

PROTECTION OF COMPUTER PROGRAMS UNDER JAPANESE COPYRIGHT LAW

Edward G. Durney*

In 1985, the Japanese Diet followed the lead of the U.S. Congress and provided special provisions under Japanese Copyright Law that would protect computer software as a copyrightable work. Some provisions were designed to improve on what the United States had done. In particular, these amendments included express exclusions to copyright protection in order to clarify what in computer programs is "protected expression" and what is "idea." In all, the changes provide clearer and more easily understood boundaries to the scope of protection than are provided under United States law.

As did the U.S. Congress, the Japanese Diet left unanswered some of the major questions concerning the protection of software under copyright law. Unlike American courts, however, Japanese courts would normally not be expected to provide answers to any of these questions; Japan has a civil law system, which emphasizes statutory law over common law. Instead of writing lengthy, authoritative opinions on what the law is, the Japanese courts tend to decide cases in terse opinions that only apply the law to facts, and that do little to develop the law.¹ Moreover, the number of court cases brought in Japan is quite low, making legal principles develop through the judicial process only slowly and ponderously, if indeed at all.

Fortunately, however, there have been two important decisions interpreting Japanese copyright law as that law relates to computer software. *Microsoft Corp. v. Shuuwa System Trading K.K.*² and *Sys-*

* Morrison & Foerster, Tokyo, Japan. Part of the research for this article was carried out at Tokyo University under a Fulbright Graduate Research Fellowship.

1. While largely a positive feature of the Japanese legal system and consistent with the judicial philosophy followed in many civil law nations, the practice of writing terse opinions which decide cases without making law can complicate efforts to predict court responses to novel and evolving issues, such as the protection of computer software under copyright law.

2. Judgment of Jan. 30, 1987, Chisai (District Court), Tokyo, 1219 Hanji 48.

*tem Science K.K. v. Toyo Sokuki K.K.*³ (each discussed below) give valuable clues on how Japanese courts will decide cases involving copyright law protection of computer software. Coupled with concrete software-related amendments to the Japanese Copyright Law, which give more guidance about the scope of protection for computer software than does the United States Copyright Law, these two important decisions make Japanese law relating to the protection of computer software under copyright law more coherent, stable, and well-reasoned than the rather chaotic American law.

I. STATUTORY LAW

A brief explanation of the Japanese software-related copyright law amendments will provide a useful background to the discussion of the *Microsoft* and *System Science* decisions. The software-related amendments to the Japanese Copyright Law largely address form rather than substance. They simply make software copyrightable material, and leave general copyright principles to apply unmodified to software. However, there are a few areas where the drafters of the amendments made special provision for software, such as the scope of protection for computer programs. These areas should be carefully noted.

A. Definitions

The amendments to the Copyright Law provide a definition of "program," which "means an expression of combined instructions given to a computer so as to make it function and obtain a certain result."⁴ A "work" is defined as "a production in which thoughts or sentiments are expressed in a creative way and which falls within the literary, scientific, artistic or musical domain."⁵

The Japanese courts had consistently found, even before the amendments were enacted, that computer software is a "work" protected under copyright law.⁶ However, the amendments make this explicit in Article 10(1)(ix), which adds "program works" to the list of particular examples of copyrightable "works." It should be noted that program works are no longer considered "literary works" under Japanese copyright law, since those works fall within the separate classification of Article 10(1)(i): "novels, dramas, articles, lectures and other literary works."

3. Judgment of June 20, 1989, Kōsai (High Court), Tokyo, 1322 Hanji 138.

4. Chosakuken Hō (Copyright Law of Japan) art. 2(1)(*xbis*). The translations of the provisions of the Japanese Copyright Law used in this article are taken from Agency of Cultural Affairs, Chosakuken Hō. The Agency of Cultural Affairs is the Japanese government agency which executes the copyright law.

5. *Id.* art. 2(1)(i).

6. *See, e.g.*, Judgment of Dec. 8, 1982, Chisai (District Court), Tokyo, 1060 Hanji 18.

B. Interfaces and Algorithms Not Protected

A significant provision relating to computer software is Article 10(3), which specifically excludes programming languages, rules, and algorithms from copyright coverage:

The protection granted by this Law to [program works] shall not extend to any programming language, rule or algorithm used for making such works. In this case, the following terms shall have the meaning hereby assigned to them respectively:

- (i) "programming language" means letters and other symbols as well as their systems for use as means of expressing a program;
- (ii) "rule" means a special rule on how to use in a particular program a programming language mentioned in the preceding item;
- (iii) "algorithm" means methods of combining, in a program, instructions given to a computer.⁷

This express exclusion may not be absolutely necessary. Although Japanese Copyright Law does not include an express provision excluding "ideas" from copyright protection, Japanese courts and commentators have created an idea/expression dichotomy similar to that in the U.S. Copyright Act.⁸ In particular, they have interpreted the Article 2(1)(i) definition of "works," particularly its reference to "expression," to mean that only expression is protected, not idea.⁹ Therefore, programming languages, rules (such as interfaces and protocols), and particularly algorithms, would seem to be excluded from protection even without the express exclusion of Article 10(3).

This express exclusion is nonetheless helpful. Almost all copyright laws, including those of both the U.S. and Japan, provide that the author of a copyrighted work has the exclusive right to make adaptations based on that work. In Japan, Article 27 of the Copyright Law provides that "the author shall have the exclusive right to translate, arrange musically, transform, dramatize, cinematize, or otherwise adapt his work." Particularly in the case of computer software, it is difficult to define the scope of this broadly worded right simply by using an unwritten and largely amorphous idea/expression distinction.

Given the difficulty of trying to find the line between non-infringing works which merely use the same idea and infringing adap-

7. Chosakuken Hō (Copyright Law of Japan) art. 10(3).

8. See, e.g., Karjala & Sugiyama, *Fundamental Concepts in Japanese and American Copyright Law*, 36 AM. J. COMP. L. 613, 649-50 (1988).

9. See, e.g., COPYRIGHT LAW 34 (1985) (S. Azuma, ed. 1985) ("[A] copyrighted work must be the creative 'expression' of thoughts or sentiments. That which is protected in a copyrighted work of authorship takes the form of concrete, external expressions using means such as words, letters, sound and color. The ideas or theories expressed, the thoughts and feelings themselves, are not protected.") [in Japanese].

tations, the express exclusion of algorithms and rules from copyright protection is particularly helpful. For example, a program which uses the same basic algorithm as another program conceivably could be considered an infringing adaptation under the language of Article 27. The express exclusion of Article 10(3)(iii), however, unequivocally states that such a similarity does not constitute a copyright infringement.

The exclusions also apply to interfaces and protocols. Some commentators have argued that copyright law extends some protection,¹⁰ even though interfaces and protocols are generally considered to be ideas or principles which are not subject to copyright protection. However, the express exclusion of "rules" in Article 10(3)(ii) makes it clear that interfaces and protocols themselves are not protected by the copyright law of Japan.¹¹

The mistaken belief that interfaces or protocols can be protected comes from not carefully distinguishing interfaces and protocols, which are unprotected ideas, from a particular implementation of an interface or protocol, which can be protected expression. The interface or protocol itself is usually represented using specifica-

10. Two Japanese lawyers stated their view that most interfaces and protocols are protected under Japanese copyright law. Hirakawa & Nakano, *Copyright Protection of Computer 'Interfaces' in Japan*, 12 EUR. INTELL. PROP. REV. 46 (1990). Their main topic of discussion, of which they provide convincing evidence, is that the express exclusions from protection set forth in Article 10(3) were not intended to change existing law or narrow the protection given computer programs, but instead were merely intended to delineate the existing idea/expression dichotomy as it applies to computer programs.

However, their main thesis—that most interfaces and protocols are protected under Japanese law—is unsupported. They go so far as to state that data formats for transferring data between programs and among processors in networks (which they call "program interfaces") and communications protocols for long distance data transmission (which they call "communications interfaces") are not excluded from copyright protection under Article 10(3)(ii). *See id.* at 55-57. They appear to be alone in this opinion. *See* authorities cited *infra* note 11. It is clear that interfaces and protocols such as they describe are expressly excluded from copyright protection under Japanese law.

