(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 239 342 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **87302465.7**

(51) Int. Cl.⁴: **E05B 49/00**

(22) Date of filing: **23.03.87**

(30) Priority: **21.03.86 US 842684**
10.02.87 US 13089

(43) Date of publication of application:
30.09.87 Bulletin 87/40

(84) Designated Contracting States:
DE FR GB IT

(71) Applicant: **EMHART INDUSTRIES, INC.**
426 Colt Highway
Farmington Connecticut 06032(US)

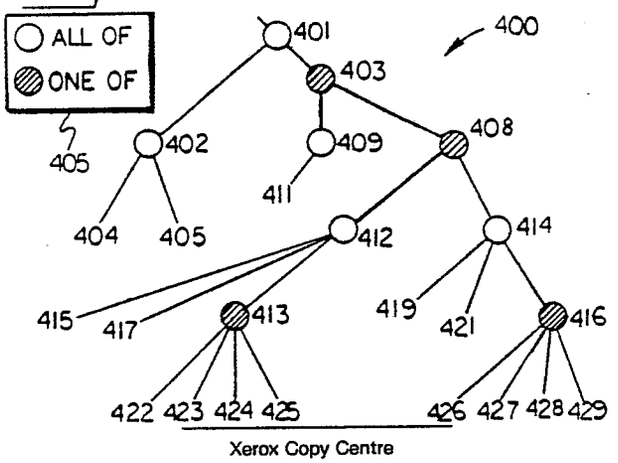
(72) Inventor: **Clarkson, Bruce A.**
106 Dodge Street
Beverly Massachusetts 01915(US)

(74) Representative: **Atkinson, Eric et al**
Emhart Patents Department P.O. Box 88
Ross Walk
Belgrave Leicester LE4 5BX(GB)

(54) Electronic locking systems.

(57) Electronic locking system including keys (30) and self-sufficient door locking units (50) both of which carry multiple "zone" codes (F1,F2...). Upon recognition of a key (30) by a door unit (50), the zone codes within the key and door unit are matched against each other so that a match between any one of the key zone codes (FI) and any one of the door unit zone codes (FJ) will result in an "allow access" decision (366). In the "basic zone" function (350), this decision will permit unlocking of the door (368); in other keying system functions or features (330), additional steps may be required for such unlocking, and the coding of either the key or door unit may be altered (340). The keying system architecture, and method of issuing keys, may be defined in terms of a directed acyclic "door group" graph (400) or equivalent data structure, door groups (401,402...) being defined hierarchically as door units, groups of door units, or groups of door groups. Each node of this graph (400) is identified with a given door group and, except for terminal modes (404,405,415,417,422-429) (which are typically identified with given door units), each such node has an associated "choice rule" (450). In issuing keys (30) the key issuing operator, working from a given door group (401) as the starting point, traverses subordinate nodes (402,403...) subject to limitations and decisions imposed by the choice rules (450), until only terminal nodes (404,405....) remain --thereby defining the coding of a particular key.

Fig. 7



EP 0 239 342 A2

ELECTRONIC LOCKING SYSTEMS

The present invention relates to electronic locking systems, and more particularly to a management system for the door locking units and keys of an installation, said door locking units and keys carrying codes which are electronically processed in a given door locking unit in response to a given key,

Electronic security systems have been well known for many years, and in recent years have seen a wide variety of approaches to designing electronically controlled door locking systems. Some of the early commercial systems have required a hard-wired connection between a central processor and the electronics of the locking systems of given doors. A disadvantage of such systems is the need to install connections between the central controller and the individual lock assemblies. Another variable of these systems is the ability to "re-key" at the door, i.e. to change the coding of the key where such recoding is required by a given keying system function.

A principal concern in the design of such locking systems is the security afforded by the system --in the sense of the ease with which an unauthorised party can determine keying system information which could be used to open one or more locks, rather than the sense of the mechanical integrity of the locking mechanisms. System specifications relating to such security include the number of possible values of the keying system codes, the ability to update such codes, and the effect on the management system of a given installation if security has been breached. The keying systems of the present invention are used to particular advantage with electronic lock technologies which provide a large number of possible values per code and which permit re-keying at the door locking system.

Other significant aspect of such systems are the convenience with which keys may be issued, permitting the use of relatively unskilled personnel as key issuing operators. One also wants sufficient flexibility in designing a keying system for a given installation to afford various levels of key-issuing authority, as another aspect of system security.

It is the object of the present invention to provide an improved management system for an electronic locking system, which management system is flexible and adaptable, suitable for large, complex installations requiring a variety of keying system features,allows the system designer to provide for more than one level of key-issuing authority.

This object is resolved, in accordance with the present invention, in that the system comprising means for storing a database representing the configuration of door locking units of said installation in the form of a directed acyclic graph of "door groups", means for accessing said database to assign codes to given door locking units and door keys, and means for encoding door locking units and keys with the assigned codes.

The invention further provides a method of keying the door locking units and keys of an installation, said door locking units and keys carrying codes which are electronically processed in a given door locking unit in response to a given key characterized in that an acyclic directed graph of "door groups", of equivalent data structure, is defined, wherein "door groups" consist of door locking units or sets of door locking units, and said graph is based upon the configuration of door locking units of said installation, and in that codes are assigned to door locking units and to keys, using predefined code assignment rules responsive to the configuration of said door group graph.

Other features of the invention will be found set out in the appended sub-claims.

The above and additional aspects of the invention are illustrated in the following detailed description of the preferred embodiment, which should be taken in conjunction with the drawings in which:

Figure 1 is a schematic drawing of an electronic locking system for a given door, in accordance with an illustrative embodiment of the invention;

Figure 2 is a block schematic diagram of electronic logic circuitry for the door locking system of Figure 1;

Figure 3 is a flow chart schematic diagram of a basic operating program for the electronic logic of Figure 2;

Figure 4 is a flow chart schematic diagram of a Basic Zone/One Use Subroutine for the electronic circuitry of Figure 2;

Figure 5 is a perspective view of an illustrative design of key/door unit initialisation console;

Figure 6 is a schematic view of a preferred management system configuration for the electronic locking system of Figure 1, embodying the console of Figure 5;

Figure 7 is a schematic diagram of a door group graph in accordance with the invention; and

Figure 8 is a partial plan view of a dormitory floor plan, corresponding to the door group graph of Figure 7.

It should be understood that the present invention resides in the design of improved keying systems which may be utilised in connection with a wide variety of electromechanical locking system technologies. For the purpose of illustration, the keying system of this invention is illustrated below in connection with an electronic locking system of the recodable key variety (i.e. wherein the "key" is similar to design to a traditional mechanical key), but the invention is easily adapted to card-reader technologies such as so-called "smart cards", to optically based systems, etc. Advantageously, such locking systems should be capable of storing a large volume of data, permit the use of multiple codes within the door locking unit and the "key", and permit the recoding of the key at the door unit when appropriate.

Fig. 1 shows highly schematically the principal elements of the electronic locking system, in which a key 30 is inserted into mortise lock cylinder 50 to open the lock. Cylinder control circuitry 100 within cylinder 50 recognises the full insertion of key 30 and extracts electronically encoded information from the key memory 40 via key connectors 45 and cylinder connectors 59. Control circuitry 100 stores and processes keying codes received from key memory 40 as well as resident cylinder codes. The control circuitry 100 can alter not only the codes in key memory 40 based on data transmitted from cylinder 50 but also codes stored within the cylinder based on data from key memory 40.

The processing of access codes from the key and cylinder by control circuitry 100 results in a decision to grant or deny access. If an "authorised access" decision is made, release mechanism 70 receives a drive signal from control circuitry 100, causing it to withdraw a radially oriented locking pin 72 from cylinder plug 55. A user may then turn key 30 to rotate cylinder plug 55 as in a mechanical mortise lock, and rotate a cam (not shown) to release a door locking mechanism. Cylinder 50 also houses a key centring and retention device 90, which interacts with a single bit 37 or notch in the key to ensure the proper location of key 30 within keyway 57.

Fig. 1 shows highly schematically the principal elements of the electronic locking system, in which a key 30 is inserted into mortise lock cylinder 50 to open the lock. Cylinder control circuitry 100 within cylinder 50 recognises the full insertion of key 30 and extracts electronically encoded information from the key memory 40 via key connectors 45 and cylinder connectors 59. Control circuitry 100 stores and processes keying codes received from key memory 40 as well as resident cylinder codes. The control circuitry 100 can alter not only the codes in key memory 40 based on data transmitted from cylinder 50 but also codes stored within the cylinder based on data from key memory 40.

