US005199066A

United States Patent [19]

Logan

[11] Patent Number: 5,199,066

[45] Date of Patent: Mar. 30, 1993

[54] **METHOD AND APPARATUS FOR PROTECTING SOFTWARE**

[75] Inventor: **Andrew J. Logan**, Gladwyne, Pa.

[73] Assignee: **Special Effects Software, Inc.**, Philadelphia, Pa.

[21] Appl. No.: **339,760**

[22] Filed: **Apr. 18, 1989**

[51] **Int. Cl.**[5] **H04L 9/00**; H04L 9/32

[52] **U.S. Cl.** **380/4**; 380/23; 380/25; 380/49; 380/50; 340/825.31; 340/825.34

[58] **Field of Search** 364/200, 900; 380/3, 380/4, 23, 25, 49, 50, 22; 360/60; 340/825.31, 825.34

[56] **References Cited**

U.S. PATENT DOCUMENTS

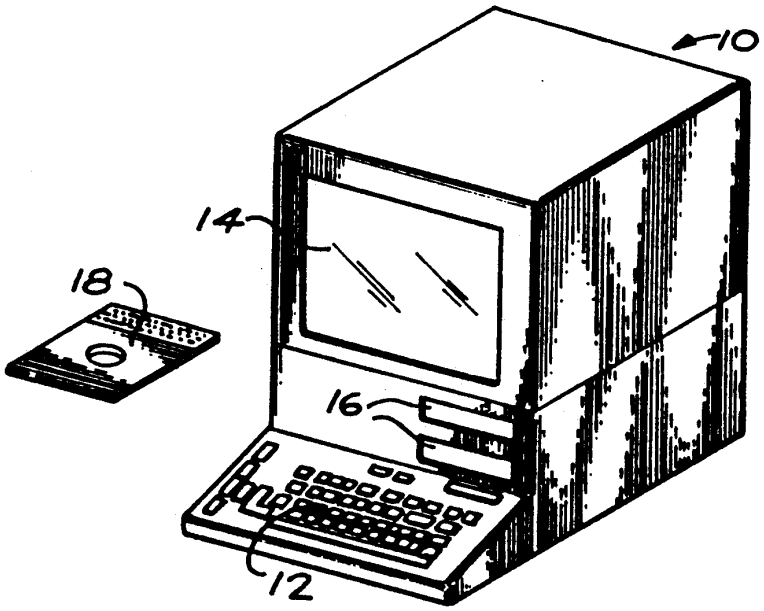
4,433,207	2/1984	Best	380/4
4,458,315	7/1984	Uchenick	380/4
4,471,163	9/1984	Donald et al.	380/4
4,558,176	12/1985	Arnold et al.	380/4
4,562,306	12/1985	Chou et al.	380/4
4,593,353	6/1986	Pickholtz	364/200
4,634,807	1/1987	Chorley et al.	380/4
4,652,990	3/1987	Pailen et al.	364/200
4,658,093	4/1987	Hellman	380/25
4,670,857	6/1987	Rackman	380/4
4,683,553	7/1987	Mollier	380/4
4,683,968	8/1987	Appelbaum et al.	380/4
4,685,055	8/1987	Thomas	364/200
4,740,890	4/1988	William	364/200
4,866,769	9/1989	Karp	380/4
4,901,168	2/1990	Yoshida et al.	360/60

Primary Examiner—Bernarr E. Gregory
Attorney, Agent, or Firm—Woodcock Washburn Kurtz Mackiewicz & Norris

[57] **ABSTRACT**

A method and system for protecting a software program recorded within a storage medium for use with or transmission to computer or processor based hardware comprises inputting a hardware code uniquely associated with the particular hardware and inputting a first software code uniquely associated with the particular embodiment of the software. A first predetermined operation is performed upon the hardware code and the first software code to produce an intermediate code. A unique activation code obtained from the software supplier is inputted and a second predetermined operation is performed upon the intermediate code and the activation code to produce a second intermediate code. The second intermediate code is compared to a second software code uniquely associated with the particular embodiment of the software and stored in a hidden location within the software. The use of the software is enabled only if the second intermediate code and the second software code are identical.

