[035] The embodiment of Figure 1 uses a board 100 with a 14 by 11 array of sensors 104 although other dimensions (not shown) are possible. Each of the sensors 104 is configurable to be reserved for tool storage or kept vacant.

[036] As mentioned above, the plurality of sensors 104 is either configured:

a) to be reserved for tool storage (i.e., referring to sensors 110 that when in use are kept for tool storage). Since tools occupy space, a plurality of adjacent sensors is typically reserved for tool storage. In one approach, more sensors are reserved for tool storage than required. For example, a 6 by 6 sensor array area may be reserved for a tool that only requires a 4 by 4 sensor array space. Accordingly, reserved sensors 110 need not necessarily be in contact with the tool they are kept aside to store; or

b) to be unoccupied / vacant (i.e., referring to sensors 112 that when in use are unoccupied by a tool).

[037] The remote terminal 108 learns the location of the reserved sensors 110 and the vacant sensors 112 during initialisation of the board 100, for example, from the signals received from the decoder 106 board through placement of tools on the board 100. With reference to the description for step 908 in Figure 9 below, the letter "y" refers to status lights for the board 100 turning yellow to indicate that the board 100 is learning the shape of the tool 116. In another approach, the remote terminal 108 learns the location of the reserved sensors 110 and the vacant sensors 112 by a user entering their co-ordinates during initialisation.

[038] When reserved for tool storage, the sensors 110 can be in an activated state or a null state. In the activated state, the reserved sensors 110 are programmed to detect for tool presence or absence during tool return. Tool presence occurs when the reserved sensor 110 detects contact with a part of the returned tool 116. Tool absence occurs when the reserved sensor 110 does not detect contact with any part of the returned tool. Each activated sensor may further be either in a positive logic state or a negative logic state.

[039] Activated sensors in the positive logic state are configured to expect tool presence during tool return i.e., there is contact with a part of the returned tool 116 during tool return. If such a positive logic state sensor does not detect tool presence during tool return, it will send an error signal. However, if the sensor detects tool presence during tool return, it will not send an error signal.

[040] Activated sensors in the negative logic state are configured to expect tool absence during tool return, i.e., there is no contact with a part of the returned tool 116 during tool return. For example, sensors 114 that fall beyond the perimeter of the tool 116 on the board 100 may be designated as negative logic state sensors. Other examples are sensors that parts of the tool 106 when lying flush on the board 100 do not contact (see Figure 5), whereby these sensors still fall within the perimeter of the tool 106 on the board 100. If the negative logic state sensor detects tool presence during tool return, it will send an error signal. However, if the sensor does not detect tool presence during tool return, it will not send an error signal.

110 and the vacant sensors 112 are provided through the remote terminal 108, where it is optional for the tools to be present on the sensors 104.

[047] Regardless of which approach is used during tools training / board 100 initialisation, the decoder 106 programs the reserved sensors 110 to have one or more sensors in the activated positive logic state; the activated negative logic state; and / or the null state. The algorithm used for such sensor designation may be the same as the one used for automatic board 100 initialisation or a separate algorithm.

[048] For example, in a first algorithmic implementation, all the reserved sensors 110 are in the activated positive logic state, so that the decoder 106 transmits an error message if any of these sensors 110 detect absence of a tool during tool return. In a second algorithmic implementation, some of the reserved sensors 110 are in the activated positive logic state, while a remainder of the reserved sensors 110 are in the activated negative logic state. The decoder 106 transmits an error message if any of the sensors 110 in the activated positive logic state detect absence of a tool during tool return or any of the sensors 110 in the activated negative logic state detect presence of a tool during tool return. In a third algorithmic implementation, a first group of the reserved sensors 110 are in the activated positive logic state; a second group of the reserved sensors 110 are in the activated negative logic state; and a remainder of the reserved sensors 110 are in the null state. The decoder 106 transmits an error message if any of the sensors 110 in the activated positive logic state detect absence of a tool during tool return or any of the sensors 110 in the activated negative logic state detect presence of a tool during tool return. The decoder 106 will not receive a signal from the sensors 110 in the null state.

