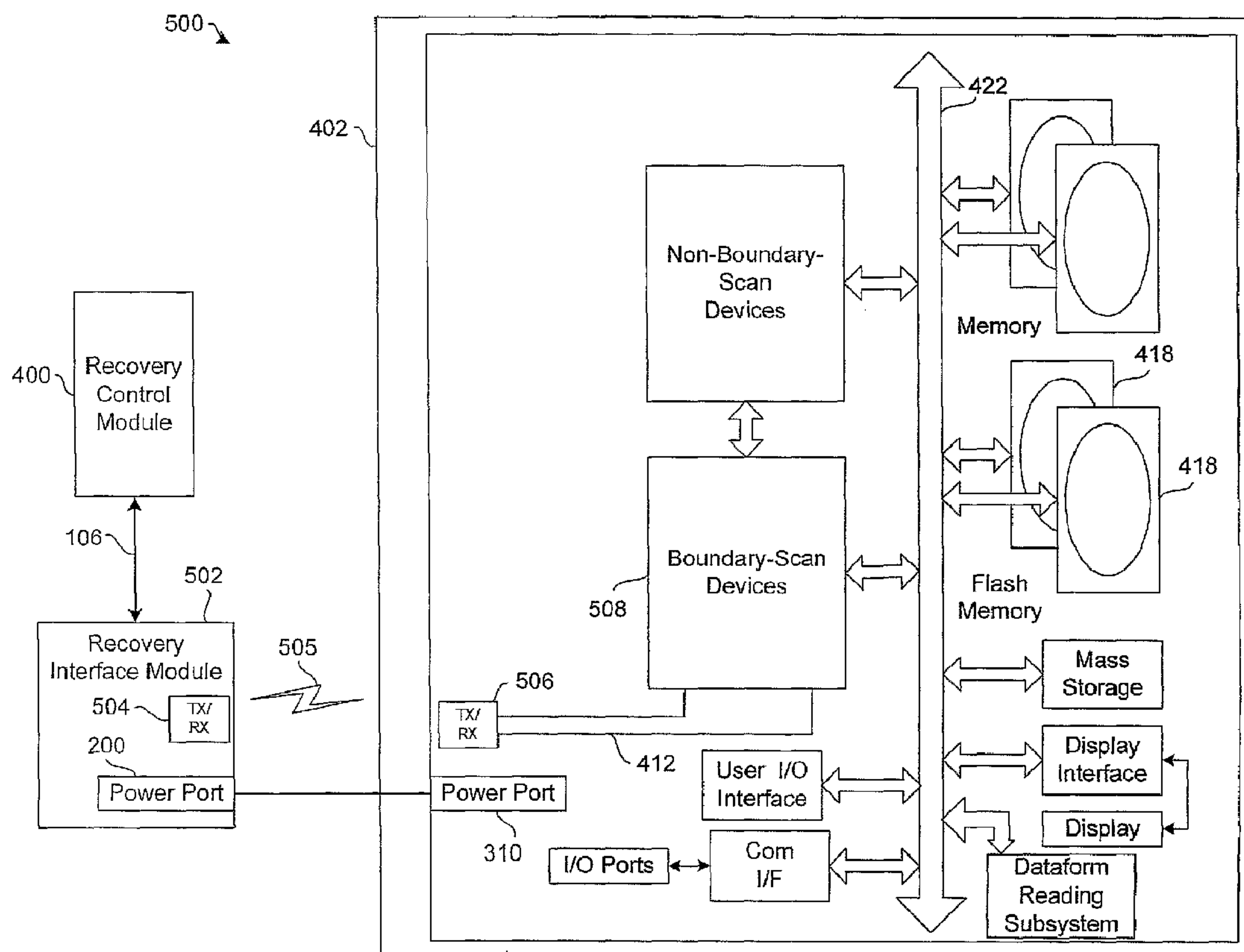




(86) Date de dépôt PCT/PCT Filing Date: 2003/11/10
(87) Date publication PCT/PCT Publication Date: 2004/06/17
(45) Date de délivrance/Issue Date: 2013/05/28
(85) Entrée phase nationale/National Entry: 2005/05/26
(86) N° demande PCT/PCT Application No.: US 2003/035764
(87) N° publication PCT/PCT Publication No.: 2004/051915
(30) Priorité/Priority: 2002/11/27 (US10/306,527)

(51) Cl.Int./Int.Cl. *H04L 1/22* (2006.01),
H02H 3/05 (2006.01)
(72) Inventeurs/Inventors:
STERN, DAVID, US;
WULFF, THOMAS, US;
DEMEO, DANA C., US
(73) Propriétaire/Owner:
SYMBOL TECHNOLOGIES, INC., US
(74) Agent: GOWLING LAFLEUR HENDERSON LLP

(54) Titre : PORT DE RECUPERATION APRES SINISTRE INFORMATIQUE DANS UN ORDINATEUR PORTABLE
(54) Title: DISASTER RECOVERY PORT IN A PORTABLE COMPUTER



(57) Abrégé/Abstract:

Architecture for updating corrupted instructions in a computing device. The computing device includes a housing (500) for enclosing a flash memory (418) for storing a boot loader routine, and at least one boundary-scan device (508) internal to the



(57) **Abrégé(suite)/Abstract(continued):**

computing device for processing the boot loader routine, and operatively disposed on a boundary-scan bus. The housing includes a boundary-scan port through which updated instructions are communicated and power provided. The port is concealed under a label, cover, or in a location that limits access by the device owner, such as the battery well.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property
Organization
International Bureau



(43) International Publication Date
17 June 2004 (17.06.2004)

PCT

(10) International Publication Number
WO 2004/051915 A1

(51) International Patent Classification⁷: **H04L 1/22**

(74) Agents: AMIN, Himanshu S. et al.; Amin & Turocy, LLP,
1900 E. 9th Street, 24th Floor, National City Center, Cleve-
land, OH 44114 (US).

(21) International Application Number:
PCT/US2003/035764

(22) International Filing Date:
10 November 2003 (10.11.2003)

(25) Filing Language: English

(26) Publication Language: English

(81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW.

