claim 1, wherein the first and second communication circuits are adapted to operate in accordance with a universal serial bus protocol.

12. The computer system of claim 1, wherein the plurality of sensing elements comprises a resistive sensor array or a capacitive sensor array.

13. The computer system of claim 1, wherein the track pad device comprises a data acquisition circuit electrically coupled between the sensing elements and the first communication circuit and encoding digital values measured for each of the plurality of sensing elements.

14. A host module operatively coupled to a track pad device, the track pad device having a first communication circuit and a plurality of sensing elements, the track pad device obtaining raw data with the sensing elements in response to a user gesture and transmitting raw data with the first communication circuit, the host module comprising:

a second communication circuit operatively coupling to the first communication circuit via a communication path;

one or more host processors operatively coupled to the second communication circuit, at least one of the host processors being responsible, at least in part, for executing user-level tasks of a host application; and

software executing on at least one of the host processors, the software interpreting raw data from the track pad device and making interpreted data available to the host application for execution as a user-level task.

15. The host module of claim 14, wherein the user gesture is selected from the group consisting of a touch of one digit, a touch of two or more digits concurrently, a touch with sliding movement of one or more digits, a touch and sliding movement of one or more digits concurrently with a stationary touch of one or more digits, a touch and contractive movement of two or more digits, a tap of one or more digits, a touch and expansive movement of two or more digits, a touch of one or more digits with a tap of one or more digits, a touch of a portion of a hand, and a touch and movement of a portion of a hand.

16. The host module of claim 14, wherein the user-level task is selected from the group consisting of a mouse operation, an editing operation, a cursor operation, a file operation, an application operation, a browser operation, and a keyboard operation.

17. The host module of claim 16, wherein the mouse operation is selected from the group consisting of a point, a click, a double click, a right click, a drag/select, and a scroll.

18. The host module of claim 16, wherein the editing operation is selected from the group consisting of cut, copy, paste, undo previous operation, and redo previous operation.

19. The host module of claim 16, wherein the cursor operation is selected from the group consisting of move cursor, select with cursor, tab cursor, move cursor to home, move cursor to end, page up, and page down.

20. The host module of claim 16, wherein the file operation is selected from the group consisting of open file, close file, save file, new file, print file, next file, and previous file.

21. The host module of claim 16, wherein the application operation is selected from the group consisting of show desktop, exit application, and switch application window.

22. The host module of claim 16, wherein the browser operation is selected from the group consisting of back, forward, scroll, zoom in, zoom out, and find in page.

23. The host module of claim 16, wherein the keyboard operation is selected from the group consisting of shift, control/command, and select key.

24. The host module of claim 14, wherein the second communication circuits is adapted to operate in accordance with a universal serial bus protocol.

25. The host module of claim 14, wherein the software comprises a driver application determining a computer command from raw data and making the computer command available to the host application for execution as the user-level task.

26. A method comprising:

measuring raw data in response to a user gesture made on sensing elements of a track pad device;

transmitting raw data to one or more host processors of a host module, at least one of the host processors being responsible, at least in part, for executing user-level tasks of a host application;

interpreting raw data with software executing on at least one of the host processors; and

making interpreted data available to the host application for execution as a user-level task by the host application.

27. The method of claim 26, wherein the user gesture is selected from the group consisting of a touch of one digit, a touch of two or more digits concurrently, a touch with sliding movement of one or more digits, a touch and sliding movement of one or more digits concurrently with a stationary touch of one or more digits, a touch and contractive movement of two or more digits, a tap of one or more digits, a touch and expansive movement of two or more digits, a touch of one or more digits with a tap of one or more digits. a touch of a portion of a hand, and a touch and movement of a portion of a hand.

28. The method of claim 26, wherein the user-level task is selected from the group consisting of: a mouse operation, an editing operation, a cursor operation, a file operation, an application operation, a browser operation, and a keyboard operation.

29. The method of claim 26, wherein the act of measuring raw data in response to a user gesture made on a track pad device comprises measuring-a characteristic for each of a plurality of sensor elements stimulated by the user gesture, each measurement being encoded by a digital value.

30. The method of claim 29, wherein the act of measuring a characteristic comprises determining a digital value representing a capacitance value or a resistance value.

31. The method of claim 26, wherein the act of transmitting raw data comprises transmitting digital values in accordance with a universal serial bus protocol.

* * * * *