| | |
|---|---|
| 0x0080 | link test |
| 0x0100 | service completed |
| 0x0200 | call for machine repair |
| 0x0400 | repair completed |

**TYPE 4**

(RAM Data Dump)

| Byte Name | Description |
|---|---|
| ADDR | starting address |
| D0 | data byte 0 |
| D1 | data byte 1 |
| D2 | data byte 2 |
| D3 | data byte 3 |
| D4 | data byte 4 |
| D5 | data byte 5 |
| D6 | data byte 6 |
| D7 | data byte 7 |
| D8 | data byte 8 |
| D9 | data byte 9 |
| DA | data byte A |
| DB | data byte B |
| DC | data byte C |
| DD | data byte D |
| DE | data byte E |
| DF | data byte F |

This data packet is given by example, while with other data systems other data packets may be utilized.

Referring now to the example FIGURE 7, once a NAK signal has been received at step 196, the microprocessor begins transmitting a data packet to the central computer by first getting a data packet first on the queue at a step 206. The data packet is then transmitted at a step 208. Following transmission, the microprocessor shown again polls the modem to determine if another NAK signal has been received at step 210. If the central computer transmits another recognizable signal, a NAK signal shown, the microprocessor knows that the transmission did not arrive correctly. Therefore, the microprocessor loops back to step 208 and the data packet is again transmitted. If no NAK signal is received in step 210, the microprocessor proceeds to a step 212 wherein the modem is polled to see if an acknowledge a second recognizable signal ("ACK") shown, has been received. If no ACK signal has been received, the program returns to the control mode at a step 214. If an ACK signal is received, the microprocessor knows the central computer system has received the data packet correctly and the data packet transmitted is removed from the queue at step 216.

After removing the data packet from the queue, the example microprocessor determines if the queue is empty at a step 218. If the queue is not empty, the microprocessor loops back to step 206 and the next data packet is transmitted as described above.

Once the queue of data packets to be transmitted is empty, the microprocessor shown proceeds to a step 220 wherein an ACK signal is transmitted to the central computer system. This ACK signal indicates to the central computer system that the remote vending machine is ready to accept data packets transmitted from the central computer to the remote vending machine. The data packets transmitted from the central computer to the remote vending machine. In the specific example shown these data packets are defined by packet type as follows:

**DATA PACKETS TRANSMITTED FROM CENTRAL COMPUTER TO THE REMOTE VENDING MACHINE**

**TYPE 101**

(Transmit 16 Bytes of Microprocessor's Memory from Starting Address)

| Byte Name | Description |
|---|---|
| ADDR | starting address (2 bytes) |

**TYPE 102**

(Rewrite N Bytes of Microprocessor's Memory from Starting Address)

| Byte Name | Description |
|---|---|
| ADDR | starting address (2 bytes) |
| D0...DN | n data bytes (n = packet length minus 9) |

**TYPE 103**

(Rewrite Phone Number of Central Computer)

| Byte Name | Description |
|---|---|
| PH1...PH36 | 36 bytes phone number (blank-no outbound alarm) |

**TYPE 104**

(Set Vending Machine's Alarm Criteria)

| Byte Name | Description |
|---|---|
| CA | compressor cycles per day max |
| CI | compressor cycles per day min |
| UNID | rewrite unit ID of vending machine |
| CB | checksum bad alarm enabled - 1 |

| | |
|---|---|
| CC | compressor cycles alarm enabled - 1 |
| IN | intrusion alarm enabled - 1 |
| TE | temperature exceeded alarm enabled - 1 |
| CD | change depleted alarm enabled - 1 |
| CP | column product alarm criterion - 1 byte |
| TPBC | total product alarm criterion - 2 bytes |
| SV | send service packet upon servicing complete alarm enabled - 1 |

**TYPE 105**

(Reset Vending Machine's Alarm Bits)

| Byte Name | Description |
|---|---|
| BPBP | set alarm bit pattern - 2 bytes |

**TYPE 106**

(Set PIN for Service Technician)

| Byte Name | Description |
|---|---|
| PWI...PW7 | 7 bytes of numeric data define PIN |

**TYPE 107**

(Record Message for Service Technician)

| Byte Name | Description |
|---|---|
| ME1...ME16 | 16 bytes of alphanumeric data for service technician |

In a step 222, the example microprocessor determines if an ASCII representation of a colon symbol as previously set forth has been transmitted. As shown in FIGURE 11, this recognizable symbol marks the beginning of all of the data packets transmitted between the vending machine and the central computer. If no colon symbol is transmitted, the microprocessor returns to the control mode at a step 224. Once a colon symbol has been transmitted, the microprocessor shown determines if the entire data packet has been received correctly at a step 226. If the data packet has not been received correctly, the microprocessor causes the modem to transmit a NAK signal at a step 220 to indicate the data packet was not received correctly. The example microprocessor then loops back to step 222 and looks for the beginning of the same data packet to be retransmitted.