(30) Priority Data:
10/306,527 27 November 2002 (27.11.2002) US

(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

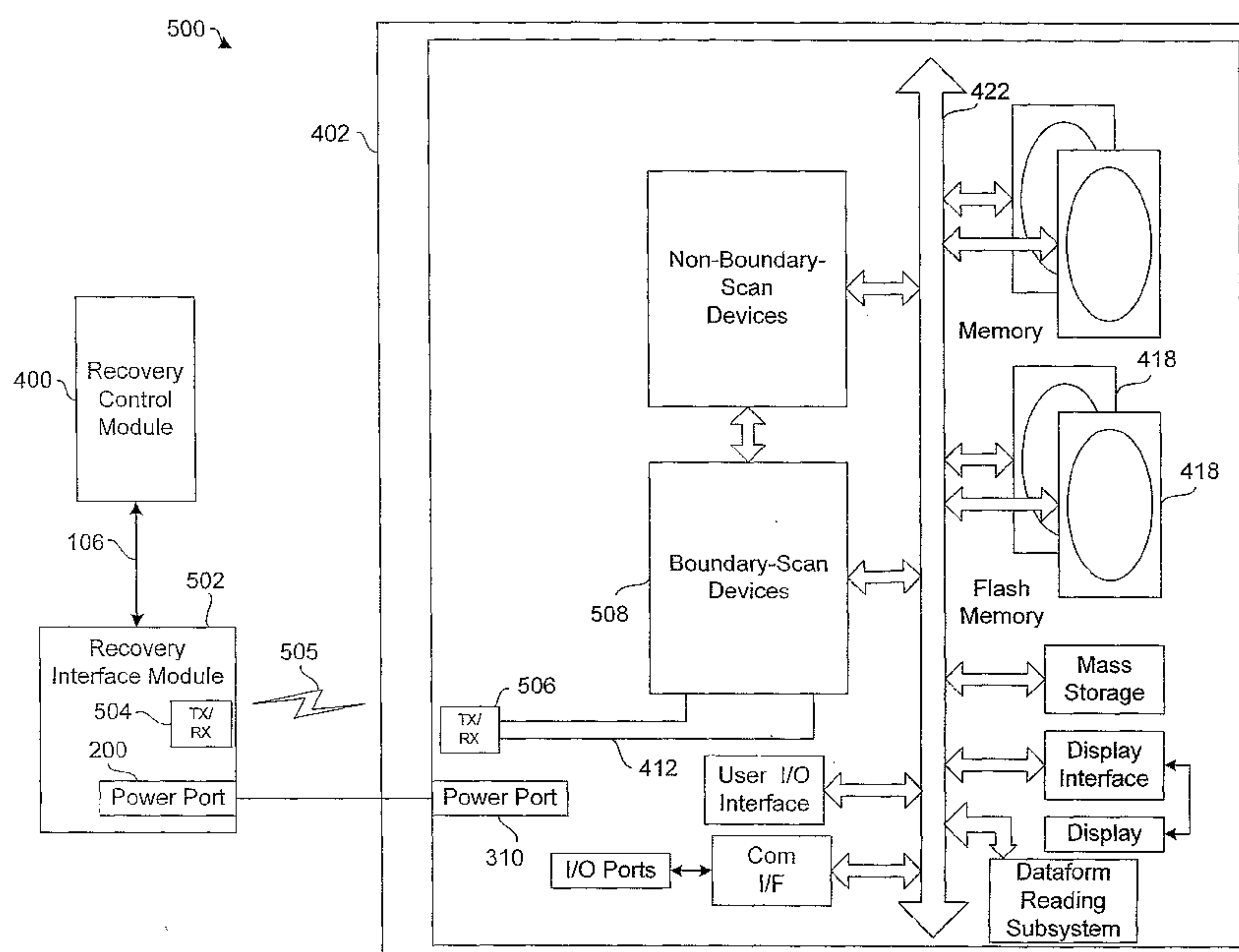
(71) Applicant: **SYMBOL TECHNOLOGIES, INC.**
[US/US]; One Symbol Plaza, Holtsville, NY 11742 (US).

(72) Inventors: **STERN, David**; 47 Jonas Boulevard, Cen-
tereach, NY 11720 (US). **WULFF, Thomas**; 25 Cheryl
Lane, North Patchogue, NY 11772 (US). **DEMEO, Dana**
C.; 16 Beaumont Lane, Lake Grove, NY 11755 (US).

Published:
— with international search report

[Continued on next page]

(54) Title: DISASTER RECOVERY PORT IN A PORTABLE COMPUTER



(57) Abstract: Architecture for updating corrupted instructions in a computing device. The computing device includes a housing (500) for enclosing a flash memory (418) for storing a boot loader routine, and at least one boundary-scan device (508) internal to the computing device for processing the boot loader routine, and operatively disposed on a boundary-scan bus. The housing includes a boundary-scan port through which updated instructions are communicated and power provided. The port is concealed under a label, cover, or in a location that limits access by the device owner, such as the battery well.

WO 2004/051915 A1

WO 2004/051915 A1



For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

DISASTER RECOVERY PORT IN A PORTABLE COMPUTER

5

TECHNICAL FIELD

This invention is related to JTAG (or boundary-scan) devices, and more specifically to accessing a JTAG device.

BACKGROUND OF THE INVENTION

10

Most portable computers contain a boot loader routine that facilitates software updates to the onboard flash memory. The boot loader itself also resides within the onboard flash memory. On a rare occasion, the software stored in flash memory inside a handheld portable computer can be catastrophically corrupted, such that the boot loader is also corrupted. This condition prevents an external host from updating the corrupted flash memory. Under normal circumstances the portable computer must be dismantled, and the printer circuit board (PCB) installed onto a test fixture in order to restore the boot loader. After the boot loader is restored, the remaining system software can be downloaded.

15

20

Additionally, there are typically programmable logic integrated circuits (ICs) on the PCB that can only be programmed via special programming pins. These pins are normally not externally accessible. Thus in order to update the programmable logic or recover from catastrophically corrupted software stored in the flash memory the product case must be opened.

25

What is needed is a method for providing accessibility to the special programming contacts from the outside of the computer case.

SUMMARY OF THE INVENTION

30

The present invention disclosed and claimed herein, in one aspect thereof, comprises architecture for updating corrupted instructions in a computing device. The computing device includes a housing for enclosing a flash memory for storing a boot loader routine, and at least one boundary-scan device internal to the computing device for processing the boot loader routine, and operatively disposed on a boundary-scan bus. The housing includes a boundary-scan port through which updated instructions

are communicated and power provided. The port is concealed under a label, cover, or in a location that limits access by the device owner, such as the battery well.

In another aspect thereof, there is provided architecture for programming a boundary-scan device in a computing device. The boundary-scan device disposed on a boundary-scan bus within a housing of the computing device. The boundary-scan bus is accessed through the housing, which is unopened, via a boundary-scan port to communicate program instructions thereacross to the boundary-scan device. An interface module is interfaced to the computing device via the boundary-scan port for providing power and communicating the program instructions thereto, the interface module interfacing to the computing device by at least one of in a battery well of the computing device and outside of the battery well.

In still another aspect thereof, there is provided a system for updating instructions in a computing device, the computing device comprising a housing for enclosing a memory for storing computer executable instructions, at least one boundary-scan device internal to the housing and operatively disposed on a boundary-scan bus for processing the computer executable instructions. The system comprises an interface module for interfacing to the computing device to communicate an updated version of the computer executable instructions thereacross for storing in the flash memory, and a control module in operative communication with the interface module. The control module can provide power to the computing device through the interface module while the instructions are being updated in the computing device. Alternatively, the interface module alone provides power to the computing device during the updating process by interfacing to the computing device via a boundary-scan port for providing power and communicating the program instructions thereto, the interface module interfacing to the computing device by at least one of in a battery well of the computing device and outside of the battery well.

To the accomplishment of the foregoing and related ends, certain illustrative aspects of the invention are described herein in connection with the following description and the annexed drawings. These aspects are indicative, however, of but a few of the various ways in which the principles of the invention may be employed and the present invention is intended to include all such aspects and their equivalents. Other advantages and novel features of the invention may become apparent from the

following detailed description of the invention when considered in conjunction with the drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

5 FIG. 1 illustrates an isometric of an implementation of the present invention in a recovery module that mates mechanically and electrically in a battery well of the recoverable computing device;

 FIG. 2 illustrates a front side isometric of the recovery module of FIG. 1;

 FIG. 3 illustrates an isometric of a recoverable device into which the module
10 fits;

 FIG. 4 illustrates a system block diagram comprising typical board of the computing device with boundary-scan components, the recovery module, and an external recovery control module; and

 FIG. 5 illustrates a system block diagram of a boundary-scan system
15 comprising a wireless boundary-scan bus connection.

