Technical Disclosure Commons

January 19, 2016

MOUNTING SCREW BASED SYSTEM IDENTIFICATION

Katie Roberts-Hoffman

Bernard Thompson

Follow this and additional works at: http://www.tdcommons.org/dpubs_series

MOUNTING SCREW BASED SYSTEM IDENTIFICATION

ABSTRACT

A mounting screw identification system can be used to determine a software configuration for an electronic device based on a type of electronic device chassis. A Printed Circuit Board (PCB), having screw holes with an open electrical circuit formed upon them, is placed in an electronic device chassis with mounting screw receiving threads. Conductive mounting screws are then inserted through the aligned PCB screw holes into the mounting screw receiving threads of the chassis to hold the PCB in place with the device chassis and also fill the electrical gap in the PCB screw holes to close the electrical circuit. A processor in the electronic device reads values on general purpose input/output (GPIO) pins, which can be either high or low, that are connected to the PCB screw holes to determine which screw holes have inserted screws that complete the electrical connection. Subsequently, the processor decodes the GPIO pin values to identify a binary code for the device chassis and determines the corresponding software configuration for the electronic device.

PROBLEM STATEMENT

In modern electronic devices, it is common for a single Printed Circuit Board (PCB) design to be used in multiple electronic devices' chassis with minimal variation. For example, a single PCB device motherboard may be used in both a small and a large laptop. This is primarily done to save development costs on components while assembling electronic devices on a large scale. For example, if a PCB motherboard is employed in a large and a small laptop electronic device, the processing software on the PCB motherboard should be able to detect the type of chassis installed in the electronic device. This may be required in order to set software tuning

values for the device's cooling fan, for example. The tuning values for an electronic device's

cooling fan is based on the thermal efficiency of its chassis, and the thermal efficiency varies for

different kinds of devices. As a result, it is helpful to identify the type of device chassis if a PCB

motherboard needs to be deployed for different electronic devices.

The identification of the PCB installed in an electronic device is performed in software

by setting general purpose input/output (GPIO) pins of a device processor, connected to the PCB

motherboard, to high or low values, i.e., 1 or 0. The binary combination formed by the GPIO pin

values forms a board ID number and can have a particular software configuration stored

corresponding to it in the device memory. The processor of the device can read these pin values

and select the software configuration for the electronic device corresponding to the determined

board ID number.

Current manufacturing practices vary the board ID number (read by GPIO pins) by

adding or removing specific resistors on the PCB. However, this requires setting the PCBs with

the proper resistors during the surface mounting (SMT) step of manufacturing. This complexity

can lead to a number of problems. For example, the SMT step is generally performed well before

final assembly of the full chassis which might lead to a scenario where printed circuit boards

with static IDs are incorrectly assembled into a different chassis. In another example, while

repairing an electronic device, repair facilities such as return merchandise authorization (RMA)

may need to replace bad PCB motherboards. As a result, these facilities are required to stock

multiple versions of the PCB motherboards with different resistor configurations, or add an

expensive rework procedure to adjust the resistors in their RMA flow.

There is an opportunity to develop a different type of system for matching a software

configuration for an electronic device to the device chassis.

DETAILED DESCRIPTION

The systems and techniques described in this disclosure relate to a mounting screw identification system that can be used to determine appropriate software configuration for an electronic device. The electronic device can be a mobile device, a smartphone, a tablet, a handheld electronic device, a laptop, a desktop computer, etc. The system is primarily employed during assembly of an electronic device.

