WIRELESS ELECTRONIC-CONTROL SYSTEM

CLAIM OF PRIORITY

[0000] This application claims the priority benefit to U.S. Patent Application Serial No. 62/770,587, filed on 21 November 2018, and entitled "WIRELESS ELECTRONIC-CONTROL SYSTEM," which is incorporated by reference herein in its entirety.

TECHNICAL FIELD

[0001] The subject matter disclosed herein relates to testing of various production tools used in the semiconductor and related industries. More specifically, the disclosed subject matter relates to a configurable "test plug" form factor input/output (I/O) control board with purposes including quick system testing, diagnostics, and troubleshooting of one or more production tools.

BACKGROUND

[0002] Various tools used in the semiconductor and related industries use a number of different types of electronic-control system. These electronic-control systems bridge the gap between software and the hardware used for substrate (e.g., silicon wafer) processing. The various tools include, for example, process tools (such as etch tools, deposition tools, and cleaning tools), metrology tools (such as substrate scanners and various types of inspection tools), and other types of tools used in the used in the semiconductor and related industries. This variety of tools is simply referred to as "production tools" herein for brevity, although various types of related research and development (R&D) and other off-line and in-situ tools are included as a part of the term "production tool" as well.

[0003] For each production tool, custom printed-circuit boards (PCBs) are developed to support routing of signals from I/O controllers, safety interlock devices, power distribution, support circuitry for external devices, and other functions. In addition, there

are normally multiple versions of each custom PCB developed due to changes in specifications of the tools, problems fixed (necessitating a new PCB), etc.

[0004] The normal process for a controls group when a new PCB is ordered and received is to manually bench test and verify the functionality of the entire PCB. Depending on the complexity of the PCB, the testing procedure may take one or more engineers anywhere from an hour to several days to complete.

[0005] Contemporaneous testing includes manually moving jumpers between connectors on the PCBs to read or activate an I/O as shown in FIG. 1A. For example, FIG. 1A shows a first node 110 (node 0), a second node 120 (node 1), and a third node 130 (node 2). Each of these nodes is used to connect to various ports on a PCB manually. More or fewer nodes may be needed depending an a level of complexity of a connected PCB. The higher the number of nodes, the more complexity in wiring to the PCB and the more possibility of errors in making connections. Further, as noted above, many tools have safety-interlock devices that must also be tested for proper functionality. Safety-interlock devices are generally programmable and therefore quickly updated. However, the safety-interlock devices often require manual jumper-based testing after every change.

[0006] FIG. 1B shows custom test plugs 150 produced by a manufacturer of PCBs to test specific nodes of a tool. However, these custom test plugs are expensive. Significantly, these custom test plugs provide no automated testing capabilities, so all tests must be run manually. Also, once a PCB or tool is modified, a new test plug must be constructed and provided by the manufacturer.

[0007] FIG. 1C shows one test platform 170 that uses existing I/O controllers (IO Cont. Node 0, IO Cont. Node 1, . . . IO Cont. Node x) to test various nodes. However, for each tool, one or more custom PCBs

and/or cables must be constructed to connect the system to various test nodes (Test IO Cont. Node x . . . Test IO Cont. Node 1, Test IO Cont. Node 0). Therefore, the test platform of FIG. 1C still requires custom-made PCBs for every type of I/O controller used in a tool.

[0008] In addition to hardware verification, software has to be developed and tested. Much of the software development is done with simulated I/O. Simulation of the IO allows the software developer to test many aspects of the software but does not allow for testing of the low-level communication required when communicating with hardware. Many new problems are often discovered by the software developers once they get access to the real hardware and they are no longer simulating I/O of the tool. Testing an actual tool can result in delays to system integration and the rest of the project since much of this testing has to wait until the tool is built. Moreover, current software-simulation models cannot always be programmed to capture actual problems encountered by the tools such as system lockups and substrate breakage scenarios. When the software is run on the actual tool, the tool is taken out-of-production and can shut down an entire process line (if already installed in a fabrication facility). Also, the software testing often requires the scrapping of test substrates or, in the case of deposition tools, a waste of expensive precursor gases and/or precursor liquids.