 FIGS. 6A and 6B illustrate front and rear views of a portable terminal device in accordance with the subject invention.

DETAILED DESCRIPTION OF THE INVENTION

20 The present invention is now described with reference to the drawings, wherein like reference numerals are used to refer to like elements throughout. In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It may be evident, however, that the present invention may be practiced without these specific
25 details. In other instances, well-known structures and devices are shown in block diagram form in order to facilitate describing the present invention.

 The present invention solves the aforementioned problem by providing accessibility to the set of special programming contacts from outside of the computer case. This can be achieved by locating the pin contacts underneath a label, recess, or
30 other protective door. These placements would allow access to a set of signals by a technician that facilitate disaster recovery without opening the case of the computer.

Most portable computers contain a processor with JTAG (Joint Test Action Group) debugging capabilities. JTAG is a consortium of North American chip companies that developed onboard chip input/output testing capabilities such that one

5

10

15

20

25

or more chips of densely packed circuits could be tested through test access ports on a software-controlled basis. JTAG, also known as boundary-scan, was developed in 1980s, and standardized in 1990 as the now defined IEEE 1149.1 standard, allows full access to the internal registers of a digital device. Only five signal lines are needed in the boundary-scan interface to access the internal chip registers (i.e., TDI (test data in), TDO (test data out), TCK (test clock), TMS (test mode select), and TRST (test reset) in lieu of a power-up reset). Additionally, many programmable logic devices such as FPGAs (Field Programmable Gate Arrays) and CPLDs (Complex Programmable Logic Devices) use the JTAG architecture for programming and configuration. The JTAG also supports a daisy-chained topology. Thus all microprocessors and a number of other devices, such as the CPLDs, can be included in a single JTAG chain. In this way, the technician can access most, if not all, of the chips in the computer digital system via the JTAG contacts (i.e., the five JTAG pins plus ground) without opening the device case to access pins and other ports.

Referring initially to the figures, FIG. 1 illustrates an isometric of an implementation of the present invention in a recovery interface module 100 that mates mechanically and electrically in a battery well of the recoverable computing device. As illustrated from the back side, the interface module 100 includes a battery compartment 102 enclosed with a battery compartment cover 104. The battery compartment 102 is provided where the application is such that an external power source cannot be utilized. Where the recovery operation can utilize an external power source, there is also provided a power and signal cable 106 for conducting power and communicating signals therethrough to the recovery module 100, and further to the recoverable device (not shown). The cable 106 includes an interface connector 108 having a number of power and signal pins 110 for interfacing power and signals to the recoverable device. The cable 106 exiting the interface module 100 connects to a JTAG control module (not shown) to interface signals therebetween. Instead of providing battery power from batteries onboard the interface module 100, the interface module 100 can be configured to route battery power directly to the battery system of the recoverable device.

It is appreciated that where the batteries of the module 100 are rechargeable, the module 100 can include a recharging circuit for recharging the module batteries.

In this implementation, the cable 106 would then provide AC power to the charging circuit of the module 100. Alternatively, the charging circuit is in an external module such as the JTAG control module so that the batteries are charged therefrom. Where the module 100 includes the batteries, a computer service depot can more easily and quickly gain access to the internal digital logic for rewriting code in the programmable logic ICs without having to ensure that external power is made available. This configuration also facilitates operating in a field environment where the technician may need to go to the device instead of the device being brought to the depot for service.

Referring now to FIG. 2, there is illustrated a front side isometric of the recovery module 100 of FIG. 1. The module 100 includes a battery interface connector 200 for conducting power to the recoverable device. The module 100 also includes a JTAG interface pin connector 202. The pin connector 202 includes a number of pins 204 for contacting a mating JTAG pad connector of the recoverable device. The types of pins 204 utilized can be pogo-style pins, or any pin style that is compatible with the mating JTAG connector of the recovery device.

Referring now to FIG. 3, there is illustrated an isometric of a recoverable computing device 300 into which the module 100 fits. In this particular embodiment, the module JTAG connector 202 of the module 100 mates with a device JTAG connector 302 (also called the JTAG port or boundary-scan port) constructed in a port opening of the unopened housing and hidden underneath a label (or seal) 306 inside a battery well 308 of the recoverable device 300. In order to gain access to the JTAG circuits for updating or recovering the flash memory contents, the existing standard battery pack (not shown) is removed. The label 306 is then peeled back to expose access contacts 304 of the JTAG port connector 302 to the special disaster recovery module 100 that could, but is not required to, mate to the structure of the battery well 308. Thus the housing of the computing device does not need to be opened. The JTAG port connector 302 extends through the module housing exposing the access contacts 304. It is appreciated that the module 100, when conforming to the battery well structure 308, could also be adapted to provide battery power to the recoverable device 300 via the battery contacts 200 to mating device battery contacts 310 when properly seated into the battery compartment 308.

Alternatively, the device JTAG connector 302 may be concealed or made relatively inaccessible to the device owner by way of a screw-down hatch or cover 312 located outside of the battery well 308, which also may then be further concealed under a label (or seal) 314.

5 Referring now to FIG. 4, there is illustrated a system block diagram comprising typical board of the computing device 300 with boundary-scan components, the recovery module 100, and an external recovery control module 400. The computing device 300 includes a housing 402 that encloses a system board 404. The system board 404 manufactured with boundary-scan devices, includes the
10 following main components: one or more boundary-scan CPLDs 406, one or more FPGAs 408, and one or more processors 410 daisy-chained via the boundary-scan path 412. There are also included non-boundary-scan components 414 (also denoted as clusters), a first type of memory component 416, and a second type of memory component, i.e., flash memory 418. Other devices such as series resistors or buffers
15 are grouped under a block called transparent components 420. The device 300 utilizes a system bus 422 over which most, if not all, of the internal components of the computing device 300 communicate signals and data.

The device 300 can comprise a number of other components, including but not limited to: a mass storage memory 424 for storing large amounts of data and/or
20 computer executable instructions, and that communicates over the bus 422; a display interface 426 for interfacing a display 428 to one or more of the system processing devices via the bus 422 to provide a visual interface to the user; a dataform reading subsystem 430 (e.g., an optical scanner for reading bar codes, a magnetic reader for reading magnetic dataforms, an RF system for reading a transponder, etc.) and
25 communicating such dataform information over the bus 422; a user I/O interface 432 such as a keyboard, keypad, or other conventional user I/O mechanisms via which the user enters information to the bus 422; and, a communication interface 434 for communicating signals and data from the bus 422 to one or more ports 436 that facilitate external port communication utilizing architectures such as USB (Universal
30 Serial Bus), IEEE 1394, CardBus, PCMCIA, etc.

The chained boundary-scan devices (406, 408, and 410) interconnected by the boundary-scan bus 412 of the system board 404 can also receive additional data I/O

via internal edge-connector ports (440 and 442) over internal interconnect lines 438. The internal interconnect lines 438 also eventually connect to the bus 422. Note that the cluster devices 414 and transparent devices 420 can also be interrogated along with the other devices by imposing and analyzing data on devices on either side thereof.