The processing of access codes from the key and cylinder by control circuitry 100 results in a decision to grant or deny access. If an "authorised access" decision is made, release mechanism 70 receives a drive signal from control circuitry 100, causing it to withdraw a radially oriented locking pin 72 from cylinder plug 55. A user may then turn key 30 to rotate cylinder plug 55 as in a mechanical mortise lock, and rotate a cam (not shown) to release a door locking mechanism. Cylinder 50 also houses a key centring and retention device 90, which interacts with a single bit 37 or notch in the key to ensure the proper location of key 30 within keyway 57.

The "chip-on-a-key" design of Figure 1, and other recodable-key designs such as smart cards, make use of electronically alterable integrated circuit (IC) technologies.

Electronically alterable key memory 40 has the ability to store a substantial number of access codes, each of which will have a much larger range of possible values than found in traditional mechanical locks. This non-volatile integrated circuit technology involves memory which may be read like traditional read-only-memory (ROM) and may be written to after being electronically erased. Such memory devices are commonly known as EEPROM integrated circuits. EEPROM is a medium density memory, which retains adequate key memory within devices in the order of 2-3mm micron geometry. To store data in such devices, the word must be erased and then written. An alternative integrated circuit technology which may be employed for the key memory 40 and cylinder memory 180 is so-called EPROM.

Fig. 2 is a block schematic diagram of the logic circuitry constituting the cylinder control circuitry 100, which supervises the various electronic functions of lock cylinder 50. Control circuitry 100 is a microprocessor-based system including central processing unit (CPU) 105 as its central element. Other major components are key serial interface 110, which provides synchronous serial communications of access code data to and from the key memory 40, timing circuitry 120, which provides various timing signals, key sensing circuitry 150, which produces signals indicative of the full insertion of a key in keyway 57, and of the withdrawal of the key, power control circuitry 140, which regulates the delivery of power from battery 68 to the various elements of the control circuitry 100, and release driver 130, which outputs actuating signals to the release mechanism 70 in response to an appropriate command from CPU 105. Under the supervision of power control circuit 140, the control circuitry 100 undergoes various states of power distribution to the various subassemblies with a view to low power consumption.

Control circuitry 100 also encompasses various types of memory, including random access memory (RAM) 160, read only memory (ROM) 170, and electronically alterable memory (EEPROM) 180. RAM 160 receives data from key interface 110 and permits high-speed processing of this data by CPU 105. ROM 170 stores the firmware for the control circuitry; certain routines are explained below in the discussion of the lock's keying system. EEPROM 180 comprises non-volatile memory for the access codes resident in cylinder 50 and may take the form of any of a number of commercially available devices.

In one embodiment of the invention, timing circuitry 120 also comprises a real time clock to provide a time-of-day signal, i.e. a resolution of some number of minutes. Illustratively, this clock takes the form of a dedicated clock IC. The energy source 68 (Fig. 1) is designed to provide continuous input power to this clock IC. The inclusion of such clock significantly affects the access code memory structure, and keying system firmware, as discussed below.

Fig. 3 is a high-level flowchart of the basic operating program 350 for cylinder control circuitry 110, which is resident in ROM 170 (Fig. 2). At 351 the key sensing circuitry 150 detects the valid insertion of a key, causing power control circuitry 140 to provide power to CPU 105 and key 30, at 353. At 354, the logic selects a suitable communication protocol for key serial I/O 110 (Fig. 2); different protocols would typically be required for normal key 30 and for a cylinder recombinating device 355 (shown in Fig. 5 and discussed below. At 356 the key serial I/O reads data from the key memory 40 into RAM 160.

As further explained below, the key and cylinder memories are structured in the preferred embodiment in a plurality of "zone code" keying data F1, F2...FN. In the illustrated program, data is read from the key at 356 on a code-by-code basis. At the case block comprised of step 358 and steps 359...361 and 364 the program selects the appropriate function subprogram stored in ROM 170 based upon a function associated with the given zone code and interprets the just-read key codes. Depending on the nature of the particular subprogram, this interpretation process may result in an "authorise access" decision, may yield data which is intended to be delivered to the key or key-like device (such as for recombinating a key 30 or for providing information about cylinder 50 to a clerk console 350), and may result in commands to re-code the cylinder memory 180. Cylinder re-coding, if required, advantageously takes place at this stage. At 362, the CPU tests the key data in RAM 160 to determine whether an "end of data" flag is present, while at 364 the redundant check codes in the key data are analysed to confirm the valid key data had been received. A failure of the latter test causes the re-reading of the invalid key data.

At 365 any output codes resulting from the prior processing of the key codes are written to the key or key-like device (e.g. to change one or more zone codes of a key 30). At 366 the CPU determines whether the function processing had resulted in an "authorise access" state, and if such a state is present actuates the release driver 130 to open the lock. In the absence of an "authorise access" flag the system enters a "time out" state at 367, wherein the timing circuitry 120 clocks a predetermined time interval during which the key sensing circuitry 150 is not permitted to output a valid key insertion signal. Time out step 367 limits the frequency with which an unauthorised user can feed a large number of random codes to the control circuitry 100 using a key-like device. The time out state may be effected after a prescribed number of key insertions. At 369 the power control circuitry 140 turns off the supply of power to CPU 105 and release driver 130.

Table 1 shows an advantageous memory map for access codes contained within the cylinder or door unit EEPROM 180 (Fig. 2). This memory map schematically illustrates the logical addressing scheme of the lock's control program to sequentially retrieve data from memory cells within EEPROM 180, but does not necessarily depict the physical layout of such memory cells. Memory 180 includes various fixed format fields - fields with a predetermined number of assigned data bits - and a variable format portion for function storage. Fixed format fields includes a "door unit identification" - a serial number that identifies the particular cylinder 50, but has no security function - and the "programming code", a security code which must be transmitted to cylinder control circuitry 100 in order to allow modification of memory 180, as discussed below. Other fixed format fields not shown in Table 1 may be included depending on the requirements of the door unit firmware. The function storage fields contain the data associated with the particular keying system functions programmed into cylinder access code memory 180; this is illustrated below in Tables 2 and 3.

Illustratively, key memory 40 is structured similarly to the cylinder code map of Table 1, but omits the programming code field.

Table 2 illustrates the record structure of a particular keying system feature - i.e. the zone function. In its basic embodiment, the zone function implements a comparison of each of a set of key zone codes with each of a set of cylinder zone codes, and permits access if any match occurs. The header byte of this memory map gives the number of zone function records (here four). Together with preknowledge of the memory occupied by the records of each function, the header byte enables the addressing routine to scan

through logical memory to locate the next function within function storage (Table 1). In each record, the code combination represents the code which must be matched to initiate the corresponding function. The status bits S1-S5 (five are shown for purposes of illustration are associated with specialised zone features, so that the setting of a particular use bit (at most one is set) identifies the code combination with that feature. For example, S1 might be associated with "one use" - which allows keys to be issued for one time use only; and S2 might be identified with "electronic lockout" - permits a special lockout key to prevent access by normal keys, until the lockout key is reused. If no status bit S1-S5 is set, the code combination will be a Basic Zone code which yields an "authorise access" decision upon matching of key and door unit zone codes, without further processing.

As explained above, "zones" are the basic data which are matched prior to granting access. Each key and each door unit advantageously contains multiple codes, each code representing one zone. In terms of the keying system, each zone may be defined as a collection of door units and keys which share a common code. Storage is required in door units and keys for each zone in which it participates --i.e. storage limits correlate with the number of types of access to be granted. This storage scheme is much superior to having each door unit store a list of authorised keys, or conversely having each key store a list of authorised door units, both such schemes being vulnerable to security breaches, and being wasteful of storage capacity.

In the key memory 40 and cylinder memory 180, access codes are assigned a given code width (number of binary digits per code) which determines by inverse relationship the total number of available codes in EEPROM. Higher code widths will decrease processing speed, but increase the resistance of the system to fraudulent access attempts by means of random codes electrically fed to the lock; in addition higher-width codes are less likely to be inadvertently duplicated in system management.

Tables 3 and 4 give simplified record structures for cylinder and key memory function storage fields for basic zone and one use functions, and should be referenced together with the flow chart schematic diagram of Fig. 4 to illustrate the relationship between the access code memory structures and the associated keying system software routines in ROM 170. The door unit or cylinder record structure includes three zone records with associated "one use" status bits S1 (Table 3), while the key memory structure contains five zone records but no associated status or use bits (Table 4).