If the data packet was received correctly, the program branches to the analyze mode 290 to perform the task indicated by the data packet as will be described in further detail below. Upon returning from the analyze mode, the microprocessor shown causes the modem to transmit an ACK signal at a step 232 that indicates to the central computer that the data packet has been received and acted upon, and that the vending machine is waiting for another data packet to be transmitted. This process continues until the central computer fails to transmit another data packet whereupon the microprocessor returns to the control mode at the step 224.

In some cases (i.e., when a critical alarm condition exists or if the microprocessor is programmed to alert the central computer system immediately after a service call is completed), the example microprocessor will initiate a call to the central computer system. At a step 200, the microprocessor instructs the modem to connect the central computer. The microprocessor then polls the modem to determine if a carrier is present in a step 202. If no carrier is present, the microprocessor loops back to step 200 and dials again. Upon establishing a connection with the central computer system, the microprocessor transmits an alarm or data service complete packet that has been previously placed on the queue. Transmission of the data packet to the central computer takes place as described above.

FIGURE 8 is a flow chart showing the steps taken by the example microprocessor when operating in the service mode 250. Upon entering the service mode from the control mode when the microprocessor shown detects the door to the vending machine has been opened, the microprocessor determines if the service technician enters a PIN or recognizable signal within a predetermined amount of time (for example ten seconds). The particular PIN is stored in the microprocessor's RAM and can be modified at any time by the central computer system. If the PIN is not entered within this predetermined amount of time, the microprocessor sets an intrusion alarm bit at step 254 and returns to the control mode at step 256. The microprocessor then detects the intrusion alarm bit as being set and enters the alarm mode.

Assuming the PIN has been entered in the predetermined amount of time, the example microprocessor then asks the service technician to enter information regarding the service to be completed. In step 258, the microprocessor queries the technician for the total amount of product added in each column of the vending machine. In a step 260, the microprocessor asks the service technician to enter the total amount of cash removed from the machine. In a step 262, the microprocessor asks for the amount of change left in the coin changer. After the service is complete, the microprocessor generates a service data packet and places the packet on the queue at a step 264.

Once the service call is complete, the example microprocessor reads the status of a service packet bit in a pair of alarm response code bytes in a step 266. This bit indicates whether the vending machine is to contact the central computer upon completion of the service call should wait to inform the central computer of the information obtained from the service technician the next time the central computer calls the vending machine. If the service packet bit indicates the central computer is to be called at the completion of the service, the microprocessor data packet proceeds to the communications mode at a step 268. If the status of the service packet bit indicates the microprocessor is not to call the central computer upon completion of the service call, then the microprocessor returns to the control mode at a step 270.

FIGURE 9 is an example flow chart showing the steps that might be taken by the microprocessor when operating in the analyze mode 290. Upon entering the analyze mode from the communications mode, the microprocessor reads the packet type of data indicated by byte 4 of the received data packet as shown in FIGURE 11. Byte 4 shown informs the microprocessor what type of action is to be taken. At a step 294 it is determined whether the data packet is of type 101. If the data packet is of type 101, the microprocessor transmits the contents of its RAM memory beginning at a starting address which is read from the received data packet in step 296. At step 298, the example microprocessor causes the modem to transmit 16 bytes of data beginning at the starting address. Once the data has been transmitted, the program returns to the communications mode at step 334.

In step 300 shown, it is determined the data packet is of type 102. Data packet type 102 indicates to the example microprocessor that it is to rewrite portions of its RAM memory with data values transmitted from the central computer system. At step 102, the microprocessor reads the starting address and determines the number of bytes to be rewritten. The number of bytes is determined by the value of the packet length byte minus nine. In step 304 shown, the new memory values are read and the RAM memory is rewritten starting at the starting address determined in step 302. Upon rewriting the RAM memory, the microprocessor returns to the communications mode.