The boundary-scan bus 412 is made internally accessible from the system board 404 via a system board JTPA port 444. The boundary-scan bus 412 is then extended to the housing 402 to the battery well connector 302. This can be accomplished by flexible circuit or cable 446 having suitable bandwidth and power capabilities, or even by foregoing the cable 446 such that the port 444 is directly interfaceable to the port 204. Once the well connector 302 (or port 444 where made directly accessible) is uncovered, the recovery module 100 can be inserted into the battery well 308 such that well connector 302 can mate with the module connector 204. The recovery control module 400 then connects to the recovery module 100 via the cable 106 to couple power thereto, if needed, and communicate data and signals therebetween. It is appreciated that the control module 400 and interface module 100 can be combined into a single unit.

The disaster recovery data communicated to the flash memory 418 includes an updated version of the boot loader routine instructions that were corrupted. The updated version is communicated to overwrite the corrupted version currently existing in the flash memory 418. Note that the programmable boundary-scan devices (406, 408, and 410) can also be reprogrammed across the boundary-scan bus via the port 302. Such programmable devices (406, 408, and 410) may have associated therewith an internal memory that stores the programmable instructions for reprogramming of the devices (406, 408, and 410).

The nature of the computing device 300 can include a wide variety of such devices, in that an enormous number of electronic devices now include a processor of some sort, and which processors are typically designed with the boundary-scan architecture for efficient and reliable production testing before the consumer receives the device. Thus the computing device 300 includes, but is not limited to, cellular telephones, pagers, portable computers, PDAs, scanning devices, etc., and virtually

any device including the boundary-scan architecture. All that needs to be done is to provide external access to the boundary-scan bus 412 through the housing port 302.

Referring now to FIG. 5, there is illustrated a system block diagram of a boundary-scan system 500 comprising a wireless boundary-scan bus connection. The primary purpose of the labels (306 and 314) and/or the cover 312 is to discourage access to the JTAG or boundary-scan port 302 by the device owner through concealment techniques. In furtherance thereof, numerous other techniques could be made available to preclude access by the device owner to the internal boundary-scan architecture. In this particular embodiment, a recovery interface module 502 is adapted to include a wireless transceiver 504 communicating signals and data over a wireless link 505 to a compatible wireless transceiver 506 internal to the computing device housing 402.

The internal wireless transceiver 506 terminates to the boundary-scan bus 412 such that the data and signals can be communicated over the bus 412 to boundary scan devices 508. Thus physical JTAG ports 302 and/or 444 of FIG. 4 are no longer needed. The bandwidth power of the wireless transceiver pair (504 and 506) could be adjusted such that the recovery interface module 502 must be brought into close proximity to the housing 402 to ensure reliable communication therebetween. Any DC power that may still be required for powering the computing device can be provided through the battery ports (200 and 310), when the interface module 502 is designed to be utilized in the computing device battery well 308. Otherwise, a separate power connection can be provided between the computing device and the interface module 502. Thus the disaster recovery signals, including, but not limited to, the updated boot loader routine, are wirelessly communicated from the interface module 502 through the wireless link 505 to the boundary-scan bus 412, and ultimately across the system bus 422 to the flash memory 418. The computing device may include all or some number of components further described in detail in FIG. 4.

What has been described above includes examples of the present invention. It is, of course, not possible to describe every conceivable combination of components or methodologies for purposes of describing the present invention, but one of ordinary skill in the art may recognize that many further combinations and permutations of the present invention are possible. Accordingly, the present invention is intended to

embrace all such alterations, modifications and variations that fall within the spirit and scope of the appended claims. Furthermore, to the extent that the term “includes” is used in either the detailed description or the claims, such term is intended to be inclusive in a manner similar to the term “comprising” as “comprising” is interpreted
5 when employed as a transitional word in a claim.

What is claimed is:

1. A method of facilitating disaster recovery in a computing device, comprising:
receiving the computing device that includes a bar code scanner;
storing program instructions in a memory internal to the computing device;
processing the program instructions with at least one boundary-scan processing device
operatively disposed on a boundary-scan bus; and
accessing the boundary-scan bus through contacts, at least a portion of which are
accessible external to a housing of the computing device, the contacts facilitating operative
coupling to the boundary-scan bus to communicate an updated version of the program
instructions over the boundary-scan bus thereacross for storing in the memory when the program
instructions become corrupted.
2. The method of claim 1, further comprising the step of providing an interface recovery
module that is external to the housing of the computing device that communicates the updated
version of the program instructions in the step of accessing.
3. The method of claim 2, the interface recovery module is configured to fit into a battery
well of the computing device.
4. The method of claim 2, the interface recovery module is configured to fit into a battery
well of the computing device to provide power to the computing device and to communicate the
updated version of the program instructions.
5. The method of claim 1, the boundary-scan bus is accessed in the step of accessing
through a boundary-scan port in the housing.
6. The method of claim 5, the boundary-scan port is concealed by at least one of a label and
a cover to conceal the location of the port on the computing device.
7. The method of claim 1, the memory is a flash memory that receives the updated version
of the program instructions.

8. The method of claim 1, the memory is associated with the at least one boundary-scan device such that the updated version of the program instructions is communicated across the boundary-scan bus to reprogram the at least one boundary-scan device.
9. The method of claim 1, the program instructions include a boot loader routine stored in the memory, which memory is a flash memory.
10. A method of updating corrupted instructions in a computing device, comprising:
 - receiving the computing device that includes a bar code scanner;
 - storing a boot loader routine in a flash memory internal to a housing of the computing device;
 - processing the boot loader routine with at least one boundary-scan device internal to the housing and operatively disposed on a boundary-scan bus; and
 - in response to the boot loader routine stored in the flash memory of the computing device becoming corrupted, accessing the boundary-scan bus through contacts, at least a portion of which are accessible external to the housing of the computing device, to communicate an updated version of the boot loader routine thereacross for storing in the flash memory;
 - whereby the computing device becomes operational.
11. The method of claim 10, further comprising the step of providing an interface recovery module that is external to the housing of the computing device that communicates the updated version of the boot loader routine in the step of accessing.
12. The method of claim 10, the boundary-scan bus is accessed in the step of accessing through a boundary-scan port in the housing.
13. The method of claim 10, the memory is a flash memory that receives the updated version of the boot loader routine.
14. A disaster recovery apparatus for a computing device, comprising:
 - a bar code reader;
 - a memory internal to the computing device for storing instructions;
 - at least one boundary-scan processing device operatively disposed on a boundary-scan

bus for processing the instructions; and

a housing for enclosing the memory and at least one boundary-scan device, the boundary-scan bus accessed through the housing to communicate an updated version of the instructions thereacross for storing in the memory when the instructions become corrupted.

15. The apparatus of claim 14, further comprising an interface recovery module that is external to the housing of the computing device that communicates the updated version of the instructions.

16. The apparatus of claim 15, the interface recovery module is configured to fit into a battery well of the computing device.

17. The apparatus of claim 15, the interface recovery module is configured to fit into a battery well of the computing device to provide power to the computing device and to communicate the updated version of the instructions.

18. The apparatus of claim 14, the boundary-scan bus is accessed through a boundary-scan port in the housing.

19. The apparatus of claim 18, the boundary-scan port is concealed by at least one of a label and a cover to conceal the location of the port on the computing device.

20. The apparatus of claim 14, the memory is a flash memory that receives the updated version of the instructions.

21. The apparatus of claim 14, the memory is associated with the at least one boundary-scan device such that the updated version of the instructions is communicated across the boundary-scan bus to reprogram the at least one boundary scan device.