[049] The remote terminal 108 then saves the location of each set of reserved sensors 110 to demarcate an area of the board 100 set aside for tool storage. Details of the tool 116 stored on the reserved sensors 110 are input into the remote terminal 108, for example by indicating the type of tool / part number stored by each set of reserved sensors 110. In one approach, this may be achieved by a user entering the tool type / part number that was placed on each location on the board 100. In another approach, the remote terminal 108 may automatically assign the tool type / part number by mapping the activated sensors against a tool database. Other details include weight of each tool 116 which is used to monitor tools kept in peripheral storage, described in further detail below with respect to Figures 6 to 8.

[050] A data storage 120 saves the details of the tool 116 stored by the reserved sensors 110, a location of the reserved sensors 110, and the location of the vacant sensors 112. The data storage 120 may be integrated with the remote terminal 108 or separately located, such as realised through cloud storage.

[051] When in use (i.e., after tools training / board 100 initialisation), the decoder 106 assesses whether the correct tools are being returned to their assigned storage spaces. This is achieved by the decoder 106 determining whether activated sensors within the sensors 110 reserved for tool storage are triggered correctly, such as when none of the sensors 110 in the activated positive logic state transmit a signal that a tool is absent and none of the sensors 110 in the activated negative logic state transmit a

and 18. The other sensors 3, 6, 8, 10, 12 and 13 are programmed to be in the null state. The remaining sensors 1, 5, 9, 15, 16, 17, 19 and 20 may be programmed to be in the activated negative logic state.

[058] If any of the activated positive logic state sensors detect tool absence during tool return (example only sensors 4, 11 and 14, but not sensors 2, 7 and 18), the decoder 106 will still send an error message that a wrong tool is returned because the reserved sensors are incorrectly triggered. If all the activated positive logic state sensors detect tool presence, but any of the activated negative logic state sensors also detect presence of a tool, it would be another instance of the reserved sensors being incorrectly triggered. The decoder 106 will still send an error message that a wrong tool is returned. The null state sensors 3, 6, 8, 10, 12 and 13 will not send a signal, regardless of tool presence or absence.

[059] The second implementation is advantageous to prevent attempts at deriving the combination of sensors to deceive the board 100 that a correct tool has been returned.

[060] A 2 by 4 sensory array 424 (for a total of 8 sensors) may be reserved for storing tool 420, which does not lie flush when resting on the board 100 (see cross section view of Figure 5). Tool 424 is in contact with sensors 25, 26, 31 and 32, but not sensors 27 to 30.

[061] During tools training / board 100 initialisation, the learning algorithm detects that a tool is not in contact with sensors 27 to 30 and may program them to be in the activated negative logic state. The remaining sensors 25, 26, 31 and 32 may then be programmed to have a combination of sensors in the activated positive logic state or in the null state.

[062] If any of the activated negative logic state sensors 27 to 30 detect tool presence during tool return (example sensors 27 and 28), the decoder 106 will still send an error message that a wrong tool (not shown) is returned because the reserved sensors are incorrectly triggered. If all the activated negative logic state sensors 27 to 30 detect tool absence during tool return, but any of the activated positive logic state sensors within sensors 25, 26, 31 and 32 detect tool absence, the decoder will still send an error message that a wrong tool is returned because the reserved sensors are incorrectly triggered. The null state sensors within sensors 25, 26, 31 and 32 will not send a signal, regardless of tool presence or absence.

[063] A trained board 100 can be re-initialised / reset, providing a means to periodically rearrange the reserved sensors for each of the tools on the board 100 to prevent attempts at deriving the combination of sensors where placing a wrong tool could deceive the array of sensors that the correct tool has been returned. The tools are kept in their original locations during this re-initialisation.