In step 306 shown, it is determined if the data packet is of type 103. This data packet type causes the microprocessor to modify the communication parameters to the central computer. In step 308, the microprocessor reads 36 bytes of data. These 36 bytes are stored at the central computer in step 310. After rewriting, the microprocessor returns to the communications mode.

In step 102 shown, it is determined if the data packet is of type 104. This data packet type causes the microprocessor to rewrite its alarm response data which sets the alarm conditions for the vending machine. In step 314, the microprocessor reads the new alarm response data and in step 316, the microprocessor overrides the previous alarm response data. After the alarm response data has been rewritten, the microprocessor returns to the communications mode.

In step 318 shown, the example microprocessor determines if the data packet is of type 105. Type 105 packets cause the microprocessor to artificially set the bits in a pair of bytes which define the alarm conditions of the vending machine as described above. After the alarm bytes have been set, the microprocessor goes to the alarm mode in step 122 wherein the alarm bytes are transmitted to the central computer system.

If the example data packet is not of type 105, the microprocessor determines if the message is of type 106 at step 124. Data packet type 106 causes the microprocessor to read seven bytes of PIN's for the service technician. The old PIN is overwritten at a step 328 before returning to the communications mode.

Finally, the example microprocessor determines if the received data packet is of type 107 at a step 330. Data packet type 107 records 16 bytes of alphanumeric data that is recorded for the service technician to be read during the next service call. The message bytes are stored in memory at a step 332 before the microprocessor returns to the communications mode.

FIGURE 10 is an example of a flow chart showing the steps taken by the microprocessor shown in the alarm mode 340. Upon entering the alarm mode from the control mode, the microprocessor reads the alarm response bytes in step 342. In step 346, the microprocessor compares the alarm bytes described above and compares them to the alarm response bytes in order to determine if the alarm condition is critical. If the alarm is set as critical, the microprocessor generates an alarm data packet and places it on the queue in a step 348 before going to the communications mode in step 350. If the alarm is not critical, the microprocessor simply returns to the control mode at step 352.

FIGURE 12 is an example of a diagram of a handheld data entry terminal 400 that might be used by a service technician to enter data into the shown microprocessor. With this system, the service technician can inform the system of the amount of product added to the machine, the amount of money removed, the content of the change counter, as well as other data. The handheld terminal 400 disclosed has a case 402 that includes a series of keys 406 and an enter button 408. The keys 406 are used to type alphanumeric data on a display 404, which is transmitted to the microprocessor upon hitting an enter key 408. Communication preferably takes place between the microprocessor and the handheld terminal using either a conventional infrared transmitter/receiver indicated at

410 or via mechanical connection such as a stereo plug 412. In the stereo plug one channel is used to transmit from the handheld unit while the other channel is used to receive prompts from the vending machine.

FIGURE 13 is an example block diagram of the handheld data entry terminal 400 described above. This specific handheld terminal includes its own microprocessor 420, a read only memory 424 and a random access memory 426 which are coupled to the microprocessor on a set of bus and control leads 422. Additionally, the keys 406 and display 404 are also connected to the microprocessor on the bus 422. The microprocessor shown communicates with the sensing and communication circuit in the vending machine via a serial point 430. The port shown is a serial port connected to drive an infrared transmitter 432. Additionally, the infrared receiver 434 is used to receive infrared signals transmitted from the sensing and communication circuit to the handheld unit. If a mechanical plug is used, the transmit and receive signals are coupled to a conventional plug, which is inserted by the service technician and allows an appropriate connector to the vending machine. The handheld terminal 400 shown is powered by a battery 428. It could also be powered by the vending machine.

Upon receipt of the information relative to the vending machine from the remote link unit 30 over the network 16, the information shown is then available at the computer for selective presentation and manipulation.

In the invention of the present application, due to the data acquisition units, virtually all of the information needed in respect to the vending machines can be located in a single database, can be processed with the same programming, and can be visually presented with a limited number of easily understood video screens.

In respect to the single database, all of the data for every machine in a single system is preferably stored in a single database having a number of fields and name identity matching that of the maximum capabilities of the common signal. This allows the data for every machine to be present for analysis and presentation in a unified manner. This includes the generation of graphic representations of vending machines as well as the development of reports and other matters. It is noted that there will be empty fields in this type of system. These empty fields as present in the database preferably are ignored in developing the graphic representations and/or reports generated by this system. This can be accomplished by a sub-routine in the processing software blanking empty fields.