22. The apparatus of claim 14, the instructions include a boot loader routine stored in the memory, which memory is a flash memory.

23. An apparatus for updating corrupted instructions in a computing device, the apparatus comprising:

a bar code reader;
a flash memory internal to the computing device for storing a boot loader routine;
at least one boundary-scan device internal to the computing device and operatively disposed on a boundary-scan bus for processing the boot loader routine; and
a housing of the computing device for enclosing the memory and the at least one boundary-scan device, the boundary-scan bus accessed through the housing to communicate an updated version of the boot loader routine thereacross for storing in the flash memory.

24. The apparatus of claim 23, further comprising an interface recovery module that is external to the housing of the computing device for communicating the updated version of the boot loader routine.

25. The apparatus of claim 23, the boundary-scan bus is accessed through a boundary-scan port in the housing.

26. The apparatus of claim 23, the flash memory receives the updated version of the boot loader routine.

27. A method of programming a boundary-scan device in a computing device, comprising:
receiving the computing device that includes a bar code scanner;
providing the boundary-scan device disposed on a boundary-scan bus within a housing of the computing device; and
accessing the boundary-scan bus through the housing, which is unopened, via a boundary-scan port to communicate an updated version of program instructions thereacross to the boundary-scan device when the program instructions become corrupted.

28. The method of claim 27, further comprising the step of interfacing an interface module to the computing device via the boundary-scan port for providing power and communicating the program instructions thereto, the interface module interfacing to the computing device by at least one of in a battery well of the computing device and outside of the battery well.

29. A system for updating instructions in a computing device, the computing device comprising a bar code reader for facilitating collection of scanned data, a housing for enclosing a memory for storing computer executable instructions, at least one boundary-scan device internal to the housing and operatively disposed on a boundary-scan bus for processing the computer executable instructions, the system comprising:

an interface module for interfacing to the computing device to communicate an updated version of the computer executable instructions thereacross for storing in the flash memory when the computer executable instructions become corrupted; and

control module in operative communication with the interface module.

30. The system of claim 29, the control module providing power to the computing device through the interface module while the computer executable instructions are being updated in the computing device.

31. The system of claim 29, the interface module providing power to the computing device during the updating process.

32. The system of claim 29, the interface module interfaced to the computing device via a boundary-scan port for providing power and communicating the computer executable instructions thereto, the interface module interfacing to the computing device by at least one of in a battery well of the computing device and outside of the battery well.