Fig. 1 depicts a printed circuit board (PCB) 102 to be installed in an electronic device. The PCB mechanically supports and electrically connects various electronic components using conductive tracks. The electronic components can be, e.g., capacitors, resistors, microprocessors, microcontrollers, etc. Further, the PCB has multiple screw holes. As depicted, the PCB 102 has eight screw holes such as 104, 106,108, and 110. These holes are created at the time of PCB manufacturing as an open circuit and allow mounting screws to pass through them and mechanically attach the PCB to an electronic device chassis and electrically complete the device chassis circuit configuration as well. The screw holes 104, 106, 108, and 110 are connected to GPIO pins of the electronic device processor (further depicted in Fig. 3). The processor of the electronic device currently reads the value from these screw holes as 1; because these pins are not electrically grounded by any conductive screw, the processor value remains high.
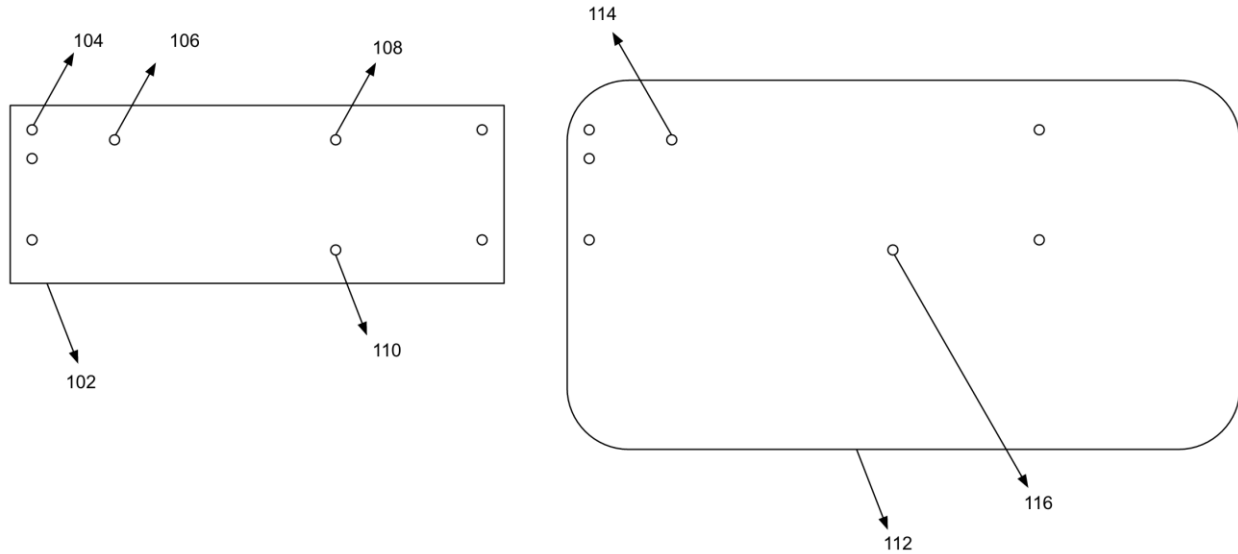
Fig. 1

Fig. 1 also depicts an electronic device chassis 112. The chassis may be manufactured for a small laptop device such as a netbook. The chassis has multiple screw threads such as 114 and 116, which receive PCB mounted screws. The screw threads are usually circular or helical structures. Through these threads, compatible screws pass to form a fastening mechanism. In this way, a component such as a PCB can be mechanically attached to the chassis using screws mounted through the PCB screw holes and inserted into chassis screw threads.

Fig. 2 depicts printed circuit board 202. The PCB is same as the one depicted in Fig. 1. The PCB 202 includes the same components, conductive paths, and mounting screw holes (204, 206, 208, and 210) as those of PCB 102. Similar to the PCB of Figure 1, screw holes 204, 206, 208, and 210 are connected to GPIO pins of the electronic device processor (further depicted in Fig. 4). The processor reads the value from these screw holes as 1, because these pins are currently not grounded by any screw and, as such, the processor value remains high.