[064] Using the sensors reserved for the array 422 as an example, if the sensors 3, 6, 8, 10, 12 and 13 are at present in the null state, resetting the board 100 may result in one or more of them being changed to the positive logic state. If the sensors 2, 4, 7, 11, 14 and 18 are at present in the positive logic state, rearrangement may result in one or more of them being changed to the null state. If the sensors 1, 5, 9, 15, 16, 17, 19 are at present in the activated negative logic state, resetting the board 100 may result in one or more of them being changed to the null logic state. If the sensors 1, 5, 9, 15, 16, 17, 19 are at

notification serves to alert the user to check whether they have returned the correct tools to the board 100. Upon removal of the mismatched tools, the board 100 returns to the status of step 806 to await detection of tool return.

[078] On the other hand, step 816 occurs if the comparison of step 814 is a match (i.e. the reduction in the weight reading tallies with the recorded weight of the tool / part number assigned to the sensors to which the tool is returned). A signal is sent in step 820 to update the database 822 that a correct tool is returned to the board 100.

[079] After the user returns all tools correctly, they may log out at step 818. The flowchart then ends in step 824.

[080] A "mother / child" relationship is thus established between the board 100 and the auxiliary toolbox 626 which allows monitoring of the number of tools that are being withdrawn from the "mother" board 100 and placed into the "child" auxiliary toolbox 626. The use of a weight sensor in the auxiliary toolbox 626 is also advantageous for compactness, since the auxiliary toolbox 626 may not have enough space to house a sensor board. Upon returning tools from the auxiliary toolbox 626 to the board 100, the remote terminal 108 will verify if the correct tools are being returned from the signals received from the decoder 106 in the manner described above with respect to Figures 3 to 5. If in the event of a missing tool or a wrong tool being returned, the remote terminal 108 will display a warning message of which tool is incorrect or missing.

[081] Figure 9 shows a flow chart 900 for tools training / board 100 initialisation. Figure 9 is described in conjunction with Figures 1 and 2 as follows.

[082] Board initialisation starts at step 902 and proceeds with tool training at step 904. In an automated approach, tools are laid spaced apart over the board 100 with the array of sensors 104. The decoder 106 communicates with the sensors 104 to learn which of them detect a tool thereon to determine which are the ones to be reserved for tool storage; and which are unoccupied / vacant. In a manual approach, co-ordinates for the reserved sensors 110 and the vacant sensors 112 are provided through the remote terminal 108, where it is optional for the tools to be present on the sensors 104. Both the automated and manual approaches thereby result in sensors being activated in step 906.

[083] As the sensors are activated, status lights on the tool cabinet 200 turn yellow to reflect that sensors are collecting, at step 908, data about the tools, like their shape and size to build a database at step 910 for a learning logic program to use to determine the number of sensors that need to be reserved to store a tool. Testing and evaluation occur at step 912, whereby step 910 is reiterated to ensure that the reserved sensors can accurately detect for tool presence. Reiteration stops at step 914 when the database has enough datasets to draw from to determine an arrangement of sensors that need to be reserved for tools placed on the board 100.

[084] At step 916, the learning logic algorithm is programmed with the location of reserved sensors 110 (along with which of them are in a positive logic state, negative logic state or null state) and vacant

f) If a tool returned to the cutout location does not trigger the correct sensors reserved for tool storage, the board will send a signal that a wrong tool has been returned.

[092] Figure 11 depicts components of the remote terminal 108 shown in Figure 1.

[093] As shown in Figure 11, the example remote terminal 108 includes a processor 1104 for executing software routines. Although a single processor is shown for the sake of clarity, the remote terminal 108 may also include a multi-processor system. The processor 1104 is connected to a communication infrastructure 1106 for communication with other components of the remote terminal 108. The communication infrastructure 1106 may include, for example, a communications bus or network.