11. The definition of "rules" in Article 10(3)(ii) is not very artfully drafted, and one article exploits this ambiguity to argue that the express exclusion of Article 10(3)(ii) covering "rules" does not exclude interfaces from copyright protection. Hirakawa & Nakano, *supra* note 10, at 46. However, it is clear from numerous authorities that the exception was designed specifically to cover interfaces and protocols. *See, e.g.,* Ozaki, *Copyright Protection of Software: The Japanese View*, 1990 COMPUTER L. REP. 950, 959 ("the legislative history [of Article 10(3)(ii)] reveals that interface specifications, including communications protocols, are intended to be included under the definition of 'rules'"); Bando, *Partial Amendment of Copyright Law Concerning Computer Programs*, 292 COPYRIGHT 2, 7 (1985) [in Japanese]; Ohashi, *A Legal Analysis of Interfaces*, 865 JURISUTO 92, 94 (1986) [in Japanese]; M. MATSUDA, *INTELLECTUAL PROPERTY RIGHTS IN THE COMPUTER AGE* 16 (1988) [in Japanese]; N. NAKAYAMA, *LEGAL PROTECTION OF SOFTWARE* 42-47 (new ed. 1988) [in Japanese]; Karjala, *The Protection of Operating Software Under Japanese Copyright Law*, 10 EUR. INTELL. PROP. REV. 359 (1988).

tions, while the implementation of the interface or protocol is usually a computer program.

The distinction between an interface or protocol and its implementation might best be explained by analogy to the rules of a game such as poker. Anyone is free to describe the rule of poker that five cards are passed to each player, for example, even though there are certainly other possibilities, like four or six cards. That rule is an unprotectable idea. However, that does not mean that one can freely use copy descriptions of the rules of poker from another source, such as *According to Hoyle*. The expression used in *Hoyle* to describe the rule is protected.

Of course, any two expressions of the same idea will quite often be very similar, so that identity, or near identity, of expression should be required in such cases before infringement is found. Moreover, when an idea and its expression merge because they are so similar, the merger doctrine provides that even identity of expression is not infringing.

The interfaces or protocols must be implemented in a computer program to be of any use, and they can often be derived from the program just as a book can be read to understand the ideas expressed in a book. However, the process of deriving interface information is usually not as simple as reading, and this information must sometimes be derived using reverse engineering techniques.

As discussed below,¹² using reverse engineering is not a copyright infringement unless expression (and not just idea) from the program being studied is copied into one's own program or documentation. However, as is shown by the *Microsoft* decision,¹³ if reverse engineering is used to copy expression in creating an infringing work, the work is no less infringing simply because the copying of expression was done through reverse engineering rather than directly.

C. Authorship By Employee

The amendments also modify some provisions of the Copyright Law to deal with the peculiar characteristics of software. One instance of this is Article 15, which applies to authorship of a work made by an employee in the course of his duties. A work must normally be made public under the name of the employer in order for the employer to claim authorship.¹⁴ In the case of computer software, however, it was recognized that computer software is often kept secret and is never made public, and the requirement was

12. See discussion *infra* Section II.

13. See discussion *infra* Section IV.

14. Chosakuken Hō (Copyright Law of Japan) art. 15(1).

dropped.¹⁵

D. Modifications

The drafters of the amendments were concerned that normal, legitimate uses and modifications of a program work might be considered an infringement of the moral rights of the program's author or the right of reproduction or adaptation of the copyright owner. Therefore, some specific exceptions were added to the Copyright Law with regard to program works.

The author of a copyrighted work normally has the moral right to preserve the integrity of his work.¹⁶ However, a broad exception is made to this right in the case of program works. This exception allows any modifications necessary to enable a program to be used on a computer on which it would otherwise be unusable, or to make more effective use of a program work on a computer.¹⁷

This last phrase makes the exception eat up the rule, since it is hard to imagine any modification that could not be justified as having been made to make more effective use of the program. However, that was probably the intent of the drafters of the amendments, since there is little need for the moral rights in the case of program works.

Additionally, the owner of a program work copy may make copies or adaptations of that work as necessary for personal use on a computer.¹⁸ This right was included in the law as one of the express "fair use" exceptions to the copyright owner's rights. It was intended to authorize the making of back-up copies and the copying of the program from the memory medium to the computer's internal memory. However, the exception is narrow, and any distribution of such a copy or adaptation is a copyright infringement.¹⁹

E. Use as Infringement

The amendments also provide that knowing use of an unauthorized copy of a copyrighted program work is itself an infringement.²⁰ Mere use of a copyrighted work is not normally a copyright infringement. Thus, this provision is an example of an attempt to adapt copyright law to make it more meaningful in view of the unique characteristics of program works.

15. *Id.* art. 15(2).

16. *Id.* art. 20(1).

17. *Id.* art. 20(2)(iii).

18. *Id.* art. 47*bis*(1).

19. *Id.* art. 49(1)(iii), 2(ii).

20. *Id.* art. 113(2).

F. Registration

The registration system also was changed to accommodate computer software. The role of registration in the Japanese copyright system is very minor; registration is not a requirement for copyright to arise or to be enforced. Nevertheless, as with other copyrighted works, registration can be made of the true name of the author and of the date of the first making public of the work. In the case of computer software, the fact that computer software is often not made public gave rise to a new section, Article 76*bis*, which allows registration of the date of creation of a program work.

These registrations are sometimes helpful to forestall problems of proving authorship, date of publication, or date of creation. More importantly, transfers of copyright or security interests in a copyright are not effective against third parties unless the transfer or security interest is registered.

Registration of program works is carried out by a private organization, the Software Information Center (SOFTIC), under the provisions of a special law relating to the registration of program works. Although registration does give rise to some advantages, it has not proven to be particularly popular with computer software authors.

Registration of transfers of copyrights so that copyrights are effective against third persons is particularly important under the strict work-made-for-hire rules in Japan. Under the Japanese Copyright Law, a work is a work-made-for-hire only if the person who created the work is an actual employee of the person claiming authorship, and if the creator created the work in the normal course of his or her duties.²¹

In many cases, particularly computer software created by a subcontractor, the person who created the work will be considered the author, even if there is an agreement with the creator that the ownership of the copyright will belong to the person who contracted for the program to be created. In such cases, however, a purported transfer of the copyright in the work is not effective against third parties unless registered. Because relatively few transfers of copyright in computer software have been registered with SOFTIC,²² there are probably many software distributors who think they have full rights to their software who actually do not have any rights against a third party, innocent or knowing, who later obtains a transfer of the copyright from the creator and registers it before the software distributor does so.

21. *Id.* art. 15.

22. About 100 transfers have been registered in the several years that SOFTIC has been accepting registrations.

Given the expense of creating software, one would think that the trouble and expense of registering the transfer of the copyright in the software would be well worthwhile. Such thoroughness, however, does not appear to be the norm.

II. THE MICROSOFT CASE

The review above of statutory protection of computer software under copyright law in Japan lays the basis for understanding the two recent court decisions regarding software protection issues. The case of *Microsoft Corp. v. Shuuwa System Trading K.K.*²³ answers two important questions regarding the legal protection of computer software. First, the *Microsoft* court held that outputting an object program from the Read-Only-Memory ("ROM") of a personal computer, disassembling and analyzing the object program to create a source code listing, attaching labels and comments, and then publishing it was an infringement of the copyright in that object program. Second, the court held that operating system programs are protected under Japanese copyright law in the same manner as are applications programs.

Decided by the Twenty-Ninth Civil Division of the Tokyo District Court, which specializes in intellectual property disputes, the case results indicate that Japanese courts will extend protection to computer software under basically the same conditions as in other countries, and should dispel any fears that Japan would not give adequate protection to computer software. At the same time, however, the decision shows that the Japanese courts will carefully consider the proper scope of protection that should be given.

A. Facts

In *Microsoft*, the Tokyo District Court held that the two defendant Japanese corporations, Shuuwa System Trading K.K. and Tokyo Sugaki Printing K.K., had violated the copyright of the American plaintiff, Microsoft Corporation. The copyright at issue related to an operating system program, an interpreter for the BASIC programming language, designed for use on the Nihon Electric Company ("NEC") PC-8001, a popular personal computer. Microsoft produced the program at the request of NEC, but retained the copyright.

The allegedly infringing work in this case was a book which contained a source program listing in the NEC PC-8001 assembly language with explanatory labels and comments. The book was cre-

23. Judgment of Jan. 30, 1987, Chisai (District Court), Tokyo, 1219 Hanji 48. See Appendix 1 of this article for an English translation of the *Microsoft* decision.

ated by the defendant Shuuwa System Trading K.K., and printed by the defendant Tokyo Sugaki Printing K.K.

The book was created by outputting the object code of the BASIC interpreter, which was stored in machine-language form in the memory of the PC-8001.²⁴ This code was then disassembled, which means it was automatically translated from machine language, which is not easily understood by humans, to assembly language, which is more easily read by humans. Finally, a software expert studied the disassembled code and derived the meaning of each of the instructions of the program, adding labels and comments to explain how the program worked. Users of the PC-8001 could employ the book to understand better the plaintiff's program.

The BASIC interpreter is one of the important operating system programs of the NEC PC-8001, since it is this interpreter which allows the user of the computer to use the BASIC programming language to input his instructions and commands to operate the computer. If the users of the computer were forced to write their programs in machine language, rather than a high-level language like BASIC, the computer would be very difficult to use.