[0009] The information described in this section is provided to offer the skilled artisan a context for the following disclosed subject matter and should not be considered as admitted prior art.

BRIEF DESCRIPTION OF THE FIGURES

[00010] FIGS. 1A-1C show examples of various types of hardware and/or software test systems currently in use;

[00011] FIG. 2 shows an exemplary embodiment of a configurable test-plug hardware-platform, according to various exemplary embodiments;

[00012] FIG. 3 shows an exemplary embodiment of a wireless mesh network that can be used with the disclosed subject matter, according to various exemplary embodiments;

[00013] FIG. 4 shows an exemplary embodiment of an implementation of the test-plug hardware-platform when used with, for example, a production tool, according to various exemplary embodiments; and

[00014] FIG. 5 shows a simplified block diagram of a machine in an example form of a computing system within which a set of instructions for causing the machine to perform any one or more of the methodologies and operations discussed herein may be executed.

DETAILED DESCRIPTION

[00015] The disclosed subject matter will now be described in detail with reference to a few general and specific embodiments as illustrated in various ones of the accompanying drawings. In the following description, numerous specific details are set forth in order to provide a thorough understanding of the disclosed subject matter. It will be apparent, however, to one skilled in the art, that the disclosed subject matter may be practiced without some or all of these specific details. In other instances, well-known process steps or structures have not been described in detail so as not to obscure the disclosed subject matter.

[00016] The disclosed subject matter contained herein described relates generally to operations and testing of production tools and electrical components thereof. Such production tools can include various types of deposition (including plasma-based tools such as atomic layer deposition (ALD), chemical vapor deposition (CVD), plasma-enhanced CVD (PECVD), and so on, as well as etching tools (e.g., reactive-ion etching (RIE) tools), and various types of thermal furnaces (e.g., such as rapid thermal annealing (RTA) and oxidation), ion implantation, and a variety of other process and metrology tools found in various fabs and known to a person of ordinary skill in the art. However, the disclosed subject matter is not limited to semiconductor environments and can be used in a number of machine-tool environments such as robotic assembly, manufacturing, and machining environments.

[00017] For ease in understanding the disclosed subject matter, various examples will be provided with reference to a semiconductor fabrication production tool. However, upon reading and understanding the disclosure provided herein, the person of ordinary skill in the art will recognize that various embodiments of the test-plug hardware-platform of the present disclosure can be used in a

wide variety of environments where various types of tool or tools are used.

[00018] In general, various embodiments of the disclosed subject matter comprise a configurable "test plug" form factor input/output (I/O) control board that can be used for, for example, quick system testing, diagnostics, and troubleshooting of production tools and associated printed-circuit boards (PCBs) and other electrical components.

[00019] Therefore, the disclosed subject matter provides a way to test quickly and substantially simultaneously the functionality of every standard and custom PCB developed for, for example, a production tool. With various embodiments of the disclosed subject matter, in order to test one or more PCBs on a production tool, a test plug is coupled from one or more of the test-plug hardware-platforms described herein to each connector on the PCB. The test engineer then uses software to configure the test plug I/O and perform tests as needed. Once configuration is complete, the engineer can start the automated testing. The configuration can be saved for faster testing of other similar or identical PCBs if needed in the future.

[00020] Once set up, multiple PCBs of the same type (or even a similar type with modifications as needed) can also be tested quickly and substantially simultaneously. Further, software will be able to test low-level communication with actual (as opposed to simulated) hardware (e.g., to the PCB while still mounted within the production tool). The test engineer can use the software to do this testing on the bench and solve problems before system integration. Therefore, the subject matter disclosed herein describes a configurable and flexible test-plug hardware-platform, comprising an I/O board and wireless electronic control system as described herein. Further, each of the arrangements and configurations of various embodiments of the test-plug hardware-platform is defined in greater detail below.