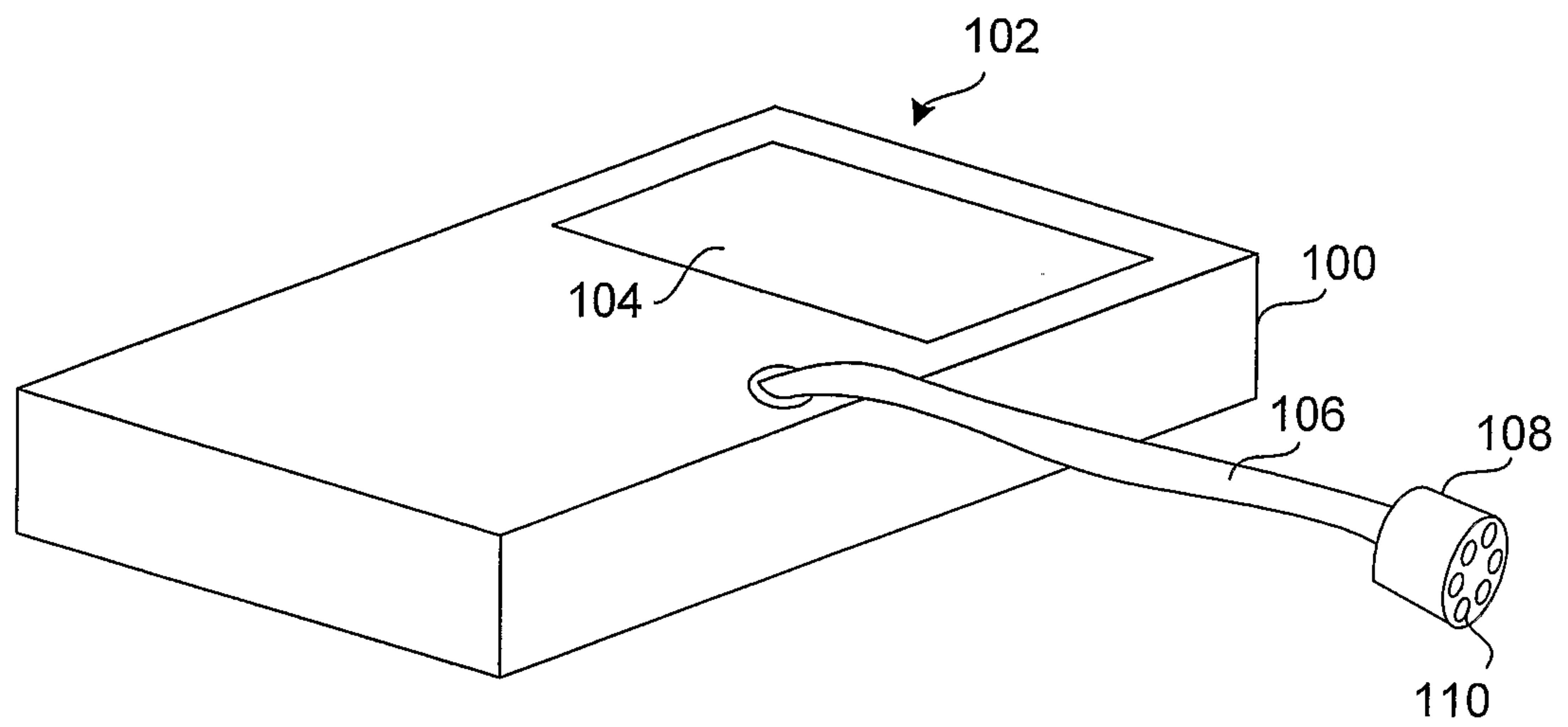


FIG. 1

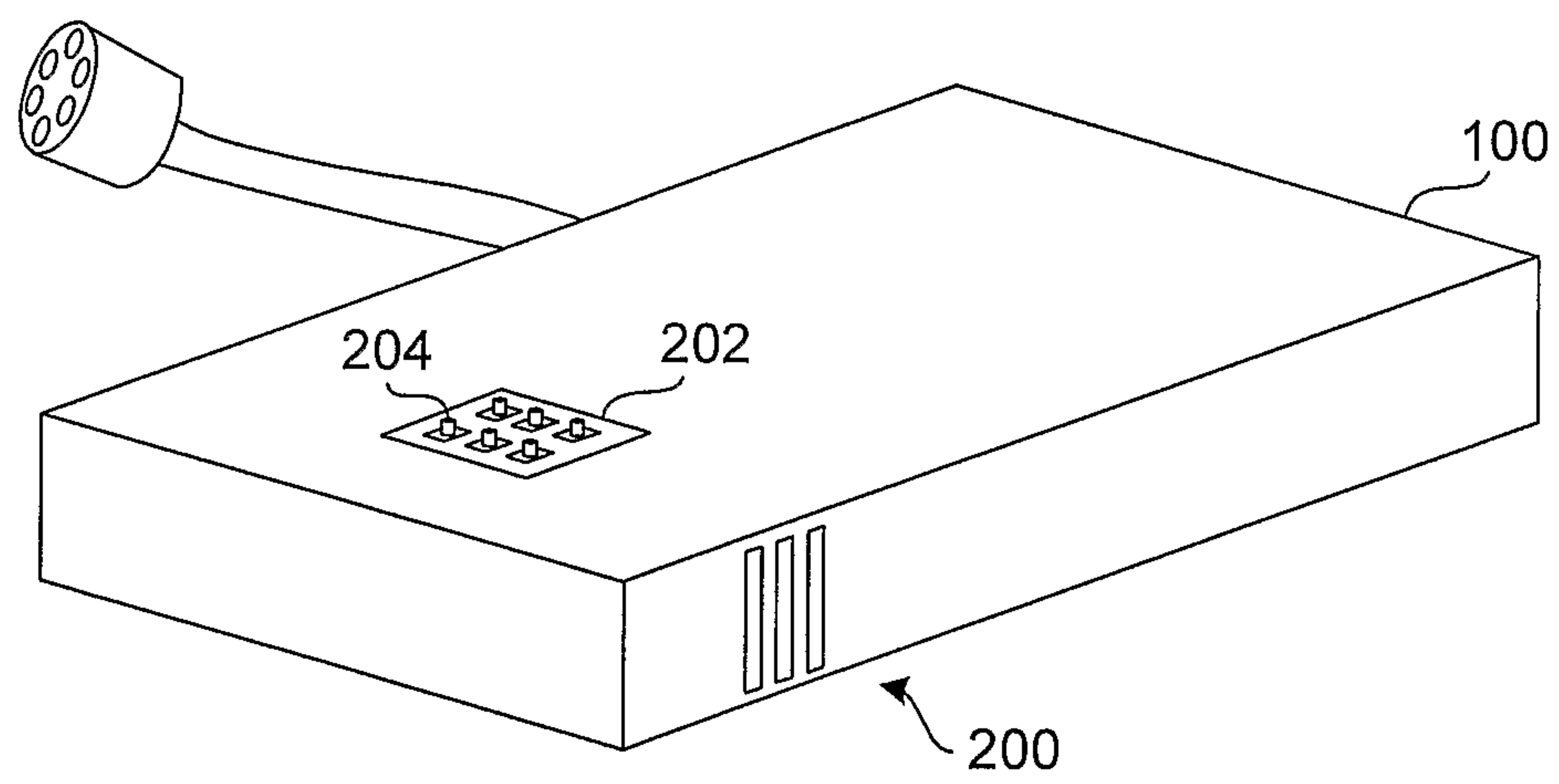