Fig. 2 also depicts another electronic device chassis 212. The chassis may be manufactured for a small tablet computing device. The device chassis may include mounting screw threads 214, 216 (and others) similar to the screw threads described in relation to Fig. 1.
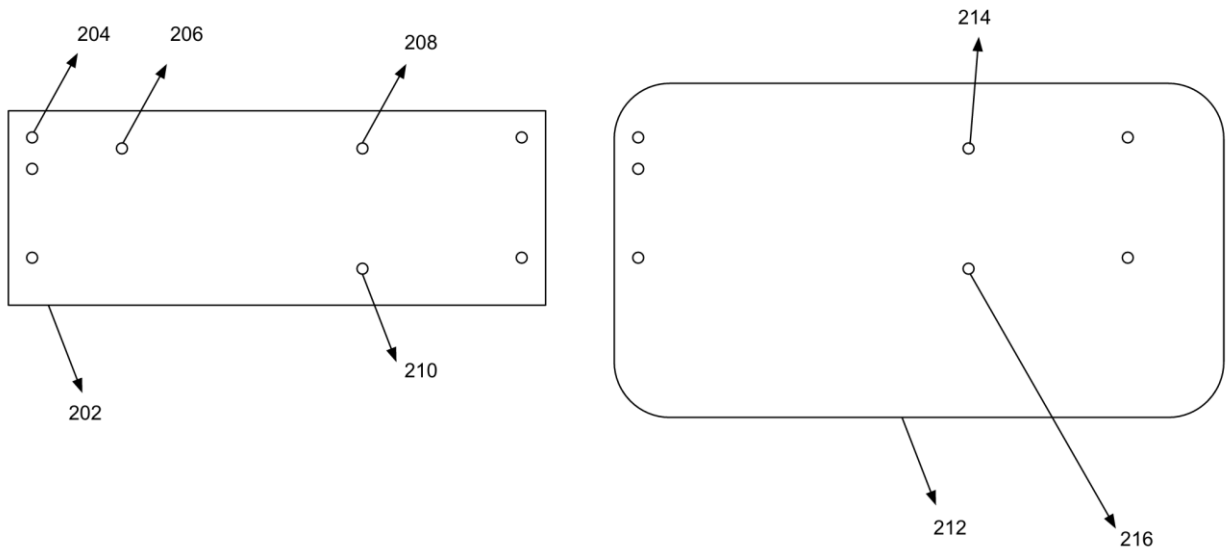
Fig. 2

The screw receiving threads created in device chassis (112 and 212) of both Fig. 1 and

Fig. 2 are distributed in a way so that they can receive identical electronic components (a PCB),

irrespective of the chassis size. However, the configuration of the mounted receiving threads in

Fig. 2 is different from the chassis configuration of mounted receiving threads in Fig. 1. For

example, as depicted, the chassis in Fig. 1 has a receiving thread 114, which is not located in

chassis of Fig. 2. The device chassis of Fig. 2, has a thread 214, not present in device chassis of

Fig. 1. In contrast, the remaining five screw receiving threads in both Fig. 1 and Fig. 2 have

similar relative locations.

Fig. 3 depicts the assembly of the PCB 302 with a first electronic device chassis 312. The

PCB and device chassis correspond to the device depicted in Fig. 1. The PCB is placed over the

chassis in such a manner so that the PCB screw holes align over their corresponding screw

receiving threads. Screws are then inserted through the PCB screw holes into their corresponding

screw threads in the device chassis. For example, if the screw and holes have a circular shape,

the insertion of the screw forms a circular latching mechanism during device assembly and the