It is noted that in the event that the common signals are decoded (as in the described FIGURE 3 matrix system) and/or otherwise processed by the computer preferably this occurs prior to storage in the database.

In respect to the same programming, this programming would develop the graphic representations and reports in a common manner from the database. This common manner would preferably include a data inhibition or blanking sub-routine set to recognize empty fields in the processing of the data and automatically act accordingly.

In respect to the graphic representation, this could include automatically developing the representations to present only the active field information, and modifying the display appropriately. For example, if a particular machine had five columns of inventory, a compressor that cycles, a temperature alarm, and an entry alarm, once utilized or preset, these items would be presented on the screen; this even though the temperature alarm icon and entry alarm icon may be inactive (i.e., normal) at the time of presentation. Further, although the programming may be capable of generating an image having 15 columns, only the active five columns would appear. This could be spread out over the entire column area or could appear as one third the available area as set by an operator. However, since there is no for example change empty sensor or field, the change icon would never appear on the screen.

In respect to the limited screens due to the use of a common signal content, one screen could technically be utilized for all machines, preferably as set forth with software programmed to ignore and not display non-data parameters. For example, with a machine having only 12 columns of inventory and an intrusion door open switch, no temperature sensor, no compressor sensor, and no other sensor, only the active information (12 columns of inventory plus the door open switch) would be presented: The missing sensors would never appear for this machine (although they would if applicable for a different machine). The software thus preferably has the ability to present a very complex screen while the system itself tracks the available data presenting on the screen and processing only the available data. Non-information, empty fields, are ignored. Further, the data can be manipulated by a limited number of computer sub-routines to provide uniform information for the vending machines. This could allow a single graphic representation to be utilized for all vending machines; presenting the common elements of the vending machines in a single manner no matter what the type or nature of the particular machine.

Note that although there are over many hundreds of specific vending machines (over 200), due to the basic commonality between machines, the basic and important date can be presented with a lesser number of screens. For example, it has been ascertained that about 20 basic screen images of vending machines will allow the presentation of most vending machines on the market today.

It is preferred that there be a central data base having the display information for these basic screens. Thus upon the specification of an appropriate screen either manual or automatic, the computer 15 would generate the appropriate image of a vending machine accurately representative of the machine then being presented.

Other parts of the screen, for example the various condition icons, can be similarly generated.

It is noted that when an inventory of an item is developed on the screen, it is preferred that the items comprising this inventory be developed with images representative thereof. For example, if a pop can machine has columns of inventory, the circular end sections of pop cans would be shown in such columns. Additional example if change status is shown, a flat rectangle representative of the edges of the coins would be shown in the change area.

Due to the common signal content, technically a single graphic display could be utilized for all vending machines; Specifically displaying the common information regardless of the type of machine. The reason for this is that the operator does not care about what any given machine is, only what its status, and this status is primarily dependent on the common operational elements. Also some operators will rely primarily on the reports generated by the system.

For operator intuitive convenience, it is preferred that a number of screens be utilized representing types of machines. For example, seven screens: 1) pop/container; 2) candy; 3) snacks; 4) frozen ice cream/popsicles; 5) coffee/cocoa/tea; 6) pop/liquid, and 7) service utilized would enable a vendor to cognitively ascertain the nature of most common food type vending machines (as set forth above, 20 screens would allow an accurate representation of most machines). A further set of screens, for example a communications screen and a route screen, would allow access to the system.

Preferably, a screen would be designed to be able to display the optimum number of pieces of information for the majority of all vending machines, with machines having lesser capabilities being presented in a modified form as previously set forth. For example, there are some very large pop/container vending machines which have nine column selections in a single row, each holding approximately 75 cans. There are also pop/container type vending machines which have but three columns, each holding 25 cans. The basic screen program under these circumstances would be designed to have the capability of presenting the larger machine data. This would be the default condition of the screen. However, upon entering of the smaller machine's type or capabilities, the screen would be automatically modified so as to present but the needed information (i.e., three columns with a 25 can maximum capability instead of nine columns each having 75 can capability of which only three are used and then only 1/3 full). This usage allows a particular vendor to use a limited number of common screens, even one, to obtain all of the information which is necessary to understand the operating status of a vast number of vending machines, each of which may be of a different type and each of which may be manufactured by a different company.