In the basic system program of Fig. 3, as part of the "select functions" case block, the control firmware includes various subroutines associated with particular keying system features, including the "basic zone/one use subroutine" of Fig. 4. This routine includes nested loops wherein key pointer I (e.g. pointing to a particular record or row: see Table 4) and cylinder pointer J (e.g. pointing to a given cylinder zone record: see Table 3) are each incremented from 1 to the respective "number of records" value. For each pair of values, I, J, this routine compares the "code combination" for the relevant cylinder and key zone records at step 335. If a match is found the program determines at 338 whether the CYL.S1 flag for the relevant record J is set. If this "one use" flag is not set, the routine simply returns a "grant access" decision at 341. If the flag is set, however, the routine first updates CYLCODE (J) with a pseudorandom number generated by the management system; this prevents a repeated use of the key to open the same lock cylinder.

Were the zone function data structure to take the more complicated form shown in Table 2, the subroutine of Fig. 4 would be modified to determine whether any of the other status or use bits S2-S5 were set, and to include appropriate algorithms to implement these additional keying system features.

The locking system of the invention can achieve all of the traditional keying system features found in mechanical mortise cylinders (e.g. great grand master keying, cross-keying etc.), as well as additional useful functions. Furthermore, the cylinder access code memory 180 can include updating key codes, which may be written to the key memory 40 in implementing certain keying system functions. Specialised keying system functions may be designed to control unauthorised copying of key codes, and in general to selectively update the key memory 40 for enhanced flexibility together with security.

It is sometimes necessary to change one or more zone codes by virtue of the nature of a given keying system feature, because of a need to recode door units due to a discovered breach of security, or for other reasons.

Advantageously, the keying system associates with each zone code a meaningful zone code identifier based upon the characteristics of the door units and keys which carry such code, in order to facilitate this recoding process. In addition, the keying system may utilise a special code to "prime" given keys and cylinders for recoding.

In the embodiment in which the cylinder control circuitry 100 includes a real time clock, the keying system can be extended to include time-of-day control. Time-of-day can be associated with each keying function. For basic zone/single use, a time can be associated with each door unit zone (i.e. set of lock cylinders containing a common zone code). The keying system functions could be modified to include one

or more time access windows, to include automatic cylinder recording at a given time of day, and other features. The cylinder memory structure must be supplemented with time-of-day codes, i.e. one byte for each significant time interval within a day. By including a calendar timing device in the timing circuitry 120 (Fig. 3), the principles discussed above can be applied to keying system features tied to particular days, weeks etc.

The electronic locking system of the invention may be incorporated in "hard-wired" electronic lock installations, comprising a communication network linking the various lock cylinders and a central management system processor. In the preferred embodiment of the invention, however, the lock cylinder 50 comprises a stand-alone system, with no hard-wired communication. The IC packages 41 (or 42) within each key 30 serve as a substitute for a direct communication link with a central controller, inasmuch as the key can be encoded at a remote station to transmit codes to lock cylinder 50. Key 30 can be encoded with special codes which are recognised by cylinder access code memory 180. As shown in Fig. 5, the management system advantageously includes one or more key/cylinder consoles 250, which may take the form for example of a portable microcomputer with specialised input/output devices. Key receptacle 252 accepts insertion of a key 30, and links the inserted key to internal logic circuitry for initialising or recoding a key. Cylinder recombinating device 255 includes a key blade 256 similar to a normal key blade 33 (Fig. 8), and a plug 257 which mates with an outlet (not shown) at the rear of console 250. Thus, the portable consoles 250 may be carried to given door units 50 for the purpose of initializing or recombinating the door unit memory 180.

The management system is advantageously adapted to the requirements of institutional users such as hotels and universities. With reference to Fig. 6, the system might include a plurality of "clerk consoles" 250a-d in accordance with the device of Fig. 5, which communicate with a central controller 260. Controller 260 acts as the central repository of the management system data base for the entire installation and downloads data into the various consoles 250a-d. In particular, controller 260 tracks all zone codes which have been assigned, together with the identities (e.g. door unit identifiers - Table 1) of all keys and door units which carry each zone code. Consoles 250a-d encode keys as required by the keying system data base and records to whom they are issued. A given console 250 can interrogate the central controller 260 to inspect the central database; sensitive information can be protected by features such as passwords and other techniques (see references below to operator authorisation, key classes, etc.) This preferred management system may be characterised as a distributed processing system, with all real time processing effected at individual lock cylinders 50.

Where the timing circuitry 120 includes a real time clock, it will be appreciated that the key/initialisation console 1350 and central controller 1360 must have the ability to keep time-of-day in operating the management system.

Preferably, the keying system incorporates a set of rules for logically and efficiently assigning zone codes to door units and keys. In a computer-automatable keying system, the keying system architect establishes a logical structure ("architecture matrix") within the framework of certain design rules, such logical structure being based upon the geographic and functional characteristics of the door units of a given installation. In the preferred embodiment of a computer-automated keying system, the architecture matrix may take the form a multidimensional array including one or more parameters representing the configuration of door locking units. Advantageously, such parameters include "choice rule" limitations governing the key-issuing operator's assignment of door units and features to a key being issued. Finally, the system automatically assigns zone codes to the issued keys in accordance with pre0established assignment rules, based upon the door unit and feature assignments. In the above scheme, the architecture matrix design rules and the rules for interpreting a given architecture map are pre-established, and the keying system architect generates or modifies a given architecture matrix in accordance with the requirements of a given installation.

An advantageous methodology for establishing and interpreting architecture matrices is discussed below.

In preferred embodiment of the invention, the process of designing a keying system for a given installation, and the issuance of keys and other aspects of the management system, rely upon an architecture matrix based upon "door groups", which in the simplest case may be defined as "a door unit or collection of door units". To permit a hierarchical definition of "door groups", however, this term is broadened to include "groups of door groups". Each door group has an identifier chosen by the keying system architect, most desirably a name related to local geography or function. By contrast, the code assigned to given zones is desirably chosen to have no obvious relationship to its related doors, door groups, and keys, thereby reducing the impact of disclosure of the key data to an unauthorised party.

The open-ended definition of "door groups" affords tremendous flexibility in system design --these may be defined by a keying system architect based only in the requirements of a given installation, unconstrained by the particular keying system employed. Optimal use of this scheme calls for multiple zones in both the door and key.

As noted above, the invention provides a logical scheme for designing a keying system architecture matrix for given installations, using the concepts of "door groups". As is illustrated below, the keying system architect defines a "door group graph" in the form of a "directed acyclic network" in which each node represents a door group; in this logical scheme, then, each door group corresponds to a unique node of the graph. A directed acyclic network, as used in this specification means a hierarchial graph or network consisting of nodes with interconnecting edges or branches, each edge having a defined direction; when this graph is traversed from node to node along the defined directions, one never returns to an earlier-encountered node. Such networks typically consist of "trees" in the mathematical sense, i.e. acyclic graphs wherein each node has only one immediately preceding (or predecessor") node, but also include graphs wherein multiple branches may meet at a single "successor" node. In issuing keys using an established keying system architecture matrix (door group graph), the key-issuing operator begins at an authorised starting node, as explained below, and traverse the subordinate nodes according to defined "choice rules" until only terminal nodes --typically identifying particular door units --remain.

"Choice rules" are a set of rules one of which is assigned to each node of a door group graph except terminal nodes, in order to govern the key-issuing operator's decision in selecting none, one or more of the immediate subordinate node(s). An advantageous set of choice rules to be utilised in defining a keying system architecture might include the following:

(1) All of

(2) One of

(3) Each of (this has essentially the same effect as "All of", but has certain advantages discussed below)

(4) Any of (i.e. none, any one, or more than one)

(5) At least one of

Of the above choice rules, the first two cover most decisions which the keying system architect would need to establish; these impose strict limitations on the key issuing operator's decision (in fact, the first rule requires no decision). The third rule, "each of", has exactly the same effect as "all of" from the perspective of the key-issuing operator, but has certain security advantages discussed below. "Any of" provides the most flexibility in choosing door groups. "At least one of" is similar to "any of", but precludes the possibility of choosing no door groups; this could be used for example to permit the issuance of one or more locker door keys to a patron of a gymnasium.

The "All of" and "Each of" choice rules both have the effect of including all subordinate door groups, but are distinguishable in the assignment of zone values to the immediately subordinate door groups. "All of" typically assigns a common zone value to such door groups, while "Each of" gives each such door group a distinct zone value. The effect of this may be illustrated by considering the issuance of a security master key. If a dormitory resident (for example) were able to determine the zone code for his room's door unit, which door unit could also be opened by the security master key, he would have the zone code for the security zone. This information could possibly be used to help him gain access to all door units accessible by the security master key. However, by using the "choose each of" rule the domain of each zone of the security master would be partitioned, so that the dorm resident would have access only to a known subset of the security master door units, limiting the potential for vandalism etc. Note that "one of", "any of", and "at least one of" all require such partitioning.