PCB gets mechanically coupled with the chassis. As depicted in Fig. 3, screws are inserted through all the screw holes except for screw hole 308. As mentioned previously with respect to Fig. 1, PCB screw holes 304, 306, 308, and 310 are formed with an open electrical circuit and connected to GPIO pins of the processor. When conductive screws are inserted through particular screw holes 304, 306, and 310 and not other screw holes 308, the values read from these screw holes are no longer all 1.
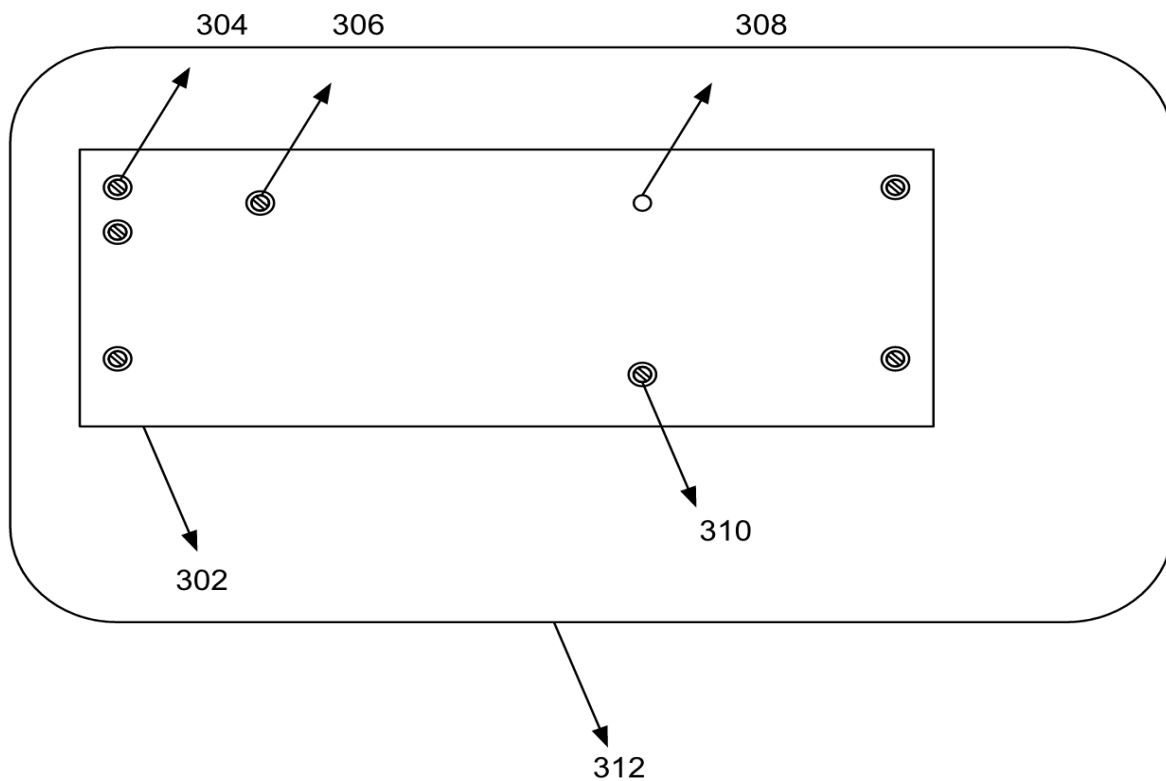
Fig. 3

Fig. 4 depicts a processor 402 of an electronic device with GPIO pins connected to selected screw holes of the device PCB. Screw holes 404, 406, 408, and 410 are connected to processor GPIO pins ID 1, ID 2, ID 3 , and ID 4 respectively. The figure corresponds to the same small electronic device as that of Fig. 1 and Fig. 3. The screw holes 404, 406, and 410 have conductive screws A (412), B (414), and C (416) inserted through them connecting to the device

chassis respectively. These three screw holes are considered grounded by the processor because they have conductive screws inserted through them connecting to the device chassis. As depicted in Fig. 4, inserting screws into the screw holes complete the electric circuit between the GPIO pins and the ground and it drives their respective pin values to 0. The processor reads the GPIO pin values for ID 1, ID 2, and ID 4 as 0. For the screw hole 408, because it is not grounded by a screw, the electrical connection between the GPIO pin, ID3 and the ground is not completed. The value read by the processor for GPIO pin ID3 is 1. The processor forms a binary combination from the read values of the GPIO pins. The binary values are read in the order from ID 1 to ID 4. Based on the pin values, the binary combination obtained by the processor as shown is 0010.
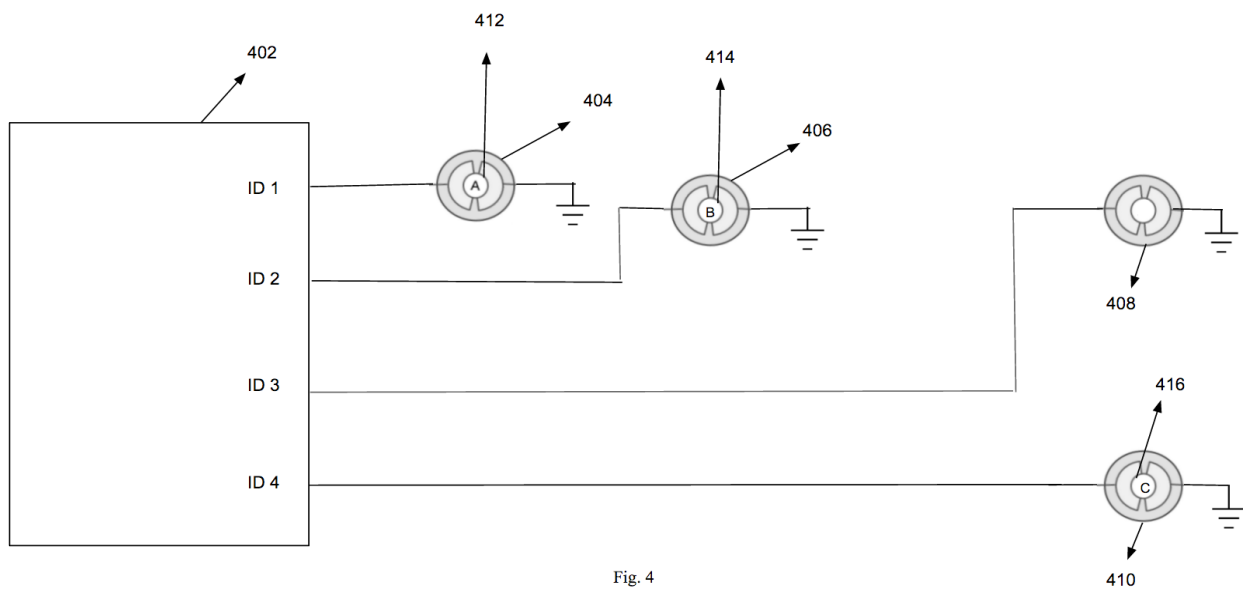