FIG. 2

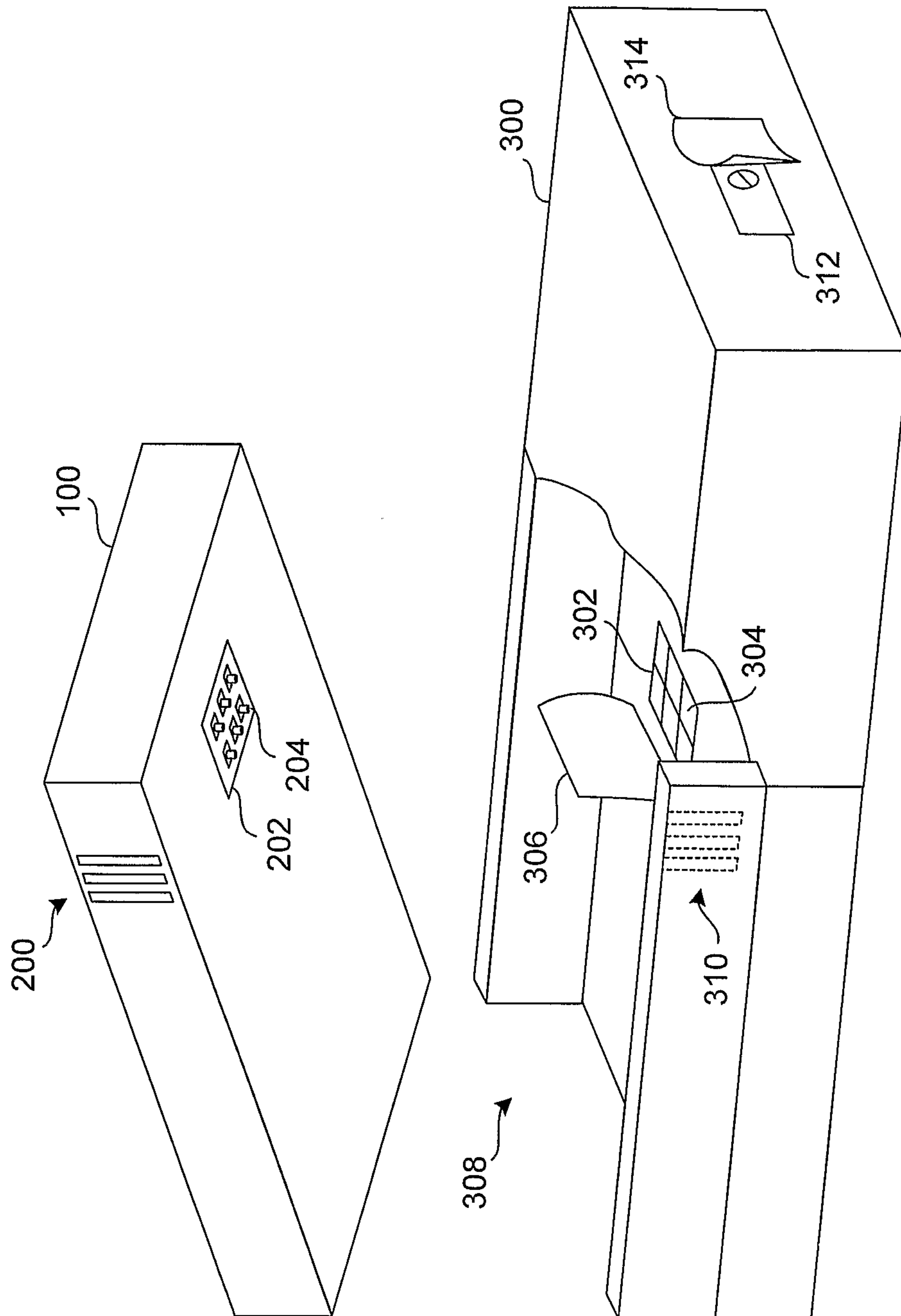
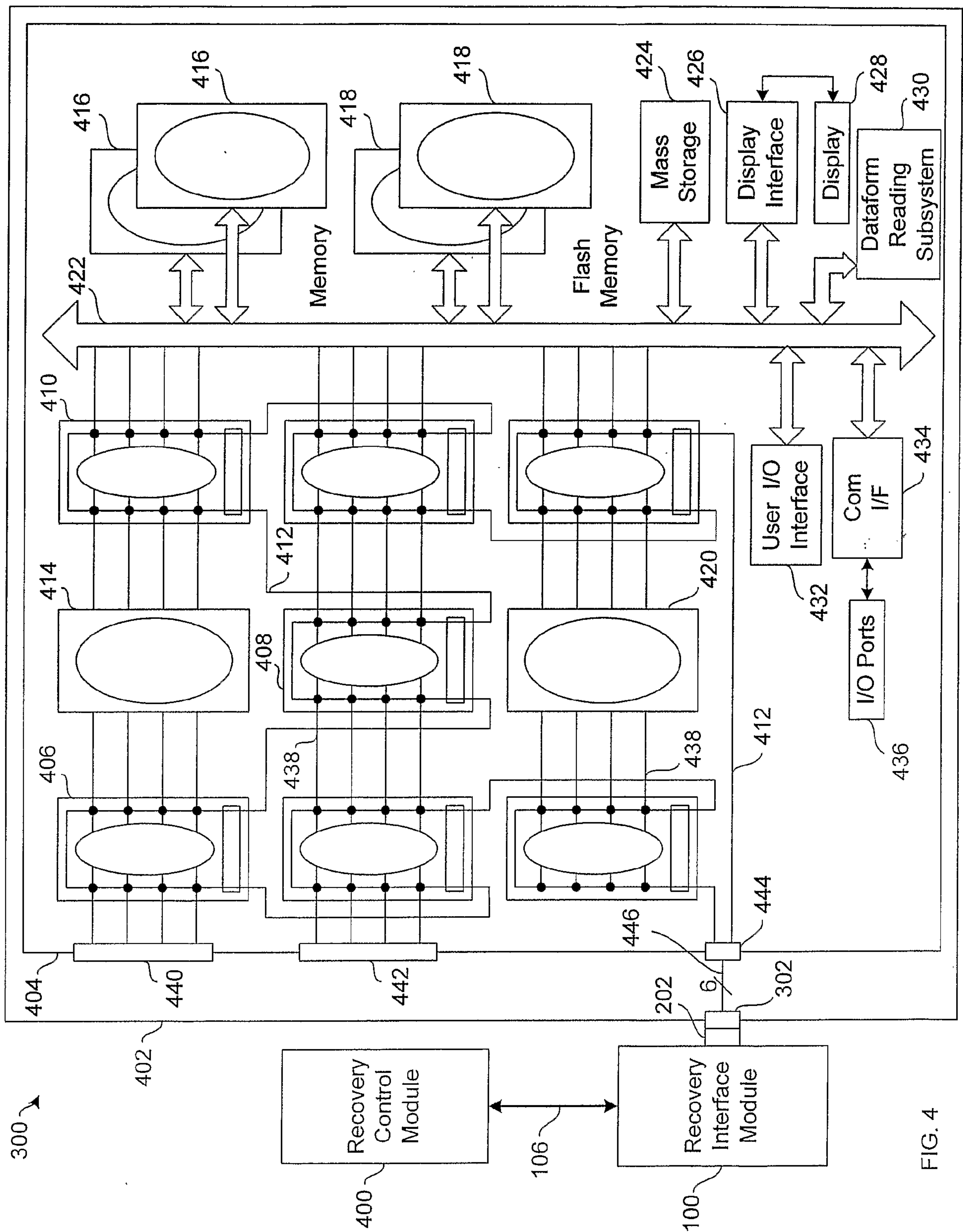