The key-issuing process may be implemented using a management system such as that illustrated in Figure 6 as an "on-line" system wherein a series of "screens" are displayed at a clerk console 250 to guide the key-issuing operator through relevant portions of the door group graph, presenting the appropriate choices, and possibly reporting the results of the decisions. This graph-traversing process is continued node by node until only terminal nodes remain. At each node encountered in the issuing of a particular key in which a decision is required, a menu would be displayed appropriate for the choice rule for that node, and the door groups for that node and the successor nodes identified using nomenclature created by the keying system architect (desirably according to key function or local geography). The key-issuing operator only implements the decisions previously established by the keying system architect, which decisions may be designed to ensure proper security, completeness, and reasonableness of issued keys.

Figure 7 illustrates a given portion of a door group graph for an academic institution, i.e. a graph 400 for issuing keys to dorm residents. In referring to both Figure 7 and the dormitory floor plan of Figure 8, TABLE 5 should be consulted to determine the significance of the various door groups and door units shown in these figures. In Figure 7 the starting point, node 401, represents the door group consisting of all doors of a given dormitory. By virtue of the "all of" rule at node 401, the issuer must branch to both the "Entrances" door group 402 and the "floors" door group 404. At the "Floors" node 4040, the operator must elect between floors 1 and 2 (nodes 408 and 409). By tracing this graph until the terminal nodes, according to the choice rules 450, with reference to dormitory floor plan of Figure 8, it will be seen that the issued key will be coded for both the front and the rear entrances, for one of the floors (including the stair well 511 for a second floor resident), for the entrance and bathroom doors to a given suite, and for one of the rooms of that suite. The key-issuing program might, for example, display screens corresponding to nodes in the sequence 401, 402, 404, 405, 403, 408, 412, 415, 417, 413, 422 –thereby encoding the key for the front and rear building entrances, the entrance and bath doors for suite 101, and door of room A of that suite.

The geometrical model of keying system architecture matrices as directed acyclic graphs has equivalent data structures which lend themselves to computer automation of the key-issuing process; as used in the specification and claims, "equivalent data structure" indicates any data organisation of door groups which has an equivalent logical significance as the directed acyclic door groups graphs. TABLE 6 gives an indented list data structure which is equivalent to the door group graph of Figure 7. This is an indented listing wherein each of the door groups represents one element of the list. This list may be ordered according to the "counting index" shown, or any of a variety of other sequences known from graph theory. Each door group is assigned a "level index" representing its level within the keying system architecture hierarchy –this is visually represented by the degree of indentation of the door group name. Each door group has an associate choice rule, except for terminal door groups (i.e. door units). The key-issuing program can process the elements of such listing in the sequence of the counting index, or any of a variety of other sequences known from graph theory, and at each level applies the choice rule to the elements at the successor level. One can therefore define a door group graph using a data structure including the parameters V, the nodes of door groups; E, the directed edges connecting these nodes; V_o, the source node; and R, the choice rules associated with each node.

Using the choice rule/zone code assignment scheme described above, the coding in the preceding example (second-to-last paragraph above) would be effected by assigning three zone codes to the key: one encompassing the front and rear entrance doors to the dorm, one including the entrance and both for the suite, and one for the student's bedroom (see TABLE 7). However, it should be noted that the zone concept is an extremely flexible one and that many alternative schemes could be employed for assigning zones to groups of doors and keys. In the above case, for example, the same door units could be assigned to a given key using just two zones, one encompassing the building entrances and one including the individual student's bedroom as well as the entrance and bath door for his suite (see TABLE 8). In the former instance, the common door units for the suite (515, 517) would only hold a single basic zone code shared by all the students, whereas in the latter instance such door units would have to hold separate codes for each student in order to deny access to the other bedrooms of the suite. The choice rules and zone assignment scheme discussed above represents an efficient and flexible approach.

In a preferred embodiment of the key-issuing scheme of the invention, key-issuing operators are authorised to commence the process only at defined door groups. Therefore, the access of each such operator is limited to those door groups and door units including his authorised starting door groups and those subordinate thereto. The system will not permit such operator to view the remainder of the door group graph during key issuance. Privileged operators may be given access to a top node or nodes by which they can see the entire structure.

Keying system architectures in accordance with the invention advantageously make use of "key classes", which in common sense terms correspond to recognisable types of key users. Taking the example of a college dormitory (Figure 8), key classes might include dorm residents, housekeeper, plumber, security personnel, resident adviser, etc. Each individual key class is associated with a defined starting node for issuing keys to members of such class, and each key class would be assigned an independent set of zone codes. Thus, with reference to Figure 7, if two different key classes were associated with the starting node 401, the keying system would include two sets of zone codes in parallel. Therefore, if the zone codes for one key class were divulged, zone codes of other key classes would not be compromised. In the operator authorisation scheme discussed above, the door groups at which key issuing operators may start are advantageously associated with key classes. Other mechanisms for operator key-issuing authorisation are possible, however.

TABLE 1

DOOR UNIT MEMORY MAP

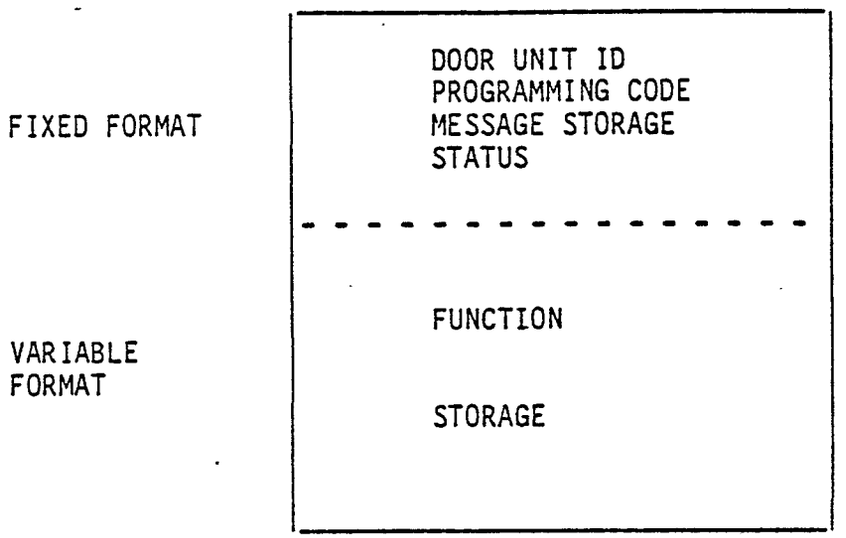


TABLE 2

ZONE FUNCTION MEMORY MAP

NUMBER OF RECORDS					
CODE COMBINATION	S1	S2	S3	S4	S5
CODE COMBINATION	S1	S2	S3	S4	S5
CODE COMBINATION	S1	S2	S3	S4	S5
CODE COMBINATION	S1	S2	S3	S4	S5

TABLE 3

SIMPLIFIED MEMORY MAP
DOOR ZONE FUNCTION

NUMBER OF RECORDS	
CODE COMBINATION	S1
CODE COMBINATION	S1
CODE COMBINATION	S1

TABLE 4

SIMPLIFIED MEMORY MAP
KEY ZONE FUNCTION

NUMBER OF RECORDS
CODE COMBINATION
CODE COMBINATION
CODE COMBINATION
CODE COMBINATION
CODE COMBINATION

TABLE 5

Guide to Figures 7 and 8

DOOR GROUP/DOOR UNIT NAME	FIG. 7-REF.NUM.	FIG. 8-REF.NUM.
GIVEN DORM	401	501
ENTRANCES	402	
FLOORS	403	
FRONT ENTRANCE	404	504
REAR ENTRANCE	405	505
FLOOR 2	409	
STAIRS	411	511
SUPPLY CLOSET		503
FLOOR 1	408	
SUITE 101	412	
ENTRANCE	415	515
BATHROOM	417	517
BEDROOMS	413	
A	422	522
B	423	523
C	424	524
D	425	525
SUITE 102	414	
ENTRANCE	419	519
BATHROOM	421	521
BEDROOMS	416	
A	426	526
B	427	527
C	428	528
D	429	529

TABLE 6

SAMPLE INDENTED LISTING

COUNTING INDEX	LEVEL INDEX	DOOR GROUP NAME	CHOICE RULE
1	1	STUDENT RESIDENCE	ONE OF
2	2	GIVEN DORM	ALL OF
3	3	FLOORS	ONE OF
4	4	FLOOR 1	ONE OF
5	5	SUITE 101	ALL OF
6	6	ENTRANCE	
7	6	BATH	
8	6	BEDROOMS	ONE OF
9	7	A	
10	7	B	
11	7	C	
12	7	D	
13	5	SUITE 102	ONE OF
14	6	ENTRANCE	
15	6	BATH	

TABLE 7

Basic Zone Assignment for a Key

Example 1

Zone 1	Door Units 504, 505
Zone 2	Door Units 515, 517
Zone 3	Door Unit 522

TABLE 8

Basic Zone Assignment for a Key

Example 2

Zone 1	Door Units 504, 505
Zone 2	Door Units 514, 517, 522

Claims

1. A management system for the door locking units (50) and keys (30) of an installation, said door locking units (50) and keys (30) carrying codes (F1,F2....) which are electronically processed in a given door locking unit (50) in response to a given key (30) characterised in that the system comprising means (26) for storing a database representing the configuration of door locking units (504,505...) of said installation in the

form of a directed acyclic graph (400) of "door groups" (401,402...), means (250) for accessing said database to assign codes to given door locking units and door keys, and means (250) for encoding door locking units and keys with the assigned codes.

