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 sensor's 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").

[**0013**] 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.

[**0014**] 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.

[**0015**] 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**.

[**0016**] 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-**1410** and to storage capacitor **415** through MUX-**2420**. (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**.)

[**0017**] Referring now to **FIG. 5**, in operation MUX-**1410** 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-**2420** 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-**2420** 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