[094] The remote terminal 108 further includes a main memory 1108, such as a random access memory (RAM), and a secondary memory 1110. The secondary memory 1110 may include, for example, a hard disk drive 1112, a removable storage drive 1114 (which may be a USB flash drive, a flash memory device, a solid state drive or a memory card). As will be appreciated by persons skilled in the art, the hard disk drive 1112 and the removable storage drive 1114 can store computer executable program code instructions and/or data.

[095] The remote terminal 108 also includes at least one communication interface 1124. The communication interface 1124 allows software and data to be transferred between the remote terminal 108 and external devices via a communication path 1126. In various embodiments of the inventions, the communication interface 1124 permits data to be transferred between the remote terminal 108 and a data communication network, such as a public data or private data communication network. The communication interface 1124 may be used to exchange data between other remote terminals (not shown) so that the remote terminals form part of an interconnected computer network. Examples of a communication interface 1124 can include a modem, a network interface (such as an Ethernet card), a communication port (such as a serial, parallel, printer, GPIB, IEEE 10394, RJ45, USB), an antenna with associated circuitry and the like. The communication interface 1124 may be wired or may be wireless. Software and data transferred via the communication interface 1124 are in the form of signals which can be electronic, electromagnetic, optical or other signals capable of being received by communication interface 1124. These signals are provided to the communication interface via the communication path 1126.

[096] As shown in Figure 11, the remote terminal 108 further includes a display interface 1102 which performs operations for rendering images to an associated display 1130.

[097] Computer programs (also called computer program code) are stored in main memory 1108 and/or secondary memory 1110. Computer programs can also be received via the communication interface 1124.

[098] Software may be stored in a computer program product and loaded into the remote terminal 108 by using the removable storage drive 1114 or the hard disk drive 1112. Alternatively, the computer program product may be downloaded into the remote terminal 108 over the communications path 1126.

instance, after the location of the reserved sensors are determined from tools training / board 100 initialisation, the remote terminal may be used to change one or more sensors, designated to be in the activated positive logic state during tools training / board 100 initialisation, to the null state.

ii) learning of tool placement and labelling of tool type / part number: the location of each set of reserved sensors is saved to demarcate an area of the sensor board reserved for a tool. Each set of reserved sensors is then labelled by indicating the type of tool / part number stored, i.e. a user will enter the tool type / part number that was placed on each location on the sensor board. This builds a record of the tools / part numbers that the tool tray stores and their assigned locations. Tools are then secured by a foam board placed on top of sensor board, the foam board having a cutaway for each tool, corresponding to the outline of that tool (see Figure 7 below).

iii) tool user registration to determine the users who have accessed the tool tray and the tools they have removed

iv) sensing of tool removal and updating of tool check-out records (to facilitate tracking of users and their checked out tools)

v) verification of correct tool return and placement. If a returned tool does not trigger the correct sensors, it is determined as a wrong tool return. Oversized tools will not be returnable because they cannot fit into the foam cutaway.

vi) onboarding of additional tools. A new tool / part number may be stored on any part of the sensor board having sufficient space. Dimensions of the new tool / part number and its intended storage location on the sensor board can be sent to a remote terminal, to reserve the required number of unoccupied / vacant sensors at the intended storage location. The new tool / part number is entered and the record of the tools / part numbers that the tool tray stores is updated. The existing foam board needs to be modified by removing the portion of foam through which the new tool / part number is to be inserted. An error message will issue if the placed tool does not match the number of reserved sensors during the process of onboarding an additional tool.

vii) relocation of existing tool placement. The location of existing tools can be rearranged. The reserved sensors at the existing locations are released (i.e., no longer reserved), while the required number of unoccupied / vacant sensors at the new locations for the tools to be relocated are reserved to store the relocated tools. The existing foam board may need to be replaced with a new foam board. The record of the tools / part numbers that the tool tray stores is updated.

viii) a “mother / child” relationship with auxiliary storage used to store a subset of tools from the sensor board. The weight of tools in the auxiliary storage is used to track tool movement by tallying against weight records of the tools removed from the sensor board.