FIG. 3



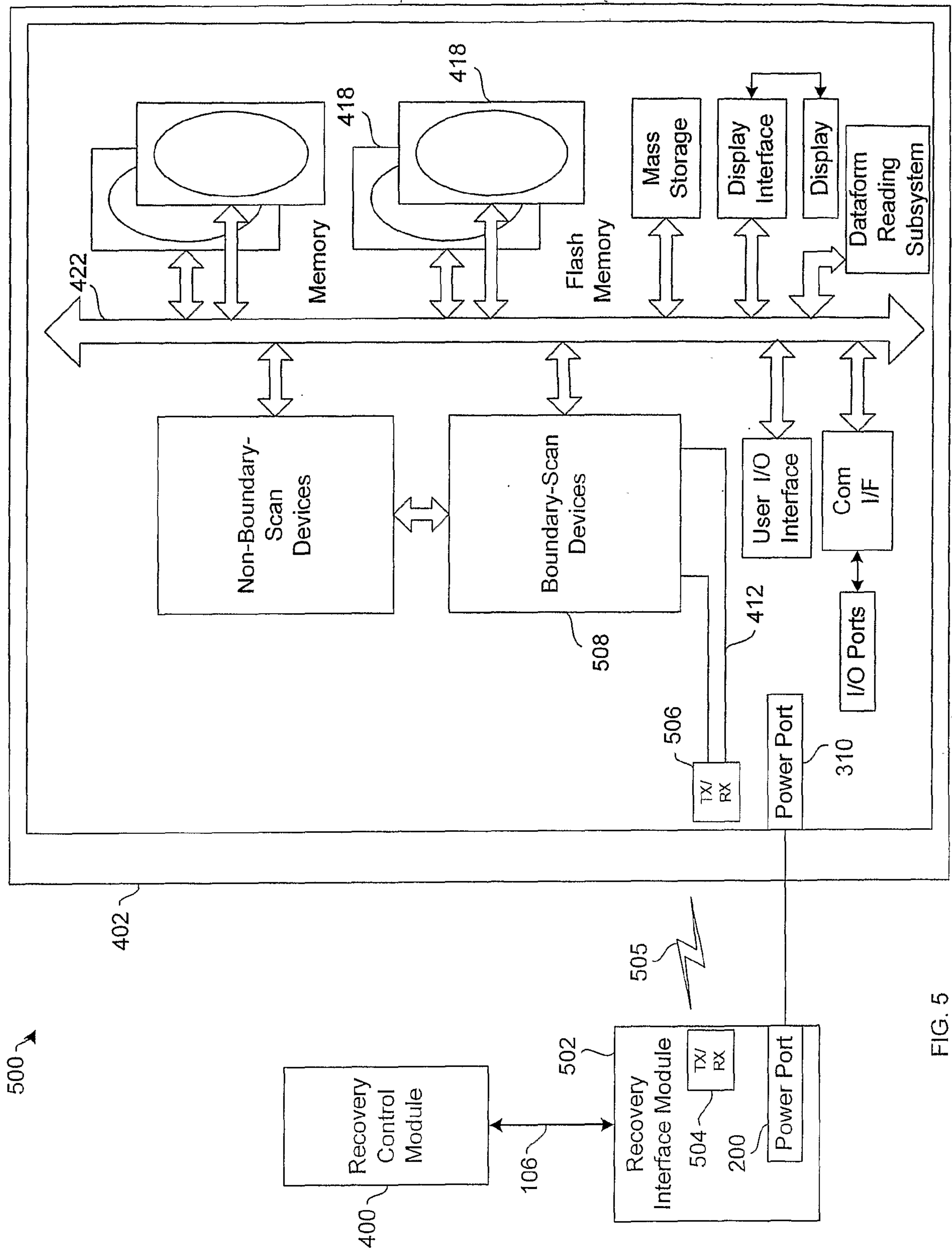


FIG. 5

5/6

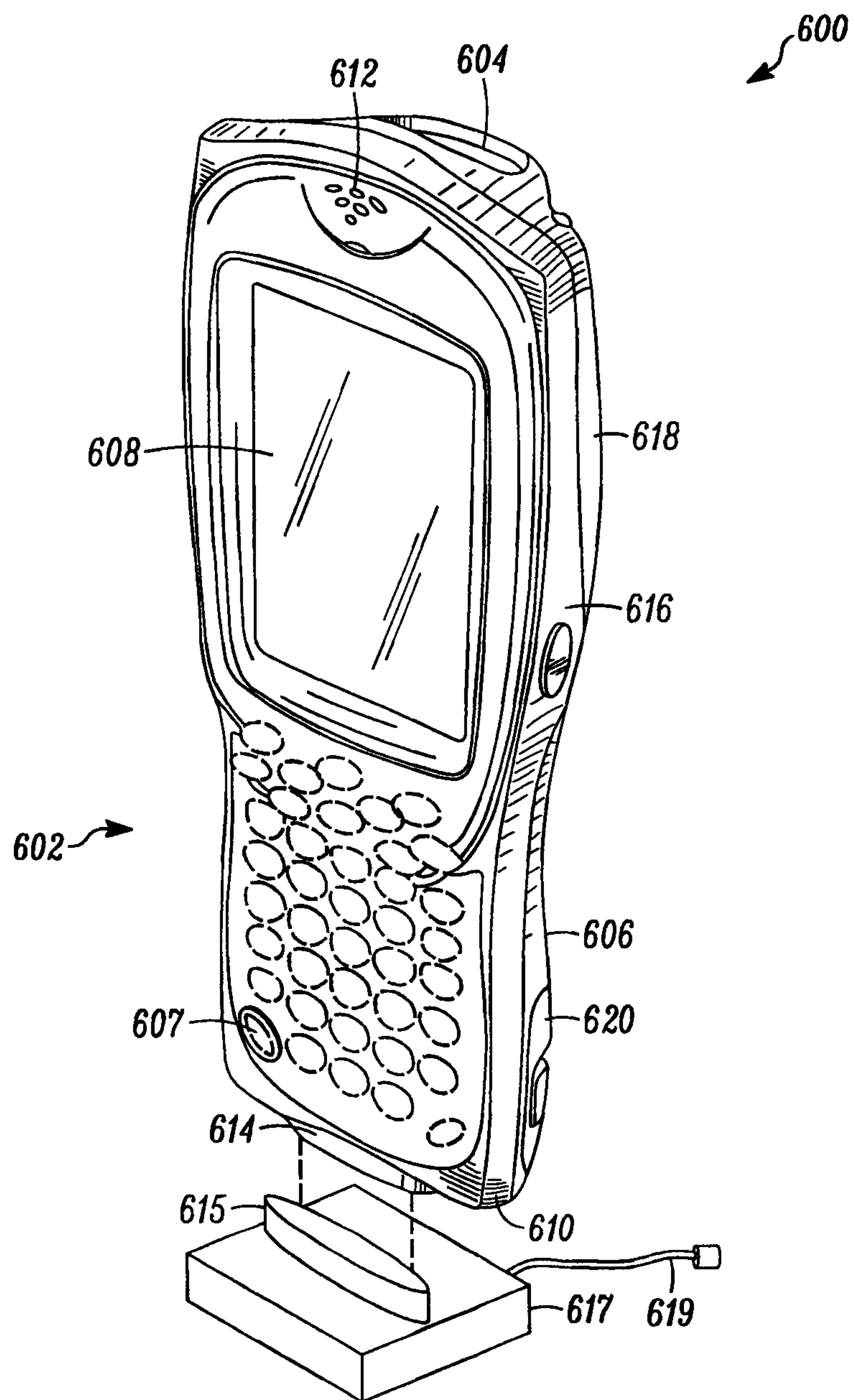


FIG. 6a

6/6

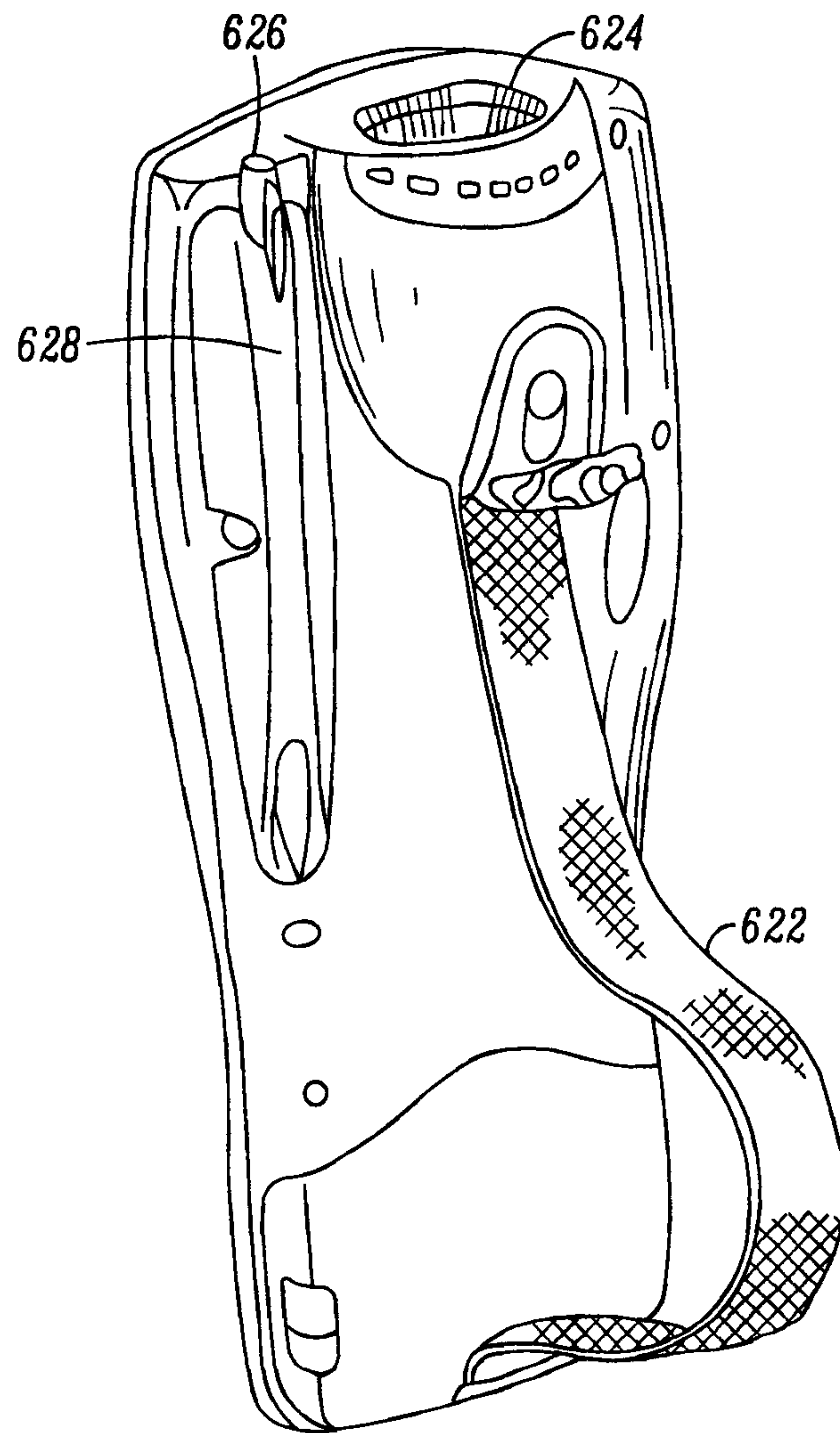


FIG. 6b