Operating system programs are usually included as a package with the computer hardware, and consequently are rather invisible to users. These programs include "operating systems" (the main or core system management program is usually called the operating system, although the term is also used more generally to refer to all operating system programs as well), programming language compilers and interpreters, and other programs which perform functions related to the internal operation and management of computer resources.

Generally, operating system programs are distinguished from application programs. The latter are more familiar to most computer users, and include accounting programs, word processing programs, games, and other programs which perform a particular task desired by the user, a task unrelated to the internal operation and management of the computer.

It is difficult to classify some programs as operating system programs or application programs, since almost all programs have at least some attributes of both. Moreover, operating system programs differ from application programs only in that they perform different tasks. The superficial characteristics of both are the same,

24. Although the court described this as converting the machine-language code to hexadecimal code, it is doubtful that such a conversion took place as an intermediate step. It is more likely that the defendant read the machine-language code from the ROM and used it as input to a disassembler, which produced assembly-language code. Thus, the conversion would have been directly from machine language to assembly language.

since both consist of the same type of instructions or commands to be performed by the central processing unit of the computer.

However, there was some question whether operating system programs would be treated the same as application programs under the Japanese Copyright Law. Application programs were held to be protected in a line of cases beginning with *K.K. Taito v. K.K. I.N.G. Enterprises*,²⁵ a case involving exact or "dead" copying of the Space Invaders video game. However, the question of whether protection should extend to operating system programs as well remained unaddressed, although most scholars thought that the eventual answer would be in the affirmative, as it was in the United States.²⁶ The facts of this case gave the court a chance to address this issue of first impression in Japan.

Finally, it is important to note that this case is based on acts which took place before the 1985 amendments to the Copyright Law²⁷ became effective (on January 1, 1986). As discussed above, the 1985 amendments added some provisions explicitly protecting computer programs under the Copyright Law.

B. Judgment

The court addressed two major questions in its reasoning. The first question was whether, after disassembling and interpreting the object code and attaching comments and labels, the publishing of the results was infringing activity. The second question was whether the BASIC language interpreter, an operating system program, was protected under the Japanese Copyright Law. The court answered both questions affirmatively.

1. Infringing Activity

In deciding whether the defendants had infringed the plaintiff's copyright, the court first found that "it is clear" that outputting the object code from the ROM of the NEC PC-8001 into hexadecimal code was reproducing the object program.²⁸ The court then discussed whether disassembling and interpreting the object program and attaching labels and comments was also making a reproduction. To determine whether this was making a reproduction, the court compared the disassembled source code that had been proffered as evidence by the plaintiff with the defendants' book.

The court noted that there were external differences between the two, but found these differences to be unimportant, since they

25. Judgment of Dec. 6, 1982, Chisai (District Court), Tokyo, 1060 Hanji 18.

26. *Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240 (3d Cir. 1983), *rev'g* 545 F. Supp. 812 (E.D. Pa. 1982).

27. Law No. 62, June 14, 1985.

28. Judgment of Jan. 30, 1987, Chisai (District Court), Tokyo, 1219 Hanji 48, 55.

arose only because the book had more explanatory labels than the plaintiff's source code. The court also found that the meaning of the corresponding parts of the two works were the same. Thus, the court held that the disassembling of the hexadecimal code, the subsequent attaching of labels and comments, and the publication of the results was infringing activity.

The defendant made two arguments in defense of its actions, one of added creativity and the other of fair use:

The defendants claim that the defendant Shuuwa's actions in interpreting the disassembled listing, attaching labels and comments, and giving descriptions of the items in the source list column of the book at issue should be viewed as the presenting of the fruits of its research and therefore as independent creative activity, and that these actions were thus not copyright infringement.²⁹

The court flatly rejected this argument. "[W]hether or not something newly created based on a preexisting work possesses creativity or individuality should have no bearing on whether or not the copyright in the preexisting work has been infringed."³⁰ Since it found that the question of whether the defendants' work possessed creativity or independence was unimportant, it made no findings regarding the issue.

The court also dismissed the argument that the purpose the defendants had for creating their work made their actions legitimate. The defendants "argue[d] that because the defendants' presenting of the book at issue was for the benefit of the users of the personal computer at issue, the defendants' actions were fair and just."³¹ The court held, however, that "it seems natural that the act of presenting the work at issue, which was not made public by its author as a source program, against the wishes of the author cannot be justified simply because it was done for the convenience of users, and the defendants' argument is therefore inapplicable and cannot be adopted."³²

2. Copyrightability of Operating System Programs

The court made two key findings regarding the issue of whether operating system programs are copyrightable. First, it considered whether the plaintiff's work was the expression of thoughts or sentiments. Second, it considered whether the plaintiff's work was within the scientific domain. Both questions were answered affirmatively.

29. *Id.*

30. *Id.*

31. *Id.*

32. *Id.*

a. Expression of Thoughts and Sentiments

In Ground for Claim 5(iii), the court reproduced the plaintiff's step-by-step analysis of the COINIT subroutine and offered its own analysis, equally detailed, of another portion of the work. These analyses were presented, apparently,³³ to demonstrate the fact that the creator of a program like the plaintiff's work "uses his originality and ingenuity with regard to the overall structure of the program, the structure of each routine, and the combining of the commands and other information at each address, in order to match the hardware architecture of the personal computer at issue and to fully utilize its capabilities."³⁴

In making this finding, the court rejected the defendants' argument that

[a]ccording to the copyright law, the reason for protecting works is to protect personal profit.

However, the work at issue here is an operating system program (basic software), and operating system programs have the purpose of managing data more efficiently and quickly and can therefore be regarded as scientific theories.³⁵ Thoughts and sentiments are completely excluded; pure logic is sought after. In this regard, application programs and game programs, which are expressions of thoughts and sentiments in line with the creator's objective, are completely different.

Because it naturally follows that the work at issue is something to which the laws of science and technology apply and in which the element of personal rights can be ignored, it is not protected under copyright law.³⁶

The court thus rejected the defendants' argument that the plaintiff's work, as an operating system program, was too scientific and theoretical to be the expression of the author's thoughts and sentiments. Moreover, the court decided that the plaintiff's work was the expression of the author's thoughts and sentiments since it reflects the author's own unique method of achieving the purpose of the software:

33. The court spent an inordinate amount of space analyzing and comparing the two programs. Since the court never explained its comparison or what the important points of comparison were, such a detailed comparison added little to the opinion and could probably have been dispensed with.

34. Judgment of Jan. 30, 1987, Chisai (District Court), Tokyo, 1219 Hanji 48, 53.

35. The defendants' use of the word "scientific" here may seem to be a tacit admission that the work at issue is in the scientific domain. However, the word "scientific" used here is a different Japanese word (*kagaku*) than is used in the list "literary, scientific, artistic, or musical domain" of Article 2(1)(i) of the Copyright Law. The Japanese word "*gakujutsu*" is translated in the latter as "scientific," although a better translation in this case might be "scholarly" or "academic." The defendant probably intended to contrast the purely scientific nature of the work at issue with the scholarly nature of normal copyrightable works.

36. Judgment of Jan. 30, 1987, Chisai (District Court), Tokyo, 1219 Hanji 48, 52.

Generally with regard to computer programs there is not just one solution to achieve a particular purpose, but it is possible to choose from a variety of solutions. Similarly, when the objective is to produce a BASIC interpreter to put into a personal computer, as in this case, the various problems in achieving that purpose must be individually and carefully analyzed and a solution found for each, as was previously examined in detail, and the program is completed by writing these solutions in assembly language as a combination of commands and other information. All of these processes are unique, and not only does each process differ in that it reflects the creator's personality and thoughts, but there is also value in that individuality. There is not the least bit of difference regarding this point whether the program is a game program, an application program, or as in this case an operating system program.³⁷

Accordingly, the court asserted that "there is no validity to the defendants' claim that the work at issue is not protected under copyright law" because it differs in some way from application software.³⁸ Thus, the court concluded that operating system programs are copyrightable subject matter in the same way as applications programs.

b. Scientific Domain

The court also found that the plaintiff's work is within the scientific domain:

According to the facts found above, the work at issue was created, even down to the program structure, routines, and use and combining of the subroutines, using a high degree of technical knowledge concerning programming languages so that commands and programs input into the personal computer at issue through the BASIC language are executed as written, and the result desired by the person inputting the command is produced, and it is clear that the work is the expression of the scientific thoughts of the program creator and therefore can be regarded as a work within the scientific domain.³⁹

Thus, the court found that an operating system program is an expression of "thoughts or sentiments" and falls "within the scientific domain," and is therefore a "work" protected under the Copyright Law.⁴⁰

The court did not discuss the question of whether operating system programs are an expression of creativity, which is also some-

37. *Id.* at 55.