[00021] For example, with reference now to FIG. 2, an example of a configurable test-plug hardware-platform 200 (a configurable control system including, for example, at least one input/output (I/O) connector and a wireless-mesh network) according to various exemplary embodiments is shown. Computer software may be used to monitor inputs and command outputs on the test-plug hardware-platform, as described in more detail below. A configuration of the computer software may be based on an existing signal list and interlock table for a particular production tool. In various embodiments, the signal list and interlock tables may be directly imported into the software from a predefined table. A software-based user interface (UI) allows for I/O status checking and manipulation. The UI can also run automated I/O tests to verify the functionality of the hardware being tested. Overall, the test-plug hardware-platform 200 can be used for quick system-testing, system-diagnostics, and troubleshooting of various PCBs for a production tool or the production tool itself. The test-plug hardware-platform 200 therefore provides a robust, reliable, and quickly configurable diagnostic and validation system for a production tool and related electrical components.

[00022] Although the configurable test-plug hardware-platform 200 of FIG. 2 shows only a single male connector 210 and a single female connector 230 to avoid obfuscating the disclosed subject matter, in various embodiments, the test-plug hardware-platform 200 can include any number of male and female connectors. Also, the single male connector 210 and the single female connector 230 may be reversed in orientation compared with the orientation shown in FIG. 2.

[00023] For example, in various embodiments, the test-plug hardware-platform 200 can include multiple types of common electrical-connector types such as D-type subminiature ("DSUB")

electrical connectors to interface with various ones of the electrical PCBs and other electrical components found on a production tool. DSUB electrical connectors are known to a person of ordinary skill in the art and include, for example, DA-15 connectors, DB-25 connectors, DC-37 connectors, DD-50 connectors, DE-9 connectors, DB13W3 connectors, and a number of other electrical connector types used in the industry. Such connectors can support various 9-, 15-, 25-, 37-, and 50-pin standard density DSUB connectors as well as 15-, 26-, 44-, 62-, and 78-pin high-density DSUB connectors. Additionally, certain types of DSUB connectors may also include, for example, coaxial connectors used to support high frequency devices (e.g., RF devices) and video-signal connections.

[00024] Additionally, in various embodiments, the test-plug hardware-platform 200 can include other common electrical-connector types such as IEEE 1394 (FireWire®), Thunderbolt®, Universal Serial Bus (USB®), and other connector types known in the art. The test-plug hardware-platform 200 can also include various types of relays, opto-couplers, safety interlocks, and other electrical components. These other electrical components may also be configurable by software.

[00025] Regardless of the type of electrical connector used, each of the pins on all of the electrical connectors are separately configurable, via software (and/or hardware in certain configurations), such as supplying or receiving power, ground, various types of analog and digital signals (e.g., AI, AO, DI, or DO) and various other types of signals known in the art (e.g., video signals for various types of metrology tools), with all tested substantially simultaneously.

[00026] Consequently, the test-plug hardware-platform 200 can readily be configured to work with a number of different types of production tools (and related PCBs and other electrical components) without the laborious rewiring (including breadboarding and other

techniques involving the use of jumper wires for electrical reconnections as shown in FIG. 1A), or the fabrication of tool-specific boards for a particular production tool only (as shown in FIG. 1B), as are used under the prior art. Therefore, various embodiments of the test-plug hardware-platform 200 allows the testing of, for example, input output controller (IOC) outputs (e.g., AO, DO) that may be mapped, via software, to inputs of the test-plug hardware-platform 200 (e.g., AI, DI). Similarly, IOC inputs (e.g., AI, DI) may be mapped, via software, to outputs of the test-plug hardware-platform 200 (e.g., AO, DO). Special IO-like contact closures can be tested with a combination of outputs and inputs as will be recognizable to a person of ordinary skill in the art upon reading and understanding the disclosure provided herein.

[00027] With continuing reference to FIG. 2, various embodiments of the test-plug hardware-platform 200 may also include universal I/O circuitry 201, one or more programmable logic devices (PLDs) 203 to build reconfigurable digital circuits, one or more microcontroller units (MCUs) 205 used to, for example, manipulate, correlate, or direct data received from or sent to a production tool (or save to a memory device (not shown)), a radio source 207 (including, for example, Bluetooth® or other wireless devices described below) to receive or transmit data to and from the production tool or other external devices, a number of LED and/or LCD buttons 209 (which may include various displays, readouts, etc.) and a number of other types of electrical-connector support circuitry as is known in the art. In various embodiments, the radio source 207 can be used to interface with a wireless-mesh network, as described in more detail below with reference to FIG. 3. FIG. 2 is also shown to include a number of 24 V connections 221. The skilled artisan will readily recognize that the 24 V level is stated merely for convenience in understanding. Various voltages, higher or lower than 24 V, may of course be included instead of or in addition to the 24 V connections.