2. A management system according to Claim 1, characterised in that the directed acyclic graph (400) comprises a plurality of terminal nodes (404,405) corresponding to given door units (504,405...), and non-terminal nodes (401,402...) corresponding to sets of door units, each non-terminal nodes (401,402...) having an associated choice rule (450) governing the possible choices of successor nodes (402,403,404,405...) in traversing the graph (400).

3. A management system according to either one of Claims 1 and 2 characterised in that given key-issuing personnel are assigned authorisation codes which determine the accessible portion (401,402...) of the door group graph.

4. A method of keying the door locking units (50) and keys (30) of an installation, said door locking units (50) and keys (30) carrying codes (F1,F2...) which are electronically processed in a given door locking unit in response to a given key (30), characterised in that an acyclic directed graph (400) of "door groups" (401,402...), or equivalent data structure, is defined, wherein "door groups" (401,402...) consist of door locking units (404,405,415...) or sets (402,403,...) of door locking units, and said graph (400) is based upon the configuration of door locking units (503,504,511...) of said installation, in that and codes (F1,F2...) are assigned to door locking units (50) and to keys (30), using predefined code assignment rules responsive to the configuration of said door group graph (400).

5. A method according to Claim 4, characterised in that the door group graph (400) represents the geographic and functional characteristics of the constituent door locking units (503,504,5121..) of the various door groups (403,404,411...).

6. A method according to either one of Claims 4 and 5 characterised in that the door group graph has terminal modes (404,405,415,417,422-429) each associated with given door units (504,505,...), and non-terminal nodes (401,402,403...) each associated with a plurality of door units, and in that each non-terminal node (401,402,403...) has an associated choice rule (450) governing the possible choices of successor nodes (402,403,404,405,408,409...) in traversing the graph (400).

7. A method according to any one of Claims 4 to 6 characterised in that keys (300) for an installation having a given door group graph (400) are encoded starting at an appropriate starting node (401) of said door group graph (400) and traversing subordinate nodes (402,403...) until only terminal nodes (404,405...), associated with given door units (504,505...) remain.

8. A method according to Claim 7 characterised in that given key-issuing personnel are assigned authorisation levels which determine the permissible set of starting nodes (401).

9. A method according to any one of Claims 4 to 8 characterised in the the door unit and key codes (F1,F2...) comprise "zone codes", so that upon recognition of a key (30) by a door unit (50) each key zone code (FI) is compared with each door unit zone code (FJ), with a positive comparison resulting in an "allow access" decision (366).

10. A method according to Claim 9 characterised in that at least some keys (30) are identified with one or more given key classes, each such key class, identifying a recognisable type of key user, each distinct key class having a distinct set of zone codes (F1,F2...) assigned to keys (30) belonging to it.

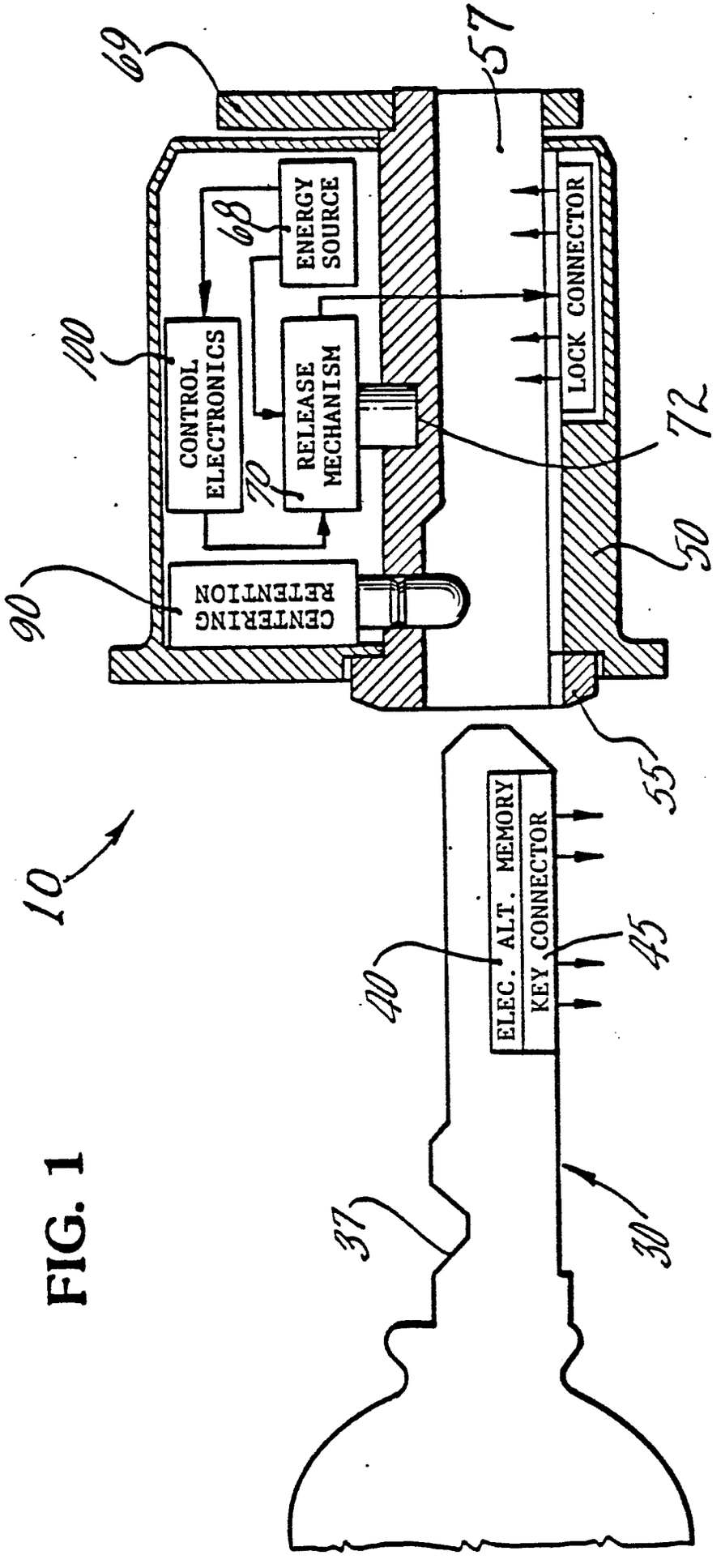

69
57
ENERGY SOURCE
68
100
CONTROL ELECTRONICS
RELEASE MECHANISM
70
LOCK CONNECTOR
72
50
90
CENTERING RETENTION
55
10
ELEC. ALT. MEMORY
KEY CONNECTOR
40
45
37
30