38. *Id.*

39. *Id.* The Japanese word translated in this sentence as "scientific" is "gakujutsu," which is probably better translated as "scholarly" or "academic." See *supra* note 35. However, since the word was translated as "scientific" in the quasi-official English translation of the Copyright Law published by the Cultural Affairs Agency, that pattern has also been followed here.

40. *Id.*

times characterized as one of the requirements of the Copyright Act's definition of a "work." Either the court did not consider "creativity" to be a requirement or the parties did not contest the point.

C. Comment

The more important question presented by the facts was whether the defendants had infringed the plaintiff's copyright by disassembling the object code, attaching comments and labels, and publishing the results. The court's analysis regarding this question was not as rigorous as it should have been, but its affirmative answer seems correct. The other question presented by the facts was whether the BASIC language interpreter, an operating system program, was protected under the Copyright Law. This question was more carefully addressed by the court, and its answer well reasoned.

1. Infringing Activity

The court held that outputting the object code in hexadecimal code from ROM was reproducing the object code, so that, after disassembling and interpreting the hexadecimal code and attaching labels and comments, the publishing of these results was an infringing act. The basis for this holding was the court's finding that the defendants' book had the same meaning as the plaintiff's assembly language program.

It should be noted that the court did *not* hold that reverse assembly alone is an infringing act.⁴¹ It is true that the defendants did reverse assemble object code into assembly language, and that the court did find the defendants to have infringed the plaintiff's copyright. Considering together these two facts, some have cited the court as holding that the act of reverse assembling object code is an act of copyright infringement. Such a citation is not correct.

The opinion focused only on the work produced by the defendants, not the process they used to create it. The court addressed

41.

"In [the *Microsoft* case], I argued, as the plaintiff's attorney, that the act of reverse assembly is an act of unauthorized copying when you consider together all the acts defendant Shuuwa carried out of reverse assembling the plaintiff's program, creating a source code listing, and then publishing it as a book, and the court accepted that argument. Therefore, it is clear that the court did not categorically hold that all acts of reverse assembly are acts of unauthorized copying, and I should add that reverse engineering was not discussed in that case."

Miki, *Currents in Intellectual Property Rights (Part 1)*, 928 JURISUTO 79 (1989) [in Japanese].

As discussed *infra* Section IV, reverse engineering is permissible under Japanese law when it is not used to create a work substantially similar in expression to the work being reverse engineered.

only the issue of whether the defendants' book has the same meaning as the plaintiff's assembly language program, and decided that issue by making a detailed comparison of the defendants' work with the plaintiff's work. By contrast, the opinion did not address at all the question of whether the defendants' use of reverse assembly to create its work was permissible under Japanese copyright law. Accordingly, citing the *Microsoft* opinion for the holding that reverse assembly of object code is an act of copyright infringement under Japanese copyright law is reading something into the opinion that is not there. The court simply did not decide that issue one way or the other.

The main problem with the reasoning of the court lay in the ruling that a book produced by reproducing the object code, then translating the object code (machine language) into source code (assembly language), then changing addresses from numbers which give no additional meaning to the reader to explanatory labels which do, and then adding comments, was making a simple reproduction. Even if one can accept that argument, it is not self-evident as the court suggests. At some point along the line, although perhaps beyond the point the defendants went in this case, a work ceases to be a reproduction and becomes an adaptation, or even an independent work.

a. Reproduction Right Analysis

In conducting its analysis, the court did find that there were differences in expression between the two works at issue, stating that "[t]he work at issue (the B-column of each page of Exhibit A-1) and the *ro*-column of each page of the book at issue have external differences."⁴² The court gave the following example:

[I]f we look at address 0004, JP L003B is written in the work at issue, while JP WAMCHK;CHECK COLD START OR WARM START is written in the *ro*-column of the book at issue, and they differ in that regard. However, while the former means an unconditional jump to location 3B (the WARM CHECK subroutine begins at location 003B), and the latter means an unconditional jump to the address labeled WAMCHK, the latter is just an explanation in English of the instruction's meaning and an English abbreviation label indicating the function performed at the place the jump is made to, and the former and the latter both have the same meaning.⁴³

The court concluded that "[t]he other differences in expression between the two texts largely arise from the difference in explanatory labels."⁴⁴

42. Judgment of Jan. 30, 1987, Chisai (District Court), Tokyo, 1219 Hanji 48, 55.

43. *Id.*

44. *Id.*

Based on the findings that the plaintiff's work and the defendants' book have the "same meaning," the court concluded that "both versions are the same program."⁴⁵ Based on this conclusion, the court then held that "the acts of the defendant Shuuwa in disassembling and then interpreting the object program at issue and attaching labels and comments to the interpreted version can be considered to be the act of reproducing the work at issue."⁴⁶

Such reasoning, however, is too conclusory to be logically sound. Merely because two works have the same meaning does not mean that one is a copy of the other. It is black-letter law that only substantial similarity in expression is infringing, and similarity of idea only, substantial or not, is not. Two works which share the same meaning merely share the same idea. They do not necessarily share the same expression.

Indeed, as the court itself pointed out, there were differences in expression between the two works, yet the court did not analyze the expression of the two works to determine whether they were substantially similar. Instead, it merely found that the differences were not as important as the similarities without ever finding that the similarity was in expression rather than idea.

Part of the weakness of the court's ruling stems from the fact that it compared the allegedly infringing book to a translation of the plaintiff's work, not the plaintiff's work itself. This was necessary because the plaintiff's work was object code (machine language) embodied in ROM and thus not readily perceived by humans, and therefore needed to be translated and interpreted to a source code version (assembly language) that can be dealt with more easily.

However, the court ignored the fact that it was not actually comparing the plaintiff's work with the defendants' work. The court's logic assumes that because the translation of the plaintiff's work was a reproduction of the work, that any work with the same meaning as the translation is also a reproduction of the work. This logic is incorrect. Because the court necessarily had to compare translations, since the original work could not be perceived by humans, it should have decided whether there had been an infringement of the adaptation right, not the reproduction right.

b. Adaptation Right Analysis

The court should have analyzed the defendants' work as an adaptation, since it was, if an infringing work at all, an adaptation of the plaintiff's work. Because it was in assembly language rather than machine language, because it had meaningful English abbreviations of the function of the memory locations as the labels for

45. *Id.*

46. *Id.*

memory locations rather than mere numbers, and because it had explanatory comments, the defendants' work should not have been analyzed merely by examining whether the "meaning of the two works are the same."⁴⁷

Nowhere did the court examine the significance of the length of the creative chain from the plaintiff's work to the defendants' book, nor did it examine the importance of the differences at each link of the chain. Yet that type of analysis is essential in determining whether a copy that is not an exact copy is indeed a reproduction of expression and thus an infringement of the author's adaptation right of Article 27 of the Copyright Law.⁴⁸ Indeed, analyzing cases, such as this one, which involve translations and adaptations using only the Article 21 right of reproduction⁴⁹ analysis writes the Article 27 adaptation right completely out of copyright law.

The court did seem to engage in an analysis of the adaptation right when it discussed whether the defendants' book contained sufficient creativity to be an independent work. Using words that are similar to the definition of "derivative work" under the United States Copyright Law (although, interestingly, not very close to the wording in the Japanese Law), the court decided that "whether or not something newly created based on a preexisting work possesses creativity or individuality should have no bearing on whether or not the copyright in the preexisting work has been infringed."⁵⁰ However, the court never followed through with the adaptation right analysis this wording seemed to suggest would be applied.

The result in this case may have been different if the court had used adaptation right analysis, focusing on whether the defendants' work was a translation or other adaptation within the meaning of Article 27. To do this, the court should have considered whether the defendants' book, which contained a source code program written in assembly language as well as additional explanatory material not contained in the machine language version of the program, was an adaptation of the plaintiff's object code program written in machine language.

While it may not seem important which of the author's rights under copyright law have been violated, it is important in this case

47. *Id.*

48. Article 27 of the Copyright Law provides for an "adaptation right" of an author, stating that "[t]he author shall have the exclusive right to translate, arrange musically, transform, dramatize, cinematize, or otherwise adapt his work." This adaptation right under Japanese law is similar to the derivative works right under United States law.

49. Article 21 of the Copyright Law provides that "[t]he author shall have the exclusive right to reproduce his work."

50. Judgment of Jan. 30, 1987, Chisai (District Court), Tokyo, 1219 Hanji 48, 55.

because the holding of the court under traditional reproduction right analysis seems incorrect. The added explanatory material in the defendants' book and the translation from one programming language to another make the defendants' book much more than a simple reproduction of the expression of the plaintiff's work. Moreover, the two works are dissimilar and cannot be used interchangeably. The market for the plaintiff's work would not be harmed, and may even be helped, by sales of the defendants' book, since only those people who already had a copy of the plaintiff's work would get any benefit of a book explaining how to use the plaintiff's work. Thus, there is little question that the holding is not supported by the reproduction rights analysis the court used.