Fig. 4

The processor of the device selects a software configuration for the electronic device corresponding to the binary combination obtained. For example, a binary combination of 0010 may have a corresponding software configuration stored for configuring a particular small laptop electronic device. The configuration can be for example, a slower fan speed, low power adapter setting for a small battery, etc.

Similarly Fig. 5 depicts screws mounted through PCB (502) screw holes and inserted into corresponding screw receiving threads of a second device chassis 512. The PCB and device chassis correspond to the small tablet computing device depicted in Fig. 2. As depicted in Fig. 5, screws are inserted through all the screw holes except for screw hole 506. As mentioned previously with respect to Fig. 2, PCB screw holes 504, 506, 508, and 510 are formed with an open electrical circuit and connected to GPIO pins of the processor. When conductive screws are inserted through particular screw holes 504, 508, and 510 and not other screw holes 506, the values read from these screw holes are no longer all 1.
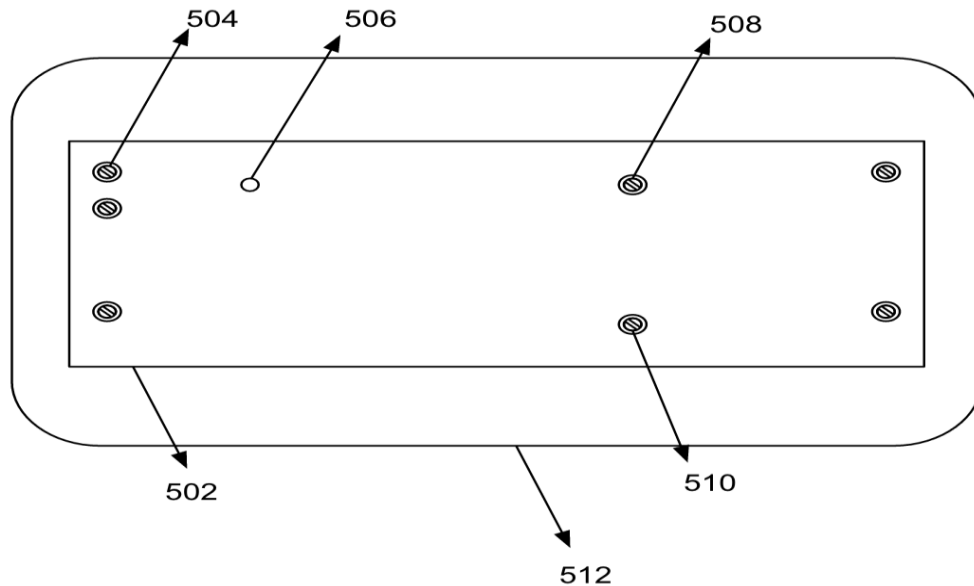