[00028] The test-plug hardware-platform 200 may be powered from IOC or various types of either internal power circuitry 220 and/or external power-supplies (not shown) as needed. As shown in FIG. 2, various embodiments of the test-plug hardware-platform 200 can include various voltage levels (e.g., 24 volts) of power connections and ground connections. These power and ground connections may be jumped manually or configured via, for example, software within, for example, a 24 V power and ground jumper module 211. Additionally, the test-plug hardware-platform 200 of FIG. 2 is shown to include onboard power and power management devices such as a lithium-polymer battery and/or other types of battery and DC-to-DC conversion circuits known in the art.

[00029] FIG. 3 shows an example of a wireless-mesh network 300 that can be used with the disclosed subject matter, according to various exemplary embodiments. The wireless-mesh network 300 can eliminate numerous interconnecting cables and jumpers. Further, the wireless mesh network provides for communications between various test-plug hardware-platforms as well as wireless communications with the production tool (or related components thereof) under test. Additionally, a USB® communications device can be used to interface the wireless-mesh network with a computer (e.g., a laptop or tablet computer).

[00030] In various embodiments, the wireless-mesh network 300 can utilize a number of different communications standards, host interfaces, and protocols, such as, for example, IEEE 802.15.4, Zigbee®, XBee®, Bluetooth®, and other wireless communications standards and protocols. These standards and protocols, and others, specify various factors such as a physical layer and access control for low-rate wireless networks.

[00031] Referring now to FIG. 4, an example of an exemplary embodiment of a test-plug hardware-platform 400 when used with a

production tool (or related electrical component, neither of which is shown explicitly but would be well understood to a skilled artisan upon reading and understanding the as-filed Application), according to various exemplary embodiments, is shown. The test-plug hardware-platform 400 may be the same as or similar to the test-plug hardware-platform 200 of FIG. 2.

[00032] In a specific exemplary implementation, a number of test-plug hardware-platforms are coupled to a number of nodes, Node 0 405A, Node 1, 405B, . . . Node X 405X. Each node is labeled as "IO Controller Node," which is a specific hardware platform that can run, for example, an Ethernet for Control Automation Technology (EtherCAT®) protocol software and/or firmware and that adheres to EtherCAT® standards as developed under IEEE 802 standards. However, any type of wireless type or wired type of node known in the art could be used.

[00033] The nodes are electrically and communicatively coupled to (e.g., hardwired or wirelessly coupled), for example, an Ethernet switch 403 (ENET Switch) or an EtherCAT® or ECAT junction box. The ENET Switch 403 is coupled to a test computer 401, which may comprise, for example, a single-board computer (SBC), or other type of MCU or even a laptop computer or tablet computer. The test computer 401 can be used for a number of different purposes as described herein including memory storage and controlling or programming various I/O protocols. In other embodiments, the test computer may also be running a number of simulated devices while actual I/O from the production tool or tools (not shown) is monitored.

[00034] In this specific exemplary embodiment, the test computer 401 may be electrically and communicatively coupled to various types of production tools or related types of equipment (not shown). However, the test computer 401 could be coupled directly to, for example, PCBs or other components within a production tool or

multiple production tools (not shown). Either the test computer 401 or the production tool may also be in direct wireless communication with the test-plug hardware-platforms through the wireless-mesh network 300 of FIG. 3. In turn, the wireless-mesh network 300 can manage an entire network of test-plug hardware-platforms as needed.

[00035] Overall, upon reading and understanding the disclosure provided herein, a person of ordinary skill in the art will recognize that significant advantages of the various embodiments of the test-plug hardware-platform disclosed herein include saving time and saving money. For example, automating testing saves significant time for production tool diagnostics and testing. Further, software testing benefits as well by saving time with hardware-interface testing. Moreover, software testing can be completed before a production tool is built such that integration goes more smoothly and either does not impact or limits impact of a project schedule.