FIG. 1

FIG. 2

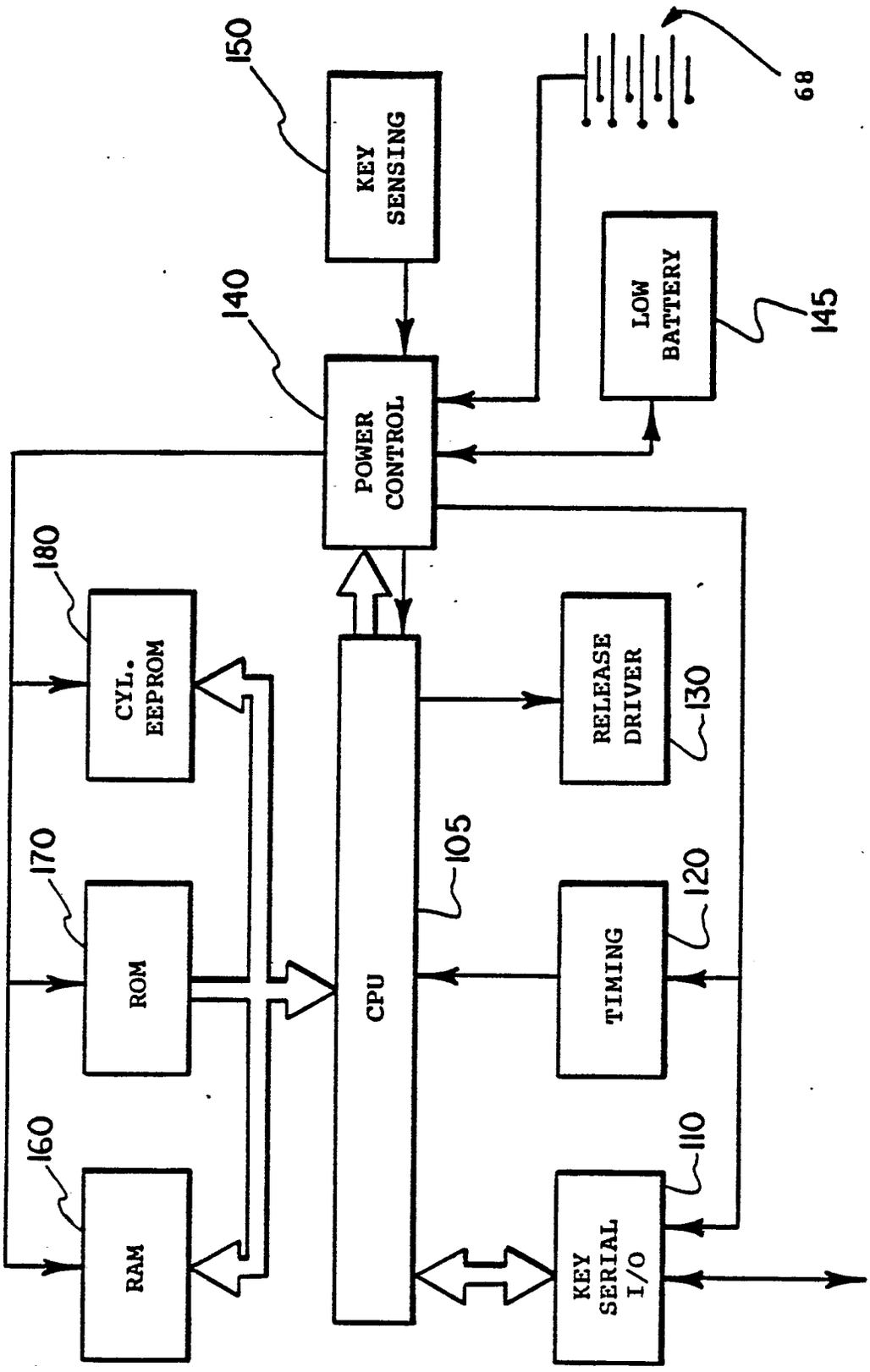

150
KEY SENSING
68
140
POWER CONTROL
LOW BATTERY
145
180
CYL. EEPROM
RELEASE DRIVER
130
170
ROM
105
CPU
120
TIMING
160
RAM
110
KEY SERIAL I/O
TO KEY EEPROM 40

FIG. 3

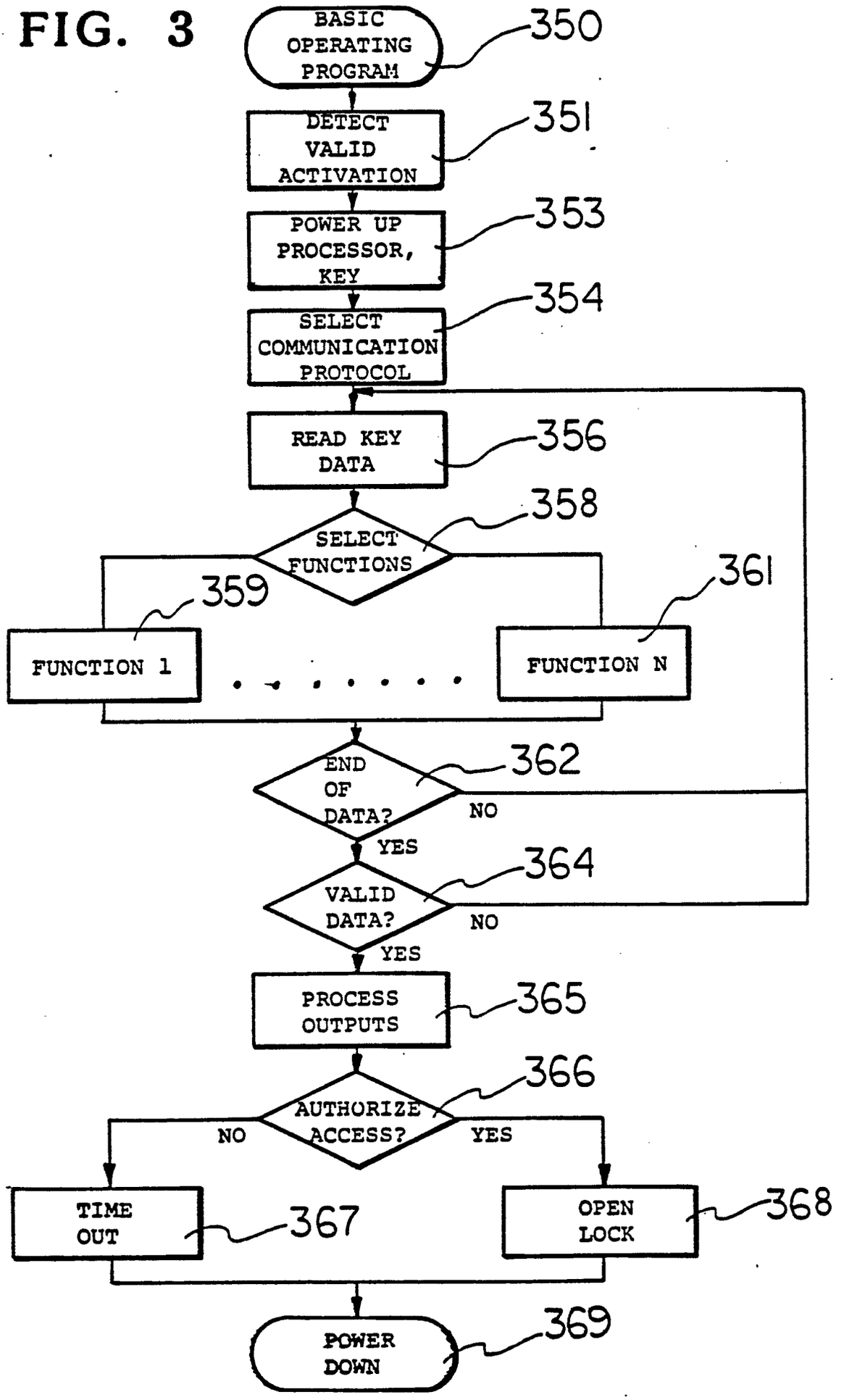
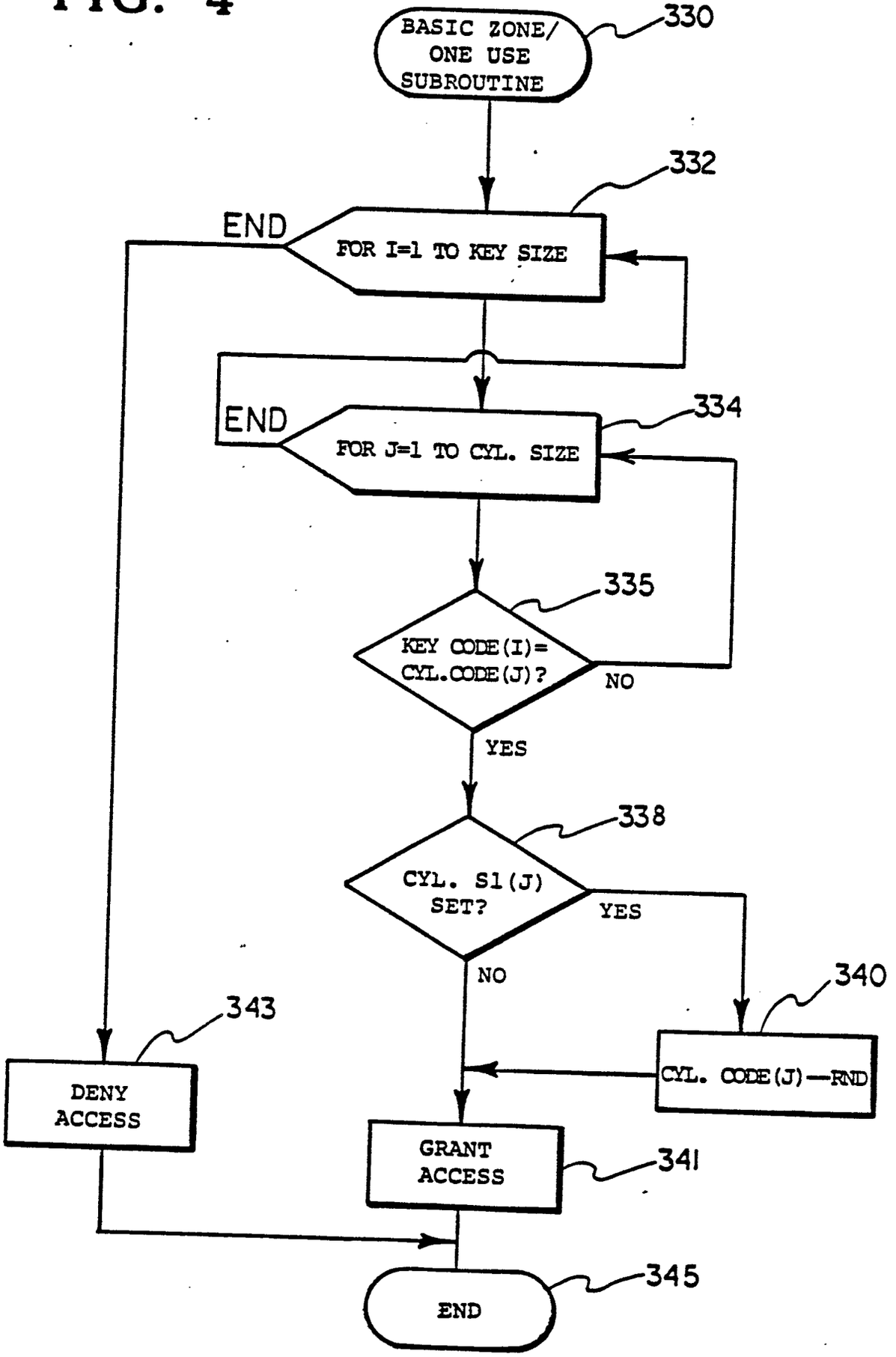