On the other hand, if the court had correctly used adaptation right analysis, its decision might still have been the same. Even though the plaintiff's market for the plaintiff's work would not be harmed by sales of the defendants' book, the plaintiff may later want to market its own book explaining the operation of its work. At that time, the market would have been affected by the sale of the defendants' book. The adaptation right is designed for just such a case, to protect the right of the author of an original work to exploit the market for any adaptations he or she may later want to create.

Of course, the author of a computer program should be entitled, under the adaptation right, to decide whether or not to make the source code for the work available to the public. Some information underlying a work is only available if one has access to the code of a program. Since ideas in a work are not protected by copyright law, the only way the creator of a program can even potentially keep certain ideas in a program secret is by keeping the source code secret.

In this case, the plaintiff probably chose to keep the source code secret to protect claimed trade secrets, and did not want the defendants to make the source code public and thus disclose what it might portray as valuable ideas underlying the work. The adaptation right will give limited protection of ideas in such a case because it will prevent even those who properly discover the ideas in a work from disclosing them *unless* they do so using independent expression. One of the purposes of the adaptation right should be to protect the author's right to decide not to create a particular type of adaptation, as well as to allow the author to create his or her own adaptations.⁵¹

51. This principle holds in most cases, but some countries provide for compulsory licenses to permit the unauthorized translation of a work in another language to the language of that country if the author does not authorize such a translation within a certain period of time. It is conceivable that the same principle might apply in some cases to computer software, but this author is unaware of any cases where it has been applied.

The court seemed about to use adaptation right analysis at another point in its decision. The court pointed out that the defendants' publishing of its book and presenting a source code version and explanation of the plaintiff's work when the author of the object code version of the work had chosen not to present such a book cannot be excused simply because having such a book available would be beneficial to users. Regrettably, the court went no further than this first step.

Had the court applied the correct analysis and decided whether the defendants' book was an adaptation of the plaintiff's work, its decision would have been much more helpful. Many important questions regarding adaptation right analysis as applied to computer programs were squarely presented by the facts of this case. Does the author of a new program based on a preexisting program infringe the adaptation right by copying only the idea of the preexisting program? Is it an infringement of the adaptation right to translate an algorithm, which is the basic method of solution used in a computer program, from one programming language to another?⁵² Does protection under the adaptation right ever protect the idea as well as the expression in computer programs? Unfortunately, the court only hinted at "no" answers in its opinion, and thus failed to address adequately any of these important questions.

2. Copyrightability of Operating System Programs

The holding of the court that operating system programs meet the criteria for copyrightable materials is well founded. The Japanese Copyright Law lists certain criteria in its definition of a "work" in Article 2(1)(i): "'work' means a production in which thoughts or sentiments are expressed in a creative way and which falls within the literary, scientific,⁵³ artistic or musical domain."⁵⁴

These criteria are met just as fully by operating system programs as by applications programs. If the instructions and commands of an application program are an expression of the "thoughts or sentiments" of the programmer, then the instructions and commands of an operating system program are no different. Indeed, the instructions and commands used are the same for operating system programs and applications programs.

52. This question is even more important under the amended Copyright Law, since Article 10(3) now expressly provides that the underlying algorithm in a computer program is not protected under the Law.

53. Although the word "*gakujutsu*" used here is sometimes translated as "scientific," "scholarly" would be more accurate. See *supra* note 35.

54. This translation is taken from the translation by the Agency of Cultural Affairs, see *supra* note 4. A better translation of Article 2(1)(i) of the Copyright Law might be: "'work' means a creative expression of thoughts or emotions which falls within the domain of literature, scholarship, art or music."

It is true that the task of an operating system program differs from the task of an applications program. However, the court correctly rejected the defendants' contention that this difference is important. The defendants argued that the only purpose of the plaintiff's work is "to manage data efficiently and quickly."⁵⁵ Yet the same can be said of the purpose of all programs, whether they be operating system programs or applications programs. Indeed, the sole use of computers is for the quick and efficient management of data. This purpose, therefore, certainly fails to distinguish operating system programs from applications programs.

There is little room to quarrel with the holding of the court that operating system programs are just as much "works" as applications programs. Both contain the thoughts of the author, and both are scholarly works.

D. Conclusion

The court in *Microsoft* did address some of the important issues in deciding that the defendants had violated the plaintiff's copyright. The opinion does clearly state that publishing a book that translates, interprets, and adds explanatory labels and comments to an object code program is infringement of the copyright in the object program under the pre-amendment Japanese Copyright Law. Moreover, it also states that operating system programs are copyrightable material for the same reasons that applications programs are copyrightable material. These holdings are important. However, the court left unaddressed some interesting and crucial issues posed by the facts in this case. Most importantly, it failed to answer one critical question: when does a software creator pass the line between a mere reproduction or adaptation, which is an infringement, and an independent work, which though based on another work is sufficiently different to be non-infringing? At least a partial answer to this question, fortunately, was forthcoming in the *System Science* case discussed below.

III. THE SYSTEM SCIENCE CASE

The decision of the Tokyo High Court in the case of *System Science K.K. v. Toyo Sokuki K.K.*⁵⁶ addresses the important question of how different a program must be from a program it is based upon for it to be a legitimate independent creation rather than an infringing work. It also clears up another important issue, that of similarity induced by hardware constraints, by stating that substan-

55. Judgment of Jan. 30, 1987, Chisai (District Court), Tokyo, 1219 Hanji 48, 55.

56. Judgment of June 20, 1989, Kōsai (High Court), Tokyo, 1322 Hanji 138. See Appendix 2 of this article for an English translation of the *System Science* decision.

tial similarity in expression will not be infringing if the expression is constrained by hardware or other requirements so that similarity cannot be avoided.

A. Facts

The Tokyo High Court heard the *System Science* case on appeal from the denial by the Tokyo District Court⁵⁷ of a petition for provisional relief, asking that copyright infringement be stopped.⁵⁸ The lower court had held that provisional relief should not issue for any of the four programs⁵⁹ that the plaintiff, System Science K.K., had alleged were infringed. This was based on the finding of the lower court that the defendants, Toyo Sokuki K.K. and K.K. Nihon Technart, had contracted for the development of replacement programs for the allegedly infringing programs, and that the defendants would sell only these replacement programs, thus making provisional relief unnecessary. However, the lower court also held that one of the programs, the CA-9 program, was not an infringement of the plaintiff's corresponding program, the CA-7 II program.

The copyright dispute grew out of a business relationship between the plaintiff and the defendants. There was no question that the defendants had access to, and based their programs on, the programs which were copyrighted by the plaintiff. Indeed, the defendants argued that the four programs had actually been created as works-made-for-hire, so that the defendants, rather than the plaintiff, owned the copyright. The various parties had all marketed the programs as incorporated in Read Only Memory (ROM) of a hardware device.

B. Judgment

The appeals court decided three main issues in its opinion. First, it quickly rejected the defendants' argument that they, rather than the plaintiff, were the owners of the copyright in the programs

57. Both the Tokyo High Court and the Tokyo District Court, with its Twenty-Ninth Civil Division, have divisions which specialize in intellectual property cases. Because these divisions of both courts reached basically the same results on the important issues involved in the *System Science* case, their decisions will probably be given great weight.

58. Under Japanese civil procedure, provisional relief proceedings are treated as a separate case from proceedings for other relief, and are usually heard by a single judge, although normal proceedings are usually heard by a three-judge panel. It is common, however, for the findings in a provisional relief proceeding to be given some deference by the judges in a later proceeding for permanent relief.

59. The programs were software to be used with hardware in biotechnology measuring and analyzing tools. It appears from the court opinions that the plaintiff's CA-7 II program, which the court found not to have been copied or adapted by the defendants, was an input/output program for a printer.

at issue. Second, the court disagreed with the holding of the district court that there was insufficient cause to order provisional relief regarding the sale or advertisement of the three programs the district court had found to be infringing.

The third holding, that the defendants' CA-9 program was not an infringing adaptation of the plaintiff's CA-7 program, is the most important issue the court discussed. The court held that the defendants' program was not an adaptation because it was not sufficiently similar to the creative, or protected, aspects of the plaintiff's program. The only similarities were in simple and common routines, in routines that had to be similar because of hardware constraints, in the basic algorithm used, or in design choices that were dictated by common sense.