[00036] Throughout this specification, plural instances may implement components, operations, or structures described as a single instance. Although individual operations of one or more methods are illustrated and described as separate operations, one or more of the individual operations may be performed concurrently, and nothing requires that the operations be performed in the order illustrated. Structures and functionality presented as separate components in example configurations may be implemented as a combined structure or component. Similarly, structures and functionality presented as a single component may be implemented as separate components. These and other variations, modifications, additions, and improvements fall within the scope of the subject matter herein.

[00037] Certain embodiments are described herein as including logic devices or a number of components, modules, or mechanisms. Modules may constitute either software modules (e.g., code embodied on a machine-readable medium or in a transmission signal) or

hardware modules. A "hardware module" is a tangible unit capable of performing certain operations and may be configured or arranged in a certain physical manner. In various embodiments, one or more computer systems (e.g., a standalone computer-system, a client computer-system, or a server computer-system) or one or more hardware modules of a computer system (e.g., a processor or a group of processors) may be configured by software (e.g., an application or application portion) as a hardware module that operates to perform certain operations as described herein.

[00038] In some embodiments, a hardware module may be implemented mechanically, electronically, or any suitable combination thereof. For example, a hardware module may include dedicated circuitry or logic that is permanently configured to perform certain operations. For example, a hardware module may be a special-purpose processor, such as a field programmable gate array (FPGA) or an Application-Specific Integrated Circuit (ASIC). A hardware module may also include programmable logic or circuitry that is temporarily configured by software to perform certain operations. For example, a hardware module may include software encompassed within a general-purpose processor or other programmable processor. It will be appreciated that the decision to implement a hardware module mechanically, in dedicated and permanently configured circuitry, or in temporarily configured circuitry (e.g., configured by software) may be driven by cost and time considerations.

[00039] Accordingly, the phrase "hardware module" should be understood to encompass a tangible entity, be that an entity that is physically constructed, permanently configured (e.g., hardwired), or temporarily configured (e.g., programmed) to operate in a certain manner or to perform certain operations described herein. As used herein, "hardware-implemented module" refers to a hardware module. Considering embodiments in which hardware modules are

temporarily configured (e.g., programmed), each of the hardware modules need not be configured or instantiated at any one instance in time. For example, where a hardware module comprises a general-purpose processor configured by software to become a special-purpose processor, the general-purpose processor may be configured as respectively different special-purpose processors (e.g., comprising different hardware modules) at different times. Software may accordingly configure a processor or other configurable device, for example, to constitute a particular hardware-module at one instance of time and to constitute a different hardware module at a different instance of time. Additionally, the software may be used to program firmware.

[00040] Hardware modules can provide information to, and receive information from, other hardware modules. Accordingly, the described hardware modules may be regarded as being communicatively coupled. Where multiple hardware modules exist contemporaneously, communications may be achieved through signal transmission (e.g., over appropriate circuits and buses) between or among two or more of the hardware modules. In embodiments in which multiple hardware modules are configured or instantiated at different times, communications between such hardware modules may be achieved, for example, through the storage and retrieval of information in memory structures to which the multiple hardware modules have access. For example, one hardware module may perform an operation and store the output of that operation in a memory device to which it is communicatively coupled. A further hardware module may then, at a later time, access the memory device to retrieve and process the stored output. Hardware modules may also initiate communications with input or output devices, and can operate on a resource (e.g., a collection of information).

[00041] The various operations of example methods described herein may be performed, at least partially, by one or more processors that are temporarily configured (e.g., by software) or permanently configured to perform the relevant operations. Whether temporarily or permanently configured, such processors may constitute processor-implemented modules that operate to perform one or more operations or functions described herein. As used herein, "processor-implemented module" refers to a hardware module implemented using one or more processors or other configurable components (e.g., such as an FPGA, and ASIC, firmware, of components of the test-plug hardware-platform itself).