20 Claims, 2 Drawing Sheets



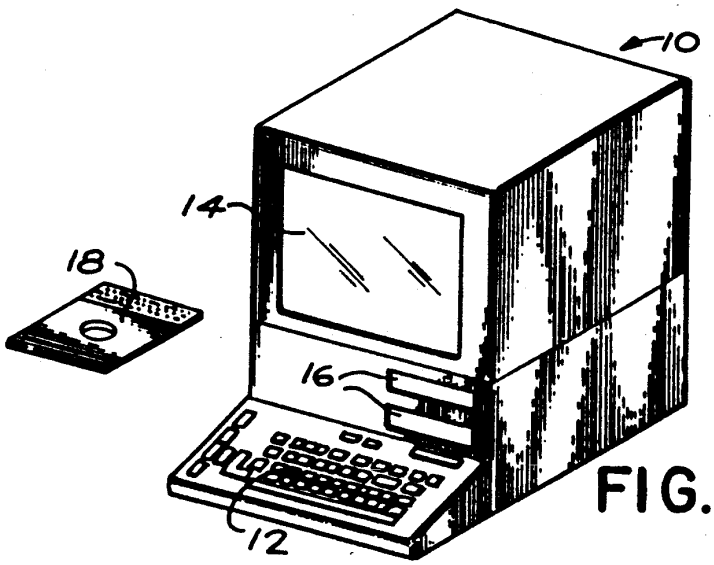


FIG. 1

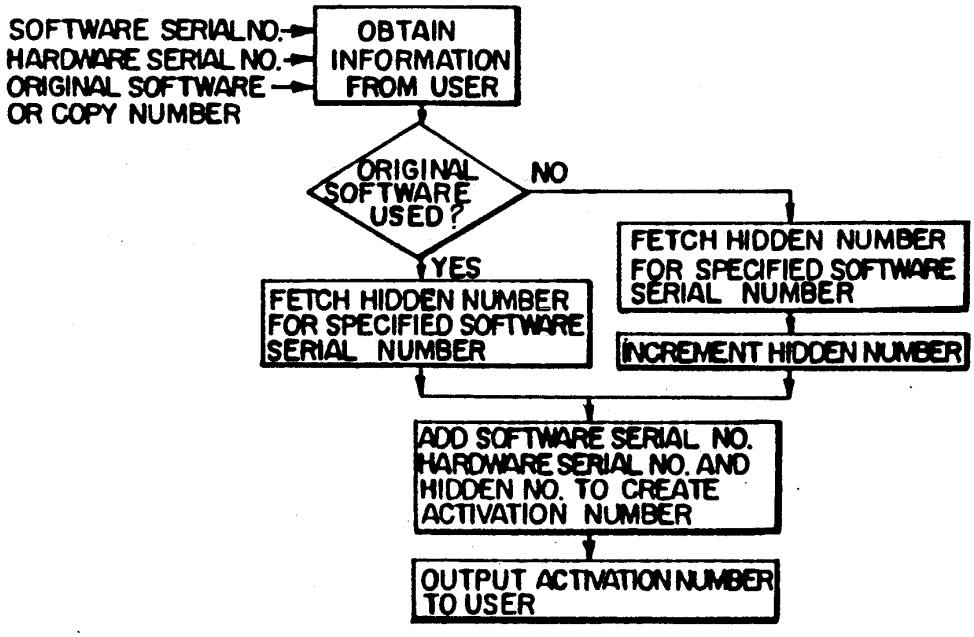


FIG. 2

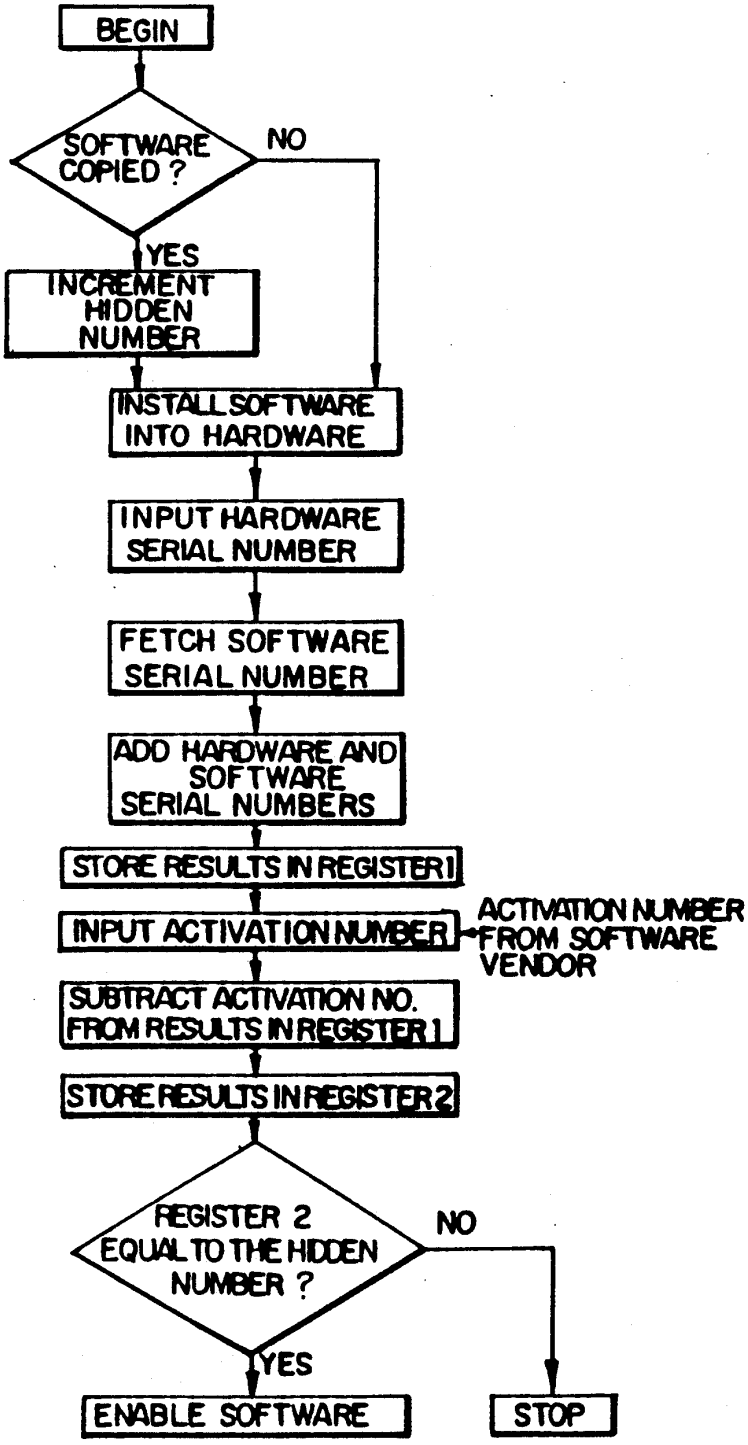


FIG. 3

METHOD AND APPARATUS FOR PROTECTING SOFTWARE

BACKGROUND OF THE INVENTION

The present invention relates to computer software or programs and, more particularly, to a method and system for protecting software programs from unauthorized use and/or copying.

With the tremendous increase in the number of home computers, as well as business computers that are currently in use, there has been a corresponding increase in computer software or programs for use with such home and business computers. For example, specialized applications programs have been developed for everything from presenting elaborate spread sheets and desk top publishing to sophisticated computer games. In general such applications programs are sold to the end user either directly by the individual or company which developed the program, or through an established distribution network which may include mail order and/or retail sales outlets. In many cases, the applications program is stored within a standard magnetic storage medium, such as floppy disk, to facilitate loading of the program into the home or business computer for performing the desired task.

Control of the software, particularly software distributed on a floppy disk, has become a major problem, due to the ease with which a program stored within a floppy disk may be duplicated. Although, in theory, the copyright laws protect software developers from unauthorized copying of such programs, it is impractical, if not impossible, for software developers to fully enforce their copyrights, particularly against companies or individuals making a relatively small number of copies. For example, a small company may purchase one copy of the original software from the developer and may then produce four or five unauthorized copies of the software for separate use on computers at different locations within the company.