The court noted that one of the requirements for finding that one program infringes the copyright in another is that "a part of the combination of instructions of the program work [must] be found to be creative, and . . . the combination of instructions of the later-created program [must] be similar to the part of the program work which can be found to be creative."⁶⁰ In other words, the similarities between the two programs cannot be in just any aspect of the programs, but rather in an aspect which is protected under copyright law, i.e., one which the court refers to as a "creative" aspect.⁶¹

The court also noted that an algorithm, which the court refers to as the "flow of processing" of a program, is not protected under copyright law. The court equated the terms "flow of processing," "algorithm,"⁶² and "*kaiho*" ("solution" or "method of solution"), thus interpreting the term "*kaiho*" in Article 10(3)(iii) of the Japanese Copyright Law as meaning algorithm. The court held that similarities in the basic algorithm underlying a program should not be considered in determining whether infringement has occurred, for the algorithm is a part of a work which does not receive protection, and thus has no relevance to the creativity of a program.

The court noted the difference in the length of the two programs; the CA-7 II was about 12,000 bytes long compared to the

60. Judgment of June 20, 1989, Kōsai (High Court), Tokyo, 1322 Hanji 138, 140.

61. The use by the court of the word "creativity" seems to be shorthand for indicating what is protected under copyright law, while lack of creativity indicates what is not protected. Karjala, *Japanese Courts Interpret the "Algorithm" Limitation on the Copyright Protection of Computer Programs*, 7 EUR. INTEL. PROP. REV. 235, 237 (1990). Another commentator cites the *System Science* case in support of his theory that the creativity requirement under Japanese copyright law operates the same way the merger doctrine does under American copyright law, and that courts will find that a work is not sufficiently creative in cases where the expression merges with the idea. Yamamoto, *The Concept of Originality and the Merger Doctrine in Copyright Law*, 456 NBL 27 (1990) [in Japanese].

62. Using the Japanese transliteration "*arugorizumu*" of the English word "algorithm."

763 bytes of the allegedly infringing CA-9 program. After considering the comparative shortness of the allegedly infringing program, and the very small number of bytes identified by the plaintiff as overly similar, the court held that "we simply cannot be convinced that program CA-9 is an adaptation of program CA-7 II."⁶³

C. Comment

The opinion in *System Science* is brief, and the important part, for the purposes here, even briefer. It is, however, very helpful. First, as does the *Microsoft* opinion, this decision indicates that Japanese courts will strongly enforce copyrights in computer software; here, the appeals court granted provisional relief with regard to three programs out of the four at issue.⁶⁴ More importantly, however, the *System Science* opinion addresses the issue of adaptations (or derivative works) and gives several rules for determining whether infringement has occurred in cases which do not involve exact copying. The *Microsoft* court should have accomplished this, but did not.

The *System Science* court stated that the constraints on writing a computer program imposed by the choice of the symbols and strict grammars of programming languages will inevitably result in similarities between two programs which are created to achieve the same result.⁶⁵ Determinations of program infringement must therefore employ great care, and similarities arising from the constraints inherent in program writing will not be considered evidence of infringement.

The court applied the general rule that naturally arising similarities will not be considered infringing to the specific facts of the *System Science* case. The court noted that a particular routine constrained by hardware characteristics would lead to unavoidable similarities, and held that such similarities are not evidence of infringement. The court also observed that both programs at issue had adopted the same "very common combination of instructions"

63. Judgment of June 20, 1989, Kōsai (High Court), Tokyo, 1322 Hanji 138, 140.

64. This continues the trend shown in the *Microsoft* case discussed above. Although the Japanese court system has been criticized by some as not providing sufficient protection to intellectual property rights, the *Microsoft* and *System Science* cases seem to indicate that such criticism is not justified in the case of copyright protection for computer software.

65. This holding was criticized in Hirakawa & Nakano, *supra* note 10, at 56 n.51. Those authors state the *System Science* court failed to consider that one usually has an infinitely broad choice of expression in writing even relatively simple computer programs and even when a particular result is intended. While there is some room for varying expression in the cases cited by the court, there is usually very little choice, and all choices will have similarities. For example, if one wanted to describe the game of poker, one could do it in many different ways, but all the ways will sound quite similar because they are describing the same game.

for a certain routine, and held that similarities in common or very ordinary routines will not be considered evidence of infringement.⁶⁶ Finally, the court pointed out that one similarity between the two programs, the location of the subroutine stack, was dictated by common sense, holding that similarities arising from common sense will also not be considered evidence of infringement.⁶⁷

Returning to its explanation of general principles, the court noted that under Article 10(3)(iii) of the Copyright Law similarities in the flow of processing are not evidence of infringement. Although that seemed the case from the language of Article 10(3)(iii), the *System Science* court made it clear that the exception for "*kaiho*" does cover what are commonly called algorithms. Similarities in the underlying algorithm between two programs will not be considered evidence of infringement.

The opinion as a whole indicates that the *System Science* court was willing to give strong protection in cases of verbatim or nearly verbatim copying, but that it was not willing to give one company the right to exclude a competitor from a market simply because the competitor's program was based on and was similar to the initial work. It is likely that courts will not consider in infringement suits similarities in algorithm, in expression constrained by hardware, in common or ordinary expression, or in expression dictated by common sense. Such similarities are not evidence of the copying of protected expression from one program to another. Courts will, however, consider similarities in expression that indicate that one company simply copied the literal code of its competitor. In other words, one is free to copy idea, but not expression.

Because of the brevity of the *System Science* opinion, one must be careful not to read more into the opinion than is really there. Some commentators⁶⁸ were concerned that *System Science* added a new concept of "originality"⁶⁹ similar to the high originality stan-

66. Judgment of June 20, 1989, Kōsai (High Court), Tokyo, 1322 Hanji 138, 140. This is probably similar to the merger doctrine, where similarities in expression which are inevitable because expression and idea have merged are not considered to be evidence of infringement.

67. This category of permissible similarity might, for example, be used to justify the common use of alphabetical order in expressing a list of terms. Since alphabetical order is a common sense method of ordering terms, it should not be considered evidence of infringement.

68. See, e.g., Dairaku, *Injunction Against Reproduction of Computer Program*, [1989] 7 IP ASIA 25; Nihon Keizai Shimbun, Oct. 30, 1989 [in Japanese].

69. The term "*sosakusei*" used several places in the *System Science* opinion can be translated to mean either "originality" or "creativity," and both terms have been used in English-language discussions of the case. Compare Dairaku, *supra* note 68 (using the term "originality") and the to-be-published English translation of his article by Yamamoto, *supra* note 61 with Karjala, *supra* note 61, at 236 n.19 and Ozaki, *supra* note 11 (using the term "creativity").

As noted by Karjala, use of the term "originality" may cause confusion with the

dard required by the German courts. While that concern cannot lightly be dismissed, it seems unlikely that such a requirement was added to Japanese jurisprudence merely by a few remarks in a single opinion. As discussed above, rather than establishing a new requirement, it is more likely that the *System Science* court was simply using the term "creativity" to refer to the protectability of certain elements of a work.

IV. REVERSE ENGINEERING

When discussing the protection of computer software under Japanese copyright law, an important question is whether reverse engineering violates that law. Reverse engineering involves the study and research of products available in the marketplace. Commonly practiced by scientists and engineers around the world, reverse engineering is expressly permitted by intellectual property law statutes in many countries.

Even though reverse engineering is usually expressly allowed in intellectual property law statutes, even when it is not expressly allowed - as is often the case under copyright law statutes - the policy reasons for allowing, and indeed encouraging, reverse engineering apply with regard to all forms of intellectual property. Consequently, there is general agreement that reverse engineering is permissible under all intellectual property law statutes.

Japanese law expressly allows reverse engineering under its patent and semiconductor chip protection laws. While Japanese copyright law does not expressly discuss the issue, the notion that reverse engineering is not prohibited by copyright law is accepted by most commentators. Recently enacted trade secrets legislation in Japan was explained by its drafters as allowing reverse engineering, consistent with United States trade secrets law. Accordingly, although the legitimacy of reverse engineering in the copyright law context has not been expressly stated in statute or court decision, it is commonly engaged in by Japanese software developers and generally accepted as a legitimate practice.