[00042] Similarly, the methods described herein may be at least partially processor-implemented, a processor being an example of hardware. For example, at least some of the operations of a method may be performed by one or more processors or processor-implemented modules. Moreover, the one or more processors may also operate to support performance of the relevant operations in a "cloud computing" environment or as a "software as a service" (SaaS). For example, at least some of the operations may be performed by a group of computers (as examples of machines including processors), with these operations being accessible via a network (e.g., the Internet) and via one or more appropriate interfaces (e.g., an application program interface (API)).

[00043] The performance of certain of the operations may be distributed among the one or more processors, not only residing within a single machine, but deployed across a number of machines. In some embodiments, the one or more processors or processor-implemented modules may be located in a single geographic location (e.g., within a home environment, an office environment, or a server farm). In other embodiments, the one or more processors or processor-

implemented modules may be distributed across a number of geographic locations.

[00044] FIG. 5 is a block diagram illustrating components of a machine 500, according to some embodiments, able to read instructions from a machine-readable medium e.g., a non-transitory machine-readable medium, a machine-readable storage medium, a computer-readable storage medium, or any suitable combination thereof) and perform any one or more of the methodologies discussed herein. Specifically, FIG. 5 shows a diagrammatic representation of the machine 500 in the example form of a computer system and within which instructions 524 (e.g., software, a program, an application, an applet, an app, or other executable code) for causing the machine 500 to perform any one or more of the methodologies discussed herein may be executed.

[00045] In alternative embodiments, the machine 500 operates as a standalone device or may be connected (e.g., networked) to other machines. In a networked deployment, the machine 500 may operate in the capacity of a server machine or a client machine in a server-client network environment, or as a peer machine in a peer-to-peer (or distributed) network environment. The machine 500 may be a server computer, a client computer, a personal computer (PC), a tablet computer, a laptop computer, a netbook, a set-top box (STB), a personal digital assistant (PDA), a cellular telephone, a smartphone, a web appliance, a network router, a network switch, a network bridge, or any machine capable of executing the instructions 524, sequentially or otherwise, that specify actions to be taken by that machine. Further, while only a single machine is illustrated, the term "machine" shall also be taken to include a collection of machines that individually or jointly execute the instructions 524 to perform any one or more of the methodologies discussed herein.

[00046] The machine 500 includes a processor 502 (e.g., a central processing unit (CPU), a graphics processing unit (GPU), a digital signal processor (DSP), an application specific integrated circuit (ASIC), a radio-frequency integrated circuit (RFIC), or any suitable combination thereof), a main memory 504, and a static memory 506, which are configured to communicate with each other via a bus 508. The processor 502 may contain microcircuits that are configurable, temporarily or permanently, by some or all of the instructions 524 such that the processor 502 is configurable to perform any one or more of the methodologies described herein, in whole or in part. For example, a set of one or more microcircuits of the processor 502 may be configurable to execute one or more modules (e.g., software modules) described herein.

[00047] The machine 500 may further include a graphics display 510 (e.g., a plasma display panel (PDP), a light emitting diode (LED) display, a liquid crystal display (LCD), a projector, or a cathode ray tube (CRT)). The machine 500 may also include an alpha-numeric input device 512 (e.g., a keyboard), a cursor control device 514 (e.g., a mouse, a touchpad, a trackball, a joystick, a motion sensor, or other pointing instrument), a storage unit 516, a signal generation device 518 (e.g., a speaker), and a network interface device 520.

[00048] The storage unit 516 includes a machine-readable medium 522 (e.g., a tangible and/or non-transitory machine-readable storage medium) on which is stored the instructions 524 embodying any one or more of the methodologies or functions described herein. The instructions 524 may also reside, completely or at least partially, within the main memory 504, within the processor 502 (e.g., within the processor's cache memory), or both, during execution thereof by the machine 500. Accordingly, the main memory 504 and the processor 502 may be considered as machine-readable media (e.g., tangible and/or non-transitory machine-readable media). The

instructions 524 may be transmitted or received over a network 526 via the network interface device 520. For example, the network interface device 520 may communicate the instructions 524 using any one or more transfer protocols (e.g., hypertext transfer protocol (HTTP)).