BASIC OPERATING PROGRAM
350
DETECT VALID ACTIVATION
351
POWER UP PROCESSOR, KEY
353
SELECT COMMUNICATION PROTOCOL
354
READ KEY DATA
356
SELECT FUNCTIONS
358
359
FUNCTION 1
361
FUNCTION N
END OF DATA?
362
NO
YES
VALID DATA?
364
NO
YES
PROCESS OUTPUTS
365
AUTHORIZE ACCESS?
366
NO
YES
TIME OUT
367
OPEN LOCK
368
POWER DOWN
369

FIG. 4


BASIC ZONE/ ONE USE SUBROUTINE
330
332
END
FOR I=1 TO KEY SIZE
END
FOR J=1 TO CYL. SIZE
334
335
KEY CODE(I)= CYL.CODE(J)?
NO
YES
338
CYL. S1(J) SET?
YES
NO
340
343
DENY ACCESS
CYL. CODE(J)—RND
GRANT ACCESS
341
345
END

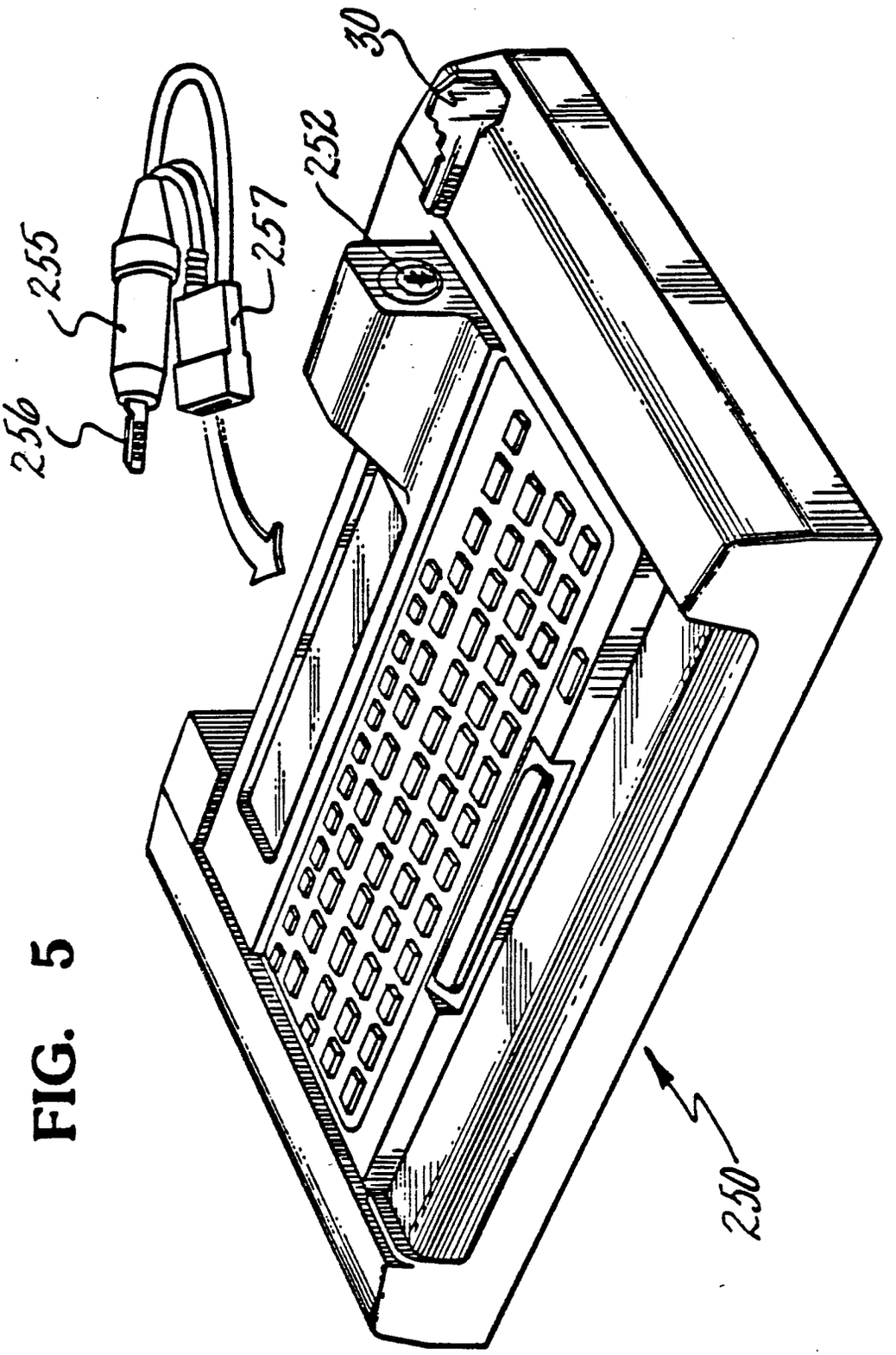


FIG. 5

FIG. 6

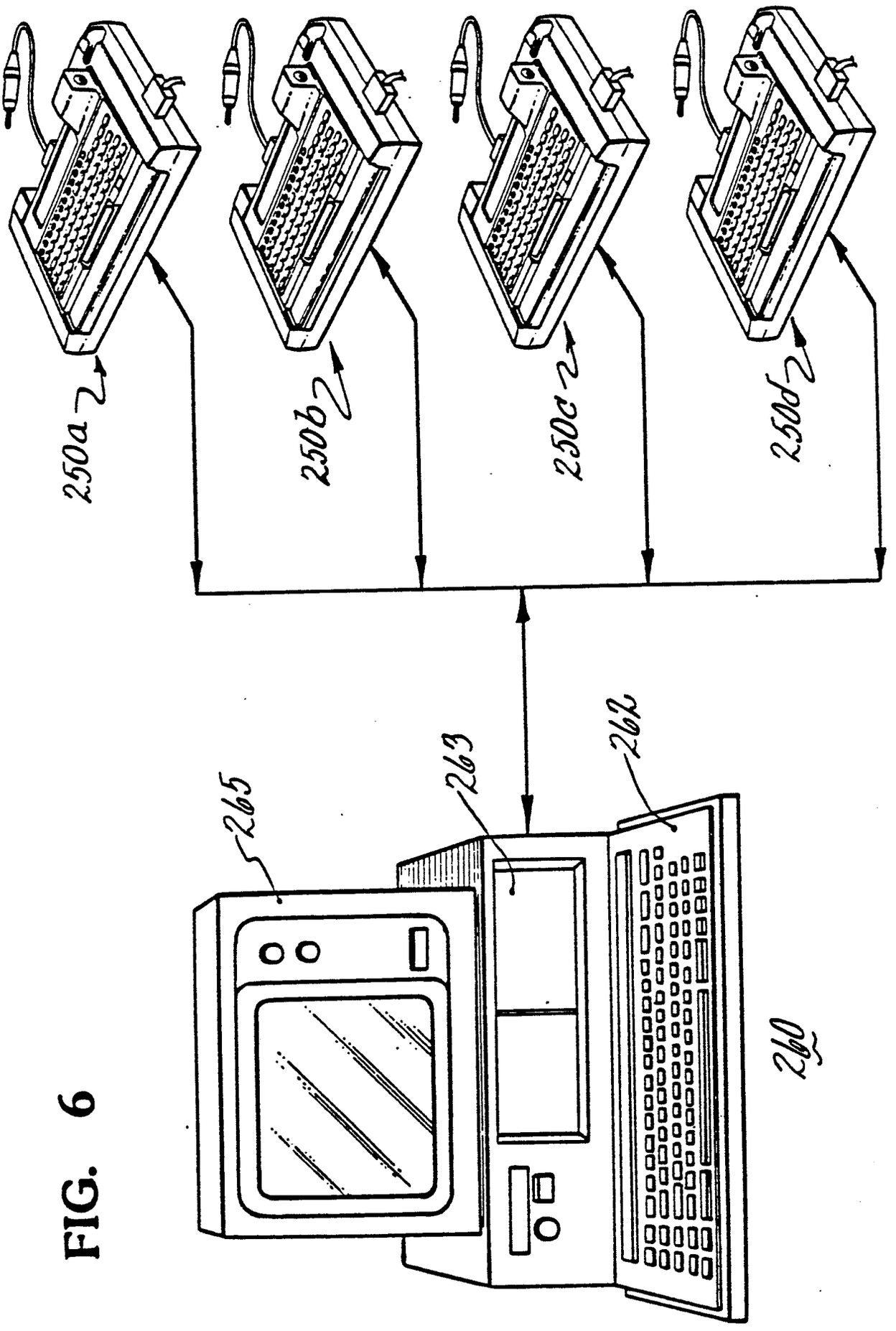


Fig. 8

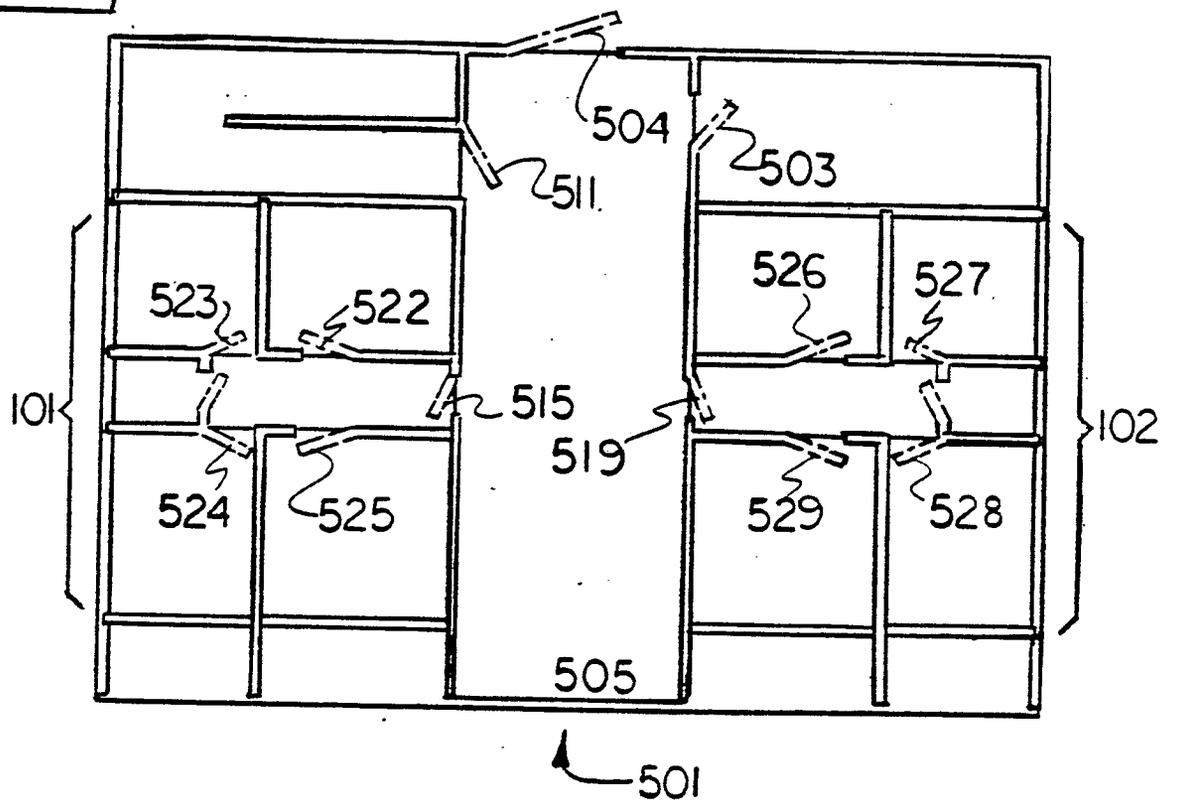
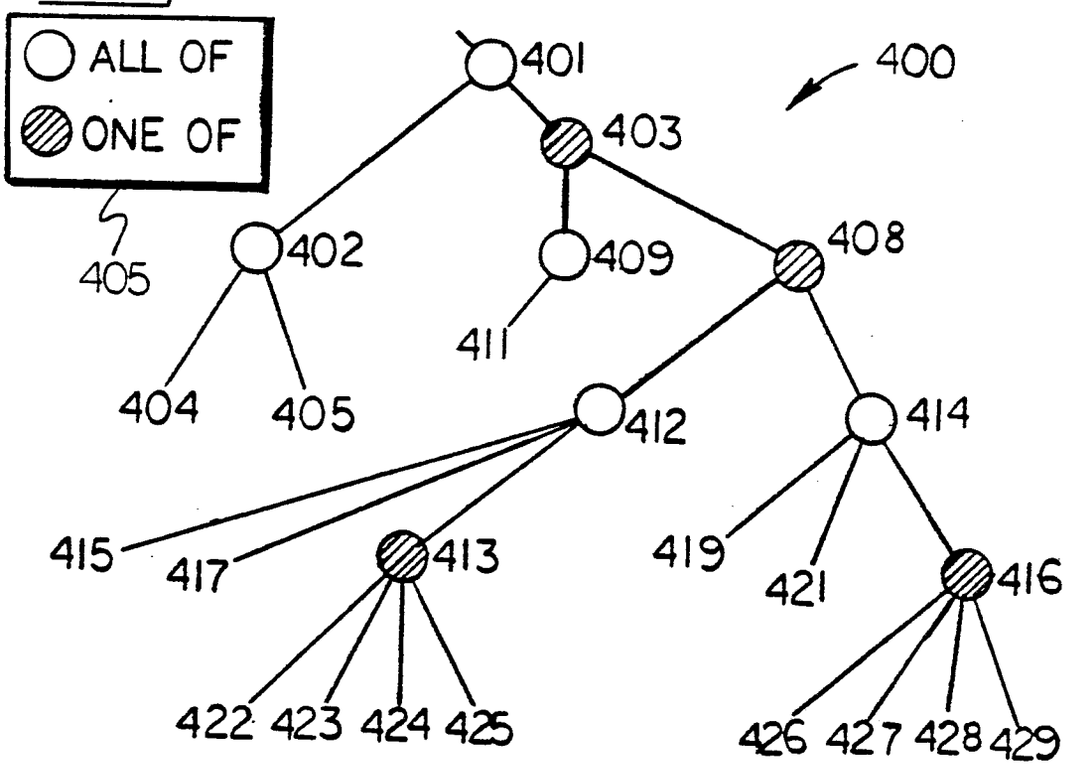

504
503
511
526
527
523
522
101
515
519
102
524
525
529
528
505
501

Fig. 7


ALL OF
ONE OF
405
401
400
403
402
409
408
411
404
405
412
414
415
417
413
419
421
416
422 423 424 425
426 427 428 429