A. Reverse Engineering Under Copyright Law

Reverse engineering is an issue relating to technology, rather than to the arts. Consequently, the Japanese Copyright Law, which is concerned chiefly with traditional, artistic works, does not have

American copyright law principle of "originality." Karjala, *supra* note 61, at 236 n.19. As in most cases when comparing two different legal systems, it is probably better to avoid using terms of art from one legal system in describing a principle in the other legal system. Using terms of art makes it too tempting to emphasize similarities between two different principles, when it is usually more important to emphasize differences. Accordingly, "creativity" seems to be the better choice in this case.

any provisions which relate specifically to reverse engineering.⁷⁰ However, after the 1985 amendments to the Copyright Law expressly extended copyright protection to computer program works, the issue of the permissibility of reverse engineering of computer programs under the Japanese Copyright Law has become important. Although Japan does not have a judicial decision that specifically addresses research or reverse engineering of computer program, there has been a great deal of discussion among Japanese professors and other commentators. With few exceptions, this discussion does not question that reverse engineering is permitted, but rather centers on which among several competing theories offers the best rationale. As discussed above, one case, *Microsoft Corp. v. Shuuwa System Trading K.K.*,⁷¹ did involve reverse engineering. In that case, the defendant disassembled the plaintiff's BASIC interpreter, added its own explanatory comments and labels, and then published the results in a book that was commercially distributed. The court found that the defendant had infringed the plaintiff's copyright. However, even one of the plaintiff's attorneys stated that the court clearly did not rule that reverse assembly was necessarily an act of unauthorized copying, but only that reverse assembly is unauthorized copying when its results are used to create a program listing which is published as a book.⁷² The attorney noted, in fact, that the issue of the legality of reverse assembly was never even an issue in the *Microsoft* case.⁷³

Although Japan relies on a civil law system, it is not highly unusual in Japan for legal principles to be developed also through case law and commentary.⁷⁴ The idea/expression dichotomy, for example, is a well-established principle of Japanese copyright law. While discussion of reverse engineering in Japan is a fairly recent development, there is general agreement that reverse engineering is permitted by Japanese copyright law. However, there is also agreement that if one uses reverse engineering to create a program substantially similar in expression to an original program, the fact that reverse engineering was used does not change the fact that the program is infringing.

One of the theories concerning the legal basis for reverse engineering under copyright law is that reverse engineering must be allowed because prohibiting it would violate the purpose of the Copyright Law. Article 1 of the Copyright Law provides that

70. N. NAKAYAMA, *supra* note 11, at 130 [in Japanese].

71. Judgment of Jan. 30, 1987, Chisai (District Court), Tokyo, 1219 Hanji 48.

72. *See supra* note 41.

73. *Id.*

74. In contrast to the United States, however, in Japan commentary by a recognized authority on a legal subject is sometimes considered as authoritative as a court decision.

[t]he purpose of this Law is, by providing for the rights of authors and the rights neighboring thereto with respect to works as well as performances, phonograms, broadcasts and wire diffusions, to secure the protection of the rights of authors, etc., *having regard to a just and fair use of these cultural products*, and thereby to contribute to the development of culture.⁷⁵

Article 1 contains a general principle of fair use, and Articles 30 through 47*bis* of the Copyright Law give specific examples of permitted uses of copyrighted works.

Even though Articles 30 through 47*bis* specify particular permitted uses, these Articles are not the exclusive test of "fair use"; instead, factors such as those in the fair use provisions of the United States Copyright Act may be considered in allowing other fair uses.⁷⁶ From this standpoint, reverse engineering could be considered a fair use under Japanese copyright law.⁷⁷

A similar theory is that Article 47*bis* of the Copyright Law, one of the 1985 software-related amendments to the Copyright Law, allows reverse engineering by analogy. Article 47*bis*(1) provides that

[t]he owner of a program work may make copies or adaptations (including making copies of a derivative work created by means of adaptation) of that work if and to the extent deemed necessary for the purpose of using that work in a computer by himself

Although the right granted in this Article applies only to copies and adaptations necessary to use a program, the theory underlying this Article should apply to reverse engineering as well. Article 47*bis* allows the owner of a program to make a legitimate use of his program in a particular set of circumstances even if this use may involve acts that might technically be considered restricted acts. To deny the user this right would severely limit the usefulness of the program, since the user would have to turn to the copyright owner every time he wanted to make even simple modifications and corrections. On the other hand, allowing the user this right has little, if any, effect on the value of the copyright owner's rights, since only a legitimate user of a program copy can benefit from it, and he is restricted from passing on any copies or adaptations he makes to others.

Reverse engineering is an analogous legitimate use. As with the circumstances described in Article 47*bis*, prohibiting reverse engineering would have severe effects, since by preventing others from discovering the ideas in a program it would effectively grant the

75. Emphasis added. This translation of Article 1 is slightly different from that of the Cultural Agency's quasi-official translation.

76. Abe, *Example Where Copyright Was Recognized in Source Programs (Operating System Programs)*, 1247 Hanji 205, 208 (1987)[in Japanese].

77. *Id.*; see also N. NAKAYAMA, *supra* note 11, at 130-31 [in Japanese].

copyright holder a monopoly on those ideas. This would violate the basic tenet that copyright law does not protect ideas. On the other hand, allowing reverse engineering has little, if any, impact on the copyright owner's rights to the extent that it is done only to extract unprotectable ideas and principles from a program, not to copy expression. Therefore, Article 47*bis* justifies reverse engineering by analogy.⁷⁸

Some have argued that one should not be able to use information obtained from reverse engineering to create a program which competes with the program that was analyzed even if only ideas are used and not expression. That argument is incorrect, since the copyright owner has no rights in the ideas underlying his work. Not allowing the results of reverse engineering to be used freely, even to compete, would allow copyright to be misused to gain protection over unprotectable ideas. Accordingly, the limits on reverse engineering must be the limits of the copyright owner's rights, which do not extend to ideas.

One must always remember, however, that reverse engineering does not provide an excuse for copying expression. Superficial changes made by means of reverse engineering techniques in an attempt to disguise the copying of expression will be ignored in finding infringement. Reverse engineering should not be allowed to provide relief for software pirates, just as Article 47*bis* of the Copyright Law should not be interpreted to allow the owner of a program copy to abuse his rights at the expense of the copyright owner.

Another legal justification for reverse engineering is found in Article 10(3)(iii) of the Copyright Law, which specifically excludes programming languages, rules, and algorithms from protection under copyright law. If the user of a program were unable to reverse the engineering of a program and analyze the underlying rules and algorithms, the user would gain de facto protection of these elements. To ensure that these otherwise unprotected elements do not receive de facto protection, reverse engineering must be allowed.⁷⁹

78. Miki, *Currents in Intellectual Property Rights (Part 2)*, 929 JURISUTO 60 (1989) [in Japanese].

79. One of the key people at the Copyright Section of the Agency for Cultural Affairs responsible for drafting the software-related amendments adopted this theory in her explanation of the 1985 amendments to the Copyright Law:

The purpose of the Copyright Law is to protect creative "expression," so it does not protect the languages used as the means of expression or the principles and rules underlying the expression. Not only should this fact be considered when determining the scope of protection of copyrighted works, but also when determining the scope of the adaptation right, and because some believed that the scope of the adaptation right regarding programs might be ambiguous, programming languages, rules and algorithms were added as examples of non-protected items to elimi-

A similar argument is that prohibiting reverse engineering would give strong, patent-like protection to ideas even when the requirements for a patent have not been met.⁸⁰ The patent law is designed to protect highly creative ideas for a limited time. In contrast, copyright law does not extend protection to ideas, but only to expressions of ideas, and therefore does not contain any of the patent law restrictions necessary to ensure that protection of ideas is not too broadly given. Because a prohibition on reverse engineering would give patent-like protection to ideas in the guise of copyright protection, reverse engineering must be allowed.

The anti-competitive effects of prohibiting reverse engineering have also been cited as a compelling reason to interpret copyright law to allow it.⁸¹ Indeed, Japanese patent law, its semiconductor chip protection law, and its laws protecting know-how all either explicitly or implicitly allow reverse engineering. There is no reason to interpret Japanese copyright law any differently from those other laws. Given these provisions of Japanese law, and the scholarly commentary, it would be remarkable indeed for a Japanese court to find that reverse engineering of a program to extract unprotected elements constitutes copyright infringement.

Some have argued under a very restrictive reading of the Copyright Law that any reverse engineering of object code is an act of infringement. They argue that object code cannot easily be understood by humans unless it is reproduced in assembly language form, that reverse engineering therefore requires reproduction or adaptation, and that this reproduction constitutes copyright infringement. While such a reading is certainly possible, it is not supported by either policy reasons or by common sense.⁸² For the reasons set forth above, the better view is that limited reproduction or adapta-

nate any uncertainty that copyright does not extend to this type of element.

That programming languages, rules and algorithms are not protected means that [the copyright owner's] rights, such as the right of adaptation, do not prevent analyzing a program, extracting the ideas and algorithms on which the program is based, and then using them to create a program with completely different expression.

Bando, *supra* note 11, at 7 [in Japanese].

80. Nakayama, *Trends in Intellectual Property Law: Focusing on the Protection of Software*, 30 ELECTRONICS INDUS. MONTHLY No. 9, at 8, 19 [in Japanese].