[105] In the application, unless specified otherwise, the terms "comprising", "comprise", and grammatical variants thereof, intended to represent "open" or "inclusive" language such that they include recited elements but also permit inclusion of additional, non-explicitly recited elements.

CLAIMS

1. A tool storage system comprising:

a board with an array of sensors, each configurable to be reserved for tool storage or vacant;

a decoder to monitor usage of the array of sensors;

a remote terminal to process signals received by the decoder, the remote terminal configured to:

learn a location of the reserved sensors and the vacant sensors from the signals received during initialisation of the board;

receive input providing details of the tool stored on the reserved sensors;

detect, from the signals when the board is used, for removal of tools from and return of tools to the reserved sensors; and

determine whether to transmit a message that a wrong tool is returned when the decoder relays that at least one of the reserved sensors is triggered incorrectly during the tool return; and

a data storage to save details of the tool stored by the reserved sensors, the location of the reserved sensors, and the location of the vacant sensors; and

wherein the reserved sensors comprise sensors in a positive logic state configured to expect tool presence during tool return; and sensors in a negative logic state configured to expect tool absence during tool return.

2. The tool storage system of claim 1, wherein the reserved sensor incorrectly triggers if during tool return the reserved sensor returns a status that is contrary to designation during initialisation.

3. The tool storage system of claim 1 or 2, wherein the incorrect triggering occurs when the sensor in the positive logic state detects tool absence during the tool return or the sensor in the negative logic state detects tool presence during the tool return.

4. The tool storage system of claim 3, wherein the sensors in the positive logic state are further configured to send an error signal when detecting tool absence during the tool return; and the sensors in the negative logic state are further configured to send an error signal when detecting tool presence during the tool return.

5. The tool storage system of any one of the preceding claims, wherein the reserved sensors further comprise sensors in a null state configured to be silent.

6. The tool storage system of any one of the preceding claims, wherein the remote terminal learns the location of the reserved sensors and the vacant sensors through placement of tools on the array of sensors.

7. The tool storage system of claim 6, wherein designation of the reserved sensors to the positive logic state, the negative logic state or the null state is randomly decided.

8. The tool storage system of any one of claims 5 to 7, wherein the remote terminal is further configured to rearrange the reserved sensors for the tools on the board, in response to a reset command.

9. The tool storage system of any one of claims 1 to 5, wherein the remote terminal learns the location of the reserved sensors and the vacant sensors through co-ordinate input.

10. The tool storage system of any one of the preceding claims, wherein the message that a wrong tool is returned is transmitted when one or more of the following occurs: number of the reserved sensors that are triggered incorrectly exceeds a threshold or the reserved sensors that are triggered incorrectly are at a particular location.

11. The tool storage system of any one of the preceding claims, wherein the remote terminal is further configured to:

communicate with a weight sensor to obtain a weight reading; and

compare whether the obtained reading tallies with a weight record of tools removed from or returned to the array of sensors.

12. The tool storage system of claim 11, wherein the weight sensor is provided in an auxiliary toolbox, and wherein the weight reading is of the tools stored in the auxiliary toolbox.

13. The tool storage system of claim 12, wherein the remote terminal is further configured to obtain an identifier of a wireless tag provided in the auxiliary toolbox when brought into proximity with the board.

14. The tool storage system of claim 13, wherein the remote terminal is further configured to record the identifier of the auxiliary toolbox with one or more of an identity of the board, or a user to which the auxiliary toolbox is assigned.

15. The tool storage system of any one of claims 12 to 14, wherein the remote terminal is further configured to transmit an error notification in response to detection of a mismatch between the weight record of the tools removed from or returned to the array of sensors and the weight reading from the weight sensor.

16. The tool storage system of any one of the preceding claims, wherein the array of sensors comprises capacitive sensors.