In addition to presenting the information to the operator visibly on a screen in a uniform manner, the system is able to store data and generate common reports for each machine, again totally independent of the exact nature and/or manufacture of any particular machine. This again is due to the use of the data acquisition unit to provide for common signal information for all machines. Due to this, the report information which can be developed can be supplier specific irrespective of the exact nature of the goods. For example, the need for a given quantity of pop/containers, candy, and coffee for a given location can be printed out in the same list independent of the actual machines needing such inventory. For additional example, the number and type of alarms in a wide geographic area could be printed out. Further example the specific inventory needs and optimal route assignments for a particular vendor operator. Similarly as previously set forth, a single database can be used for all incoming information, such database amenable for manipulation by software in any manner desired by the operator. This allows the use of value added services without the necessity of developing a unique program for each particular manufacturer's particular type of machine. Further, common summaries can be developed across the entire database by the operator.

It is preferred that the database have sufficient names and fields to handle information from the most complex vending machine in a given system. Due to the use of common signals for every vending machine, these fields would be automatically filled with data from the system. Additional fields could include for example the type and nature of the specific vending machine, its physical location by street address, and physical placement, the communication standards for such machine including route, link name, identification and number, the nature and pricing of the varied items of inventory, the various alarms available together with their triggering points (upper and/or lower), and importance (i.e., automatic transmission on occurrence enablement), together with other programmed elements.

It is preferred that the data processing, for example the graphic display on the screen and the processing software, be programmed to ignore non-active names and fields. For example for a three column pop machine, a 49 inventory item capable system would preferably ignore the 46 empty fields in producing the screen images and any reports for this machine. For additional example no compressor or temperature icon would be utilized for a dry snack machine. A separate database having information that can be called up by the identity of a particular machine could be utilized to initially set up the data processing standards for that machine.

In all systems, it is possible that reports be generated and inventory replaced in multiple unit container multiples (for example the archtypical 24 can pop box). This reduces odd lots while maximizing operator convenience.

Turning now to FIGURE 14, a diagram of a typical user interface produced by the central computer system is shown.

The central computer system provides a display of each vending machine being monitored (pop/container machine shown). With other types of vending machines (phone, snacks, cigarettes, etc.) it is preferred the display reflect the type of vending machine. Typically a limited number of universal displays will provide the required information as set forth previously. Indeed, due to the common signal content, a single screen could be utilized (preferably as set forth automatically adapted by available data so as to present only pertinent information).

The display 450 disclosed includes various icons and images that are representative of the elements of vending machines. Preferably these icons have an appearance intuitively similar to the items that they represent (example later given). Due to the common elements in vending machines, a minimum number of icons need be utilized. The particular icons utilized can be automatically generated by software based on database information or can be separately entered.

The particular display 450 disclosed includes a vending machine icon 452, which looks like the vending machine itself. This enables even the most unskilled operator to appreciate the status of that particular machine. The specific icon 452 discussed includes a series of columns each having a column count box 456 that indicates the number of product in the column, as well as a bar graph 458, which visually indicates how the number of cans in the column compares to the length of the column. Preferably, the number of columns and/or rows displayed for a given machine are equal in number to the actual number of columns and/or rows, with the bar graph at 100% when any particular column at the machine is full. This type of presentation is easily developed from the generic type of machine, the number of columns, and/or the total maximum number of containers per column and/or as entered on initial set up of the computer. For example, in a pop/container machine, the selection of the pop/container machine would initially develop a display having a default number of columns (and no rows) each with a certain default maximum number of containers. The entering of the actual number of columns and/or rows would alter the default display to the actual number of columns and/or rows (for example from 12 columns down to 6 columns). The entering of the actual number of maximum containers would likewise alter the default display respectively (for example from 75 down to 50). The bar graphs per item would remain at 100% until further manual or automatic (i.e., in use) input modified the number of cans per column. This use of defaults is preferred because it provides the operator with a usable (albeit not optimized) system with a minimum of inputs. Alternate schemes could be used including not presenting any columns and/or rows and/or any number of containers until the proper data is available. In any event, it is preferred that the display be automatically generated from a single subroutine having variable inputs. It could also be developed automatically from a pre-installed database by the entering of a specific make and model vending machine.