[00049] In some embodiments, the machine 500 may be a portable computing device, such as a smart phone or tablet computer, and have one or more additional input components (e.g., sensors or gauges). Examples of such additional input components include an image input component (e.g., one or more cameras), an audio input component (e.g., a microphone), a direction input component (e.g., a compass), a location input component (e.g., a global positioning system (GPS) receiver), an orientation component (e.g., a gyroscope), a motion detection component (e.g., one or more accelerometers), an altitude detection component (e.g., an altimeter), and a gas detection component (e.g., a gas sensor). Inputs harvested by any one or more of these input components may be accessible and available for use by any of the modules described herein.

[00050] As used herein, the term "memory" refers to a machine-readable medium able to store data temporarily or permanently and may be taken to include, but not be limited to, random-access memory (RAM), read-only memory (ROM), buffer memory, flash memory, and cache memory. While the machine-readable medium 522 is shown in an embodiment to be a single medium, the term "machine-readable medium" should be taken to include a single medium or multiple media (e.g., a centralized or distributed database, or associated caches and servers) able to store instructions. The term "machine-readable medium" shall also be taken to include any medium, or combination of multiple media, that is capable of storing instructions for execution by a machine (e.g., the machine 500), such that the instructions, when executed by one or more processors of the machine (e.g., the processor

502), cause the machine to perform any one or more of the methodologies described herein. Accordingly, a "machine-readable medium" refers to a single storage apparatus or device, as well as "cloud-based" storage systems or storage networks that include multiple storage apparatus or devices. The term "machine-readable medium" shall accordingly be taken to include, but not be limited to, one or more tangible (e.g., non-transitory) data repositories in the form of a solid-state memory, an optical medium, a magnetic medium, or any suitable combination thereof.

[00051] Furthermore, the machine-readable medium is non-transitory in that it does not embody a propagating signal. However, labeling the tangible machine-readable medium as "non-transitory" should not be construed to mean that the medium is incapable of movement – the medium should be considered as being transportable from one physical location to another. Additionally, since the machine-readable medium is tangible, the medium may be considered to be a machine-readable device.

[00052] The instructions 524 may further be transmitted or received over a network 526 (e.g., a communications network) using a transmission medium via the network interface device 520 and utilizing any one of a number of well-known transfer protocols (e.g., HTTP). Examples of communication networks include a local area network (LAN), a wide area network (WAN), the Internet, mobile telephone networks, POTS networks, and wireless data networks (e.g., WiFi and WiMAX networks). The term "transmission medium" shall be taken to include any intangible medium that is capable of storing, encoding, or carrying instructions for execution by the machine, and includes digital or analog communications signals or other intangible medium to facilitate communication of such software.

[00053] As used herein, the term "or" may be construed in an inclusive or exclusive sense. Further, other embodiments will be

understood by a person of ordinary skill in the art upon reading and understanding the disclosure provided. Further, upon reading and understanding the disclosure provided herein, the person of ordinary skill in the art will readily understand that various combinations of the techniques and examples provided herein may all be applied in various combinations.

[00054] Although various embodiments are discussed separately, these separate embodiments are not intended to be considered as independent techniques or designs. As indicated above, each of the various portions may be inter-related and each may be used separately or in combination with other test-plug hardware-platform discussed herein. For example, although various embodiments of methods, operations, and processes have been described, these methods, operations, and processes may be used either separately or in various combinations.

[00055] Consequently, many modifications and variations can be made, as will be apparent to a person of ordinary skill in the art upon reading and understanding the disclosure provided herein. Functionally equivalent methods and devices within the scope of the disclosure, in addition to those enumerated herein, will be apparent to the skilled artisan from the foregoing descriptions. Portions and features of some embodiments may be included in, or substituted for, those of others. Such modifications and variations are intended to fall within a scope of the appended claims. Therefore, the present disclosure is to be limited only by the terms of the appended claims, along with the full scope of equivalents to which such claims are entitled. It is also to be understood that the terminology used herein is for the purpose of describing particular embodiments only and is not intended to be limiting.