Likewise, a group of individuals may combine their money to purchase one original of the software for a particular computer game and then make a number of unauthorized copies for separate use of the software by each of the individuals within the group on their own computers. In either event, the developer of the software is unable to enforce its rights against the copiers since, without having inside information, it is not possible for the software developer to know that the unauthorized copies of the software were made and/or who made the unauthorized copies. Moreover, it would be prohibitively expensive to take legal action to enforce copyrights with respect to such small numbers of unauthorized copies. Accordingly, software developers are generally unable to enforce their rights and, therefore, are suffering economic loss.

Various methods have been developed to prevent the unauthorized copying of software. One such method involves requiring the purchaser of the software to enter into a license agreement which permits use of the software only upon a single designated computer and prohibits the purchaser from making unauthorized copies. This form of protection is difficult and expensive to enforce, particularly when dealing with smaller companies and individuals.

A second form of protection requires utilizing a secret code or password which must be obtained from the software supplier and entered when using the software. While this form of protection has merit, it still does not preclude unauthorized use or copying on a relatively small scale since the code or password can be obtained by one person from the software supplier and can be given to the other users within the company or group.

A third form of protection involves placing restrictions within the computer program which completely preclude copying or permit only a single copy of the program to be made. While this form of protection can be effective, it may prevent a legitimate purchaser of the software from making a single backup copy, as permitted by law. In addition, specialized programs have been developed to circumvent or override this type of protection. Other forms of copy protection have been developed and employed with limited success. In some cases, the other forms of protection are too expensive to employ with some software, and, in other cases, these other forms of protection are not technically suitable for some software.

The present invention overcomes many of the problems inherent in the existing forms of protection for computer software and provides protection from both unauthorized use and copying. With one embodiment of the present invention, the serial number of the particular hardware, as well as the serial number of the particular copy of the software, must be entered, along with a unique activation number obtained from the software supplier, in order to enable use of the software. The software performs an operation upon the hardware serial number, the software serial number and the activation number to produce an intermediate code which is compared to a number hidden within the software and uniquely associated with the particular copy of the software. The program is arranged to automatically change the hidden number in a predetermined manner whenever the software is copied. The software is only operable if the comparison indicates that the intermediate code and the hidden number are identical. In this manner, only a single embodiment of the software can be used with the activation number initially supplied by the software developer. Every time a copy of the software is made, the user must contact the software developer to obtain a new activation number. In this manner, the software developer is able to keep accurate records with respect to the number of copies made and may charge the user accordingly.

SUMMARY OF THE INVENTION

Briefly stated, the present invention comprises a method and system for protecting a software program recorded within a storage medium for use with, or transmission to, computer or processor based hardware. The method comprises inputting a hardware code uniquely associated with the particular hardware with which the software is to be employed and inputting a first software code uniquely associated with the particular embodiment of the software being employed. A first predetermined operation is performed upon the hardware code and the first software code to produce a first intermediate code. A unique activation code for the particular embodiment of the software being employed is inputted, the activation code being received from the software supplier. A second predetermined operation is performed upon the first intermediate code and the activation code to produce a second intermediate code. The second intermediate code is compared with a second software code uniquely associated with the particu-

lar embodiment of the software and stored at a hidden location within the software, the second software code not being ascertainable by the user. The software is enabled for use if the second intermediate code and the second software code are identical.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing summary, as well as the following detailed description of a preferred embodiment, will be better understood when read in conjunction with the appended drawings. For the purpose of illustrating the invention, there is shown in the drawings an embodiment which is presently preferred, it being understood, however, that the invention is not limited to the specific methods and instrumentalities disclosed.

In the drawings:

FIG. **1** is a perspective view of a personal computer and a floppy disk, within which is stored a computer program;

FIG. **2** is a schematic flow diagram which depicts a method for a software supplier to generate an activation code in accordance with a preferred embodiment of the invention; and

FIG. **3** is a schematic flow diagram of a preferred embodiment of a portion of a computer program which verifies that an activation number received from the software supplier is correct and enables use of the remainder of the computer program.