With altering input of other generic types of machines, other initial displays will be developed, displays that could be different than a column type display. For example, a generic type snack machine might have many options developed in an X by Y column/row matrix (for example 7x7), with the display having 3d type bar protruding out of the screen in a step manner (number of snacks at the end of each bar) while a generic type cigarette machine might use only columns like the example pop/container machine. In addition, the displays could have either or both decreasing or increasing indicators. As an example of the latter, a hotel might as a courtesy extend to a guest a credit of $50.00 worth of services or supplies on the guests room card key before room payment. As the guest bought pop or used the phone, this initial $50.00 credit could appear as an increasing bar, indicating the total usage. The charges could also be billed directly to the room (possibly subject to an upper limit). In addition, in this case, warning indicators might appear at the top of the bar not bottom. Thus the displays, although of a few generic types, might differ in actual presentation.

In general, columns are preferred subject to screen resolution limitations.

Note that historical type information can be presented in the display. This could occur by presenting multiple graphic displays showing vend cycles over time on a single screen (in narrow columns), by requiring an operator to click on a particular column to display multiple columns showing historical data in respect to that particular item, or otherwise.

The icons that are developed in the graphic representation are preferably accomplished dependent on the available active data and/or the programming of the machine. These include as follows:

The particular display 452 shown includes a power icon 460 that represents a power connection to the vending machine. If power is interrupted, the icon 460 will flash to the user thereby informing the user that the remote vending machine is without power. This type of sensor and indication and others would be common to most electrically powered machines.

The particular machine disclosed is a pop/container vending machine. Other types of machines, vending goods, and/or services could be utilized. This type of pop/container machine normally includes a compressor. Abnormal cycling of a compressor, either low or high, is an indication of either a refrigeration loop or general machine malfunction. In addition, the cost of electricity for operation increases. For this reason, preferably a compressor cycling sensor and indicator is included in devices utilizing temperature altering mechanisms (i.e., cold or hot). A compressor icon 460 that represents a compressor is graphically illustrated in the display and has located below it a compressor cycles box 464 indicating the number of compressor cycles completed in a 24-hour period. Should the number of compressor cycles exceed or be less than predefined limits as set by the alarm response bytes described above, the compressor icon 460 will light.

In the particular machine disclosed, loss of refrigeration will not potentially cause injury. However, most people prefer cold pop to warm pop. For this reason, a temperature sensor and indicator is preferably included in the system 10. This type of sensor would be utilized with most machines containing temperature changing devices.

A thermometer icon 466 is provided to indicate when the temperature is out of a predefined range. Again, if the temperature range is abnormal, the thermometer icon 466 will flash.

Other types of universal sensors could also be utilized with many differing types of vending machines. Examples of these in the preferred embodiment disclosed include: A coin icon 468 represents when exact change is needed. If the coin icon 468 flashes, a user knows that the change counter is out of change. A key icon 470 representing an unauthorized entry flashes when the door to the vending machine is opened and either no PIN or an incorrect PIN was entered. A communications icon 472 represents the communications link between the remote vending machine and the central computer. If the icon 472 flashes, a user knows that there is some problem with the communications link.

In all instances it is preferred that the operator be able to set the levels a respective alarm activates as well as whether the alarm is automatically sent or merely stored for routine transmission at the regular time. This allows an operator to custom design his system to his own specifications. (The alarms could also be ignored at the computer 15 subject to display on calling up the particular vending machine. For example one operator might not be concerned with low inventory of a particular item as long as the machine itself had something to sell, while another operator might be very concerned with low column inventory. For another example ambient temperature in a pop machine causes no damage to the items therein. Thus one operator may choose to not have automatic transmission or recognition of a temperature alarm (although may choose to have a below Y or above X compressor alarm so activated).

The above graphic interface is given by example. Others may be utilized by the invention.

With this type of graphic interface, an operator can rapidly step through a vast multiple of individual displays, each representing a particular vending machine, with the information necessary to establish a condition needing immediate attention (an alarm such as door open) or a condition needing eventual attention (low inventory of an item as indicated by a yellow short bar). The operator can do this intuitively without the necessity of appreciating let alone taking the time to read and interpret an alphabetic/numeric presentation of the same data. Further, due to the common signal and/or universal displays per generic machine, the graphic information will be presented in a non-confusing manner. The operator can subsequently leisurely go through the displays for additional more specific information.