[00056] The Abstract of the Disclosure is provided to allow the reader to quickly ascertain the nature of the technical disclosure. The

abstract is submitted with the understanding that it will not be used to interpret or limit the claims. In addition, in the foregoing Detailed Description, it may be seen that various features may be grouped together in a single embodiment for the purpose of streamlining the disclosure. This method of disclosure is not to be interpreted as limiting the claims. Thus, the following claims are hereby incorporated into the Detailed Description, with each claim standing on its own as a separate embodiment.

CLAIMS

What is claimed is:

1. A test-plug hardware-platform, comprising:
at least one input/output (I/O) connector;
a configurable control system to provide command and operational signals through the I/O connector, the control system further to collect data through the I/O connector from at least one printed-circuit board (PCB) under test; and
a wireless-mesh network configured within the test-plug hardware-platform to interface wirelessly with at least the PCB under test.

2. The test-plug hardware-platform of claim 1, wherein the configurable control system is configured to capture actual input/output signals to and from the at least one PCB under test.

3. The test-plug hardware-platform of claim 1, wherein the configurable control system further comprises universal I/O circuitry to interface with the at least one I/O connector, one or more programmable logic devices (PLDs) to build reconfigurable digital circuits, one or more microcontroller units (MCUs) to direct data received from and sent to the at least one PCB, and a radio source.

4. The test-plug hardware-platform of claim 1, wherein the radio source is configured to operate under at least one communications standard.

5. The test-plug hardware-platform of claim 4, wherein the communications standard includes at least one standard protocol selected from IEEE 802.15.4, Zigbee®, XBee®, and Bluetooth®.

6. The test-plug hardware-platform of claim 1, wherein the configurable control system is configured to test substantially simultaneously the functionality of all inputs and outputs of the at least one PCB.

7. The test-plug hardware-platform of claim 1, wherein the configurable control system is configured to test each connector on the at least one PCB via software within the test-plug hardware-platform.

8. The test-plug hardware-platform of claim 1, wherein the at least one PCB is operating within a production tool.

9. The test-plug hardware-platform of claim 1, wherein the test-plug hardware-platform is configured to test multiple ones of the at least one PCBs within a production tool substantially simultaneously.

10. The test-plug hardware-platform of claim 1, wherein the test-plug hardware-platform is configured to test analog signals and digital signals substantially simultaneously.

11. The test-plug hardware-platform of claim 10, wherein at least one of the analog signals and digital signals include video signals.

12. The test-plug hardware-platform of claim 1, wherein the test-plug hardware-platform is configured to supply power to the at least one PCB.

13. The test-plug hardware-platform of claim 1, wherein the test plug is configured to test low-level communication with the at least one PCB connected to the actual hardware in which each of the at least one PCB is located.

14. The test-plug hardware-platform of claim 1, wherein configuration of software within the configurable control system includes an existing signal list and an interlock table for a particular tool in which the at least one PCB is located.

15. A test-plug hardware-platform, comprising:
at least one input/output (I/O) connector; and
a configurable control system to provide command and operational signals through the I/O connector, the control system further to capture actual input/output signals to and from at least one printed-circuit board (PCB) under test, the at least one PCB being operated within a production tool.

16. The test-plug hardware-platform of claim 15, further comprising a wireless electronic control system and a wireless-mesh network to interface wirelessly with the at least the PCB under test.

17. The test-plug hardware-platform of claim 15, wherein the test-plug hardware-platform further comprises power and ground connections to supply to the at least one PCB, the power and ground connections being configurable as an output to the at least one I/O connector via software.

18. The test-plug hardware-platform of claim 15, further comprising a test computer, the test computer being configured to:
run a number of simulated devices; and
monitor actual I/O from the each of the at least one PCB.

19. A test-plug hardware-platform, comprising:
a wireless electronic control system; and
a configurable control system to provide command and operational signals through the wireless electronic control system, the control system further to collect data through the wireless electronic control system from at least one printed-circuit board (PCB) under test, the control system further to capture actual input/output signals to and from the at least PCB under test with the at least one PCB being operated within a production tool.

20. A test-plug hardware-platform, wherein the wireless electronic control system is configured to provide communications between test-plug hardware-platform via wireless communications with the production tool.