Fig. 5

Fig. 6 depicts a processor 602 of an electronic device with GPIO pins connected to selected screw holes of the device PCB. Screw holes 604, 606, 608, and 610 are connected to processor GPIO pins ID 1, ID 2, ID 3 , and ID 4 respectively. The figure corresponds to the same small electronic computing device as that of Fig. 2 and Fig. 5. Similar to Fig. 4, the screw holes 604, 608, and 610 are considered grounded by the processor as they have screws A (612), B (614), C (616) inserted through them respectively. As depicted in Fig. 6, inserting screws into the

screw holes complete the electric circuit between the GPIO pins and the ground and it drives their respective pin values to 0. Screw hole 606 is read as high by the processor as no screw is inserted through it connecting the GPIO pin, ID2 to the ground. The processor reads its GPIO pin value as 1. The binary combination obtained by the processor in this case is 0100.
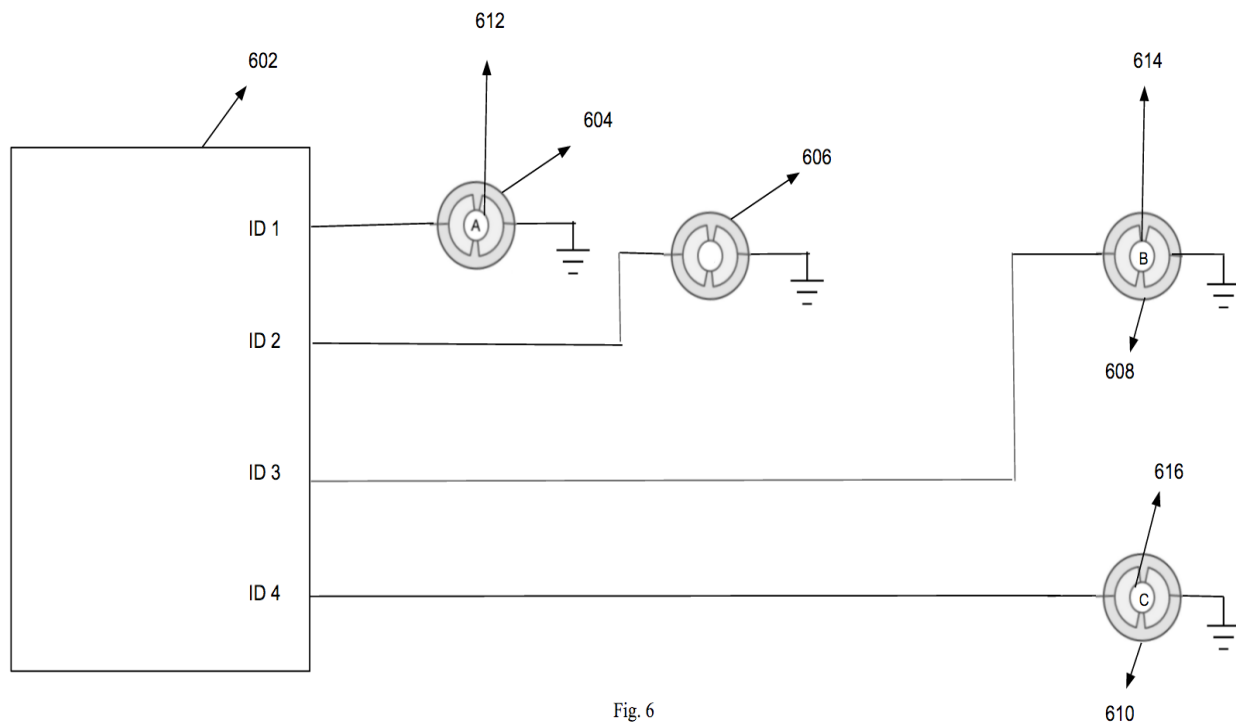
Fig. 6

The processor of the device selects a software configuration for the electronic device corresponding to the binary combination obtained. For example, a binary combination of 0100 may have a corresponding software configuration stored for configuring a small tablet electronic device. The configuration can be for example, a fast fan speed, high power adapter setting for a large battery, etc.

The subject matter described herein in this disclosure can be implemented in hardware (for example, computers, circuits, or processors). The subject matter can be implemented on devices such as a smartwatch, smart thermostat, electronic fitness band/gear, electronic wearable

device, etc. Specific examples disclosed are provided for illustrative purposes and do not limit

the scope of the disclosure.