DESCRIPTION OF PREFERRED EMBODIMENT

Referring to the drawings, wherein like numerals indicate like elements throughout, there is shown in FIG. **1** a typical personal computer **10** of a type well known in the art, and commercially available from a variety of manufacturers, for example, IBM Corporation. The personal computer **10** includes a standard keyboard **12**, a standard cathode ray tube (CRT) or screen **14** and a pair of floppy disk drives **16**. The keyboard **12** is employed to facilitate communication between an individual user and the computer **10** in a manner which is generally well known in the computer art. The CRT **14** also functions in a manner well known in the computer art for displaying information inputted through the keyboard **12**, as well as information outputted by the inner workings of the computer **10**. The disk drives **16** are employed in a manner well known in the computer art for receiving one or more floppy disks to facilitate the loading or entry of computer software or programs stored within a floppy disk into the computer **10**. A typical floppy disk **18** is illustrated in FIG. **1**. As used herein, the terms, "program," "computer program," "software" and "software program" are interchangeably used to mean a series of instructions which are used to control the operation of computer hardware or other computer based or processor based hardware.

While in the present description of a preferred embodiment of the invention, a personal computer **10** is shown and described, it will be appreciated by those skilled in the art that the present invention may be employed in conjunction with any other type of computer, including standard computers such as a mini computer or a main frame computer, and/or special purpose computers. In addition, the present invention may be employed in connection with any other type of computer or processor based hardware such as computer or processor controlled machinery or equipment and any device or network of devices using digital signals or switching.

Likewise, while in connection with the description of the presently preferred embodiment, the computer program or software is illustrated as being stored within a floppy disk **18**, it will be appreciated by those skilled in the art that the program or software could alternatively be stored in any other type of storage medium, for example, a different magnetic medium, such as a hard disk magnetic card, magnetic tape, etc.; a semiconductor based storage medium, such as a random access memory (RAM), a read only memory (ROM), a programmable read only memory (PROM), etc.; or a nontraditional storage medium, such as a digital audio or video tape or disk or network of storage devices. Accordingly, it should be clearly understood that the present invention is not limited to the particular computer hardware **10** or storage medium **18** used to illustrate the preferred embodiment of the invention.

FIG. **3** shows the operation of the presently preferred embodiment of the invention. Each original copy or embodiment of the computer software has a first software code which is uniquely associated with that one particular embodiment. In the presently preferred embodiment of the invention, the first software code is comprised of the serial number for that particular copy of the software. When the software is stored within a floppy disk **18**, the serial number is generally imprinted upon a label or other indicia applied to the upper portion of the floppy disk **18** in a manner well known in the art. In the present embodiment, the first software code or serial number is preferably numeric and is comprised of any number of digits.

Associated with each original copy of the software is a second software code which is stored within the software at a hidden location. The second software code is unique for each original copy of the software and may have a predetermined relationship with the first software code or serial number. In any event, the software supplier is able to identify the second software code for each particular embodiment of the software by reference to the first software code or serial number. As previously indicated, the second software code is hidden within the software at a location which is not identifiable or ascertainable by the software user. In the presently preferred embodiment, the second software code is numeric and may sometimes be referred to as the hidden number. However, it will be appreciated by those skilled in the art that the second software code could be some other specialized code, such as alpha, alphanumeric or digitally coded signal.

The computer program automatically changes or increments the second software code in a predetermined manner each time the software is copied. In the presently preferred embodiment, the numeric second software code or hidden number is incremented by the addition of a predetermined number, such as 7, each time the software is copied. The software user is not made aware of the manner in which the hidden number is changed and cannot ascertain this information. It will be appreciated by those skilled in the art that the second software code could alternatively be changed in some other manner such as by multiplying the code by a predetermined number, or performing any other type of mathematical or logical operation upon the second software code.

When a user wishes to use a program protected by the present invention, the software program is installed into the hardware being employed by the user. Of course, it will be appreciated by those skilled in the art

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.