81. Negishi, *Intellectual Property and Antimonopoly Law*, in INTELL. PROP. & ANTIMONOPOLY L. 39 (1989) [in Japanese].

82. This kind of restrictive reading of copyright law principles would lead to absurd results. One could argue under this theory, for example, that anyone who reads a book in Japanese by translating the words he or she does not understand on paper, or even in his or her head, is infringing the copyright in the book by making the translation. Taken to its logical conclusion, such a theory could even be used to justify calling an "unauthorized" act of reading an act of infringement, since the printed page is optically reproduced on the retina of the reader's eye. That type of restrictive interpretation strains copyright law principles beyond reason.

tion necessary to put the object code in human-readable form is, if an act of infringement at all, a *de minimis* violation not actionable under the Copyright Law.⁸³

United States law is similar. Although there is no express provision in the United States Copyright Act relating to reverse engineering, the majority view among courts⁸⁴ and commentators⁸⁵ is that reverse engineering of computer programs is permissible under United States copyright law.

B. Reverse Engineering Under Patent Law⁸⁶

In general, unless a patent owner has granted an exclusive license to another, he has the exclusive right in Japan to practice his patented invention in business. Any unauthorized practice of the invention in business by another is, in principle, an infringement of the patent, giving rise to tort liability under Article 709 of the Japanese Civil Code. However, because practicing an invention "in business" means practicing it in commercial activities rather than for personal use, practicing an invention at home would not be an infringement.⁸⁷ Thus, any reverse engineering which qualified as private or personal use would be permitted.

In practice, however, most reverse engineering is not done for private or personal reasons, but instead is done by scientists and engineers in their professional capacities. Under Japanese patent law, this is also expressly permitted. Article 69(1) of the Patent Law provides that "[t]he effect of a patent right does not extend to the practicing of the patented invention for the purposes of experimentation or research." Since reverse engineering is usually, if not

83. Even if the translation from machine language to assembly language were considered an unauthorized reproduction, Japanese copyright law would not provide any effective remedy unless the results of the reverse engineering were used to create another program that is substantially similar in expression to the program which was translated. If the results of reverse engineering were used to create a non-infringing program, the copyright owner's only remedies would probably be limited to obtaining the destruction of the unauthorized reproduction and compensation for making that reproduction. There would be no basis for an injunction or damages relating to the non-infringing program. Ozaki, *supra* note 11, at 960.

84. See, e.g., *NEC Corp. v. Intel Corp.*, 10 U.S.P.Q. 2d (BNA) 1177 (N.D. Cal. 1989); *Vault Corp. v. Quaid Software Ltd.*, 847 F.2d 255 (5th Cir. 1988).

85. The *LaST Frontier Conference Report on Copyright Protection of Computer Software*, reprinted in 30 *JURIMETRICS* 15 (1989), presents the conclusions of ten leading intellectual property academic specialists who met and heard presentations from the software industry. They concluded that reverse engineering carried out to extract and use unprotectable elements of a program, such as program logic and ideas, ought not to constitute infringement under United States copyright law.

86. Because the provisions relating to reverse engineering in the Utility Models Law correspond to those in the Patent Law, the discussion relating to the Patent Law can be assumed to apply equally to the Utility Models Law.

87. Amemiya & Nishimoto, *Patents and Utility Models* § 2.15[1], in *DOING BUSINESS IN JAPAN* (Z. Kitagawa ed. 1987).

always, conducted for "experimentation or research" purposes, practicing an invention in the course of reverse engineering is specifically permitted under Japanese law.

It may seem unnecessary to provide for reverse engineering of a patented invention, since the patent specification is supposed to tell those skilled in the art all they need to know to practice the invention. Therefore, it may seem that studying the patent specification would be sufficient to give all the information necessary for experimentation or research.

While this sounds good in theory, it is often necessary to practice an invention to conduct adequately experimentation or research. Such practice promotes scientific and technical progress, allowing others to learn from a patented invention, without causing significant economic harm to the patent owner. Of course, should one attempt to exploit the results commercially by producing a product or process which falls within the scope of the patent, the fact that experimentation and research were permissible does not mitigate the infringing nature of the exploitation.

Although United States patent statutes do not expressly provide exceptions for research and experimentation, case law does,⁸⁸ although the limits of the exception are not clear. At the least, however, case law strongly suggests that the exception covers practicing an invention either to better understand how that invention works, or to discover improvements on or alternatives to it. The European Community Patent Convention, although not yet in force, also provides that experimental working of a patented invention is not an infringement.⁸⁹ As in Japan, this kind of experimental use is permitted because it promotes scientific and technical progress without significantly harming the patent owner's rights.

C. Reverse Engineering Under the Semiconductor Chip Protection Law

The Japanese Law Concerning Circuit Layouts for Semiconductor Integrated Circuits also expressly allows for reverse engineering. Although the reproduction of a semiconductor chip circuit

88. See, e.g., *Dugan v. Lear Avia*, 55 F. Supp. 223, 61 U.S.P.Q. 404 (S.D.N.Y. 1944), *aff'd* 156 F.2d 29, 69 U.S.P.Q. 357 (2d Cir. 1946); *Akro Agate Co. v. Master Marble Co.*, 18 F. Supp. 305, 33 U.S.P.Q. 335 (N.D. W. Va. 1937). See also *Douglas v. United States*, 181 U.S.P.Q. 170 (Ct. Cl. Trial Div. 1974) (stating that only limited use of an invention is permitted only for a short time even if the use is solely experimental).

89. Luxembourg Convention on the Community Patent of Dec. 15, 1975, art. 31(b), which provides that:

The rights conferred by a Community patent shall not extend to:

.....

(b) acts done for experimental purposes relating to the subject-matter of the patented invention.

layout is generally considered to be an infringing act, Article 12(2) of the Law provides that the right to use a registered circuit layout does not extend to the act of using the layout in creating a semiconductor chip circuit for analysis and evaluation purposes.

The United States Semiconductor Chip Protection Act is similar, providing in 17 U.S.C. § 906 that it is not an infringement to reproduce a semiconductor chip mask work

solely for the purpose of teaching, analyzing, or evaluating the concepts or techniques embodied in the mask work or the circuitry, logic flow or organization of components used in the mask work; or . . . to incorporate the results of such conduct in an original mask work which is made to be distributed.

Here, as with patent law, reverse engineering itself is expressly permitted, but the fact that reverse engineering was used in the creation of an infringing mask work does not excuse distribution of an infringing mask work. An infringement is still an infringement.

D. Reverse Engineering Under Trade Secrets Law

Japan recently amended its Unfair Competition Prevention Law to centralize and strengthen the protection currently given to trade secrets under its tort and unfair competition laws. Since the new law was passed just a few months ago, it is difficult to tell how it will be interpreted. Many of the United States have offered trade secret protection for many years, so it is instructive to consider how reverse engineering is treated under United States law. This is particularly true because the drafters of the Japanese trade secrets law stated that Japanese law would allow reverse engineering in the same fashion as does American law.⁹⁰ Moreover, Japanese courts will probably look to foreign judicial precedent, particularly that in the United States and Germany, when there is no Japanese court decision which provides any guidance.⁹¹

Although most state laws in the United States that relate to trade secrets do not expressly provide that reverse engineering is a fair and honest means of discovering trade secrets, the United States

90. Mr. Yuuji Tanahashi, head of the Industrial Policy Bureau of the Ministry of International Trade and Industry (MITI), stated before the Commerce and Industry Committee of the Japanese House of Representatives that reverse engineering is commonly engaged in, and would not be considered an unfair act under the trade secrets law. Hearings of Commerce and Industry Comm. (No. 8), Japanese House of Rep., 108th Sess., June 13, 1990, at 8 [in Japanese]. He stated that this would be in accord with the practice in the European countries and states in the United States that offer trade secret protection. *Id.*

91. This contrasts with the American court practice, where foreign court decisions are rarely considered to have any persuasive authority. In Japan, it is quite common for United States court decisions to be studied by lawyers and cited in briefs to a court, where appropriate.

Supreme Court has clearly so stated. In an oft-cited case, the court said that

[a] trade secret law, however, does not offer protection against discovery by fair and honest means such as . . . by so-called reverse engineering, that is by starting with the known product and working backward to divine the process which aided in its development or manufacture.⁹²

The court reaffirmed its position in a more recent case, noting that the public at large is free to exploit any information concerning so-called "trade secrets" that can be discovered through reverse engineering of products in the marketplace.⁹³

In its opinions regarding reverse engineering and trade secrets law, the United States Supreme Court suggested several benefits from allowing reverse engineering. The Court noted that reverse engineering often leads to significant advances in science and technology, and that the competitive reality of reverse engineering may act as a spur to the inventor, creating an incentive to develop inventions which meet the rigorous requirements of patentability. The Court also pointed out the benefits of competition in technical fields and the necessity of bringing new designs and technologies into the public domain.

The Court also implied that the protection offered under trade secrets law should not be so great as to divert inventors from more creative efforts which deserve greater protection under patent law. If the level of protection available under one intellectual property law does not accurately correspond to the requirements of that law, the system of protection will not operate properly. Consequently, it would be improper for an inventor to obtain patent-like protection under trade secret law or, as discussed above, copyright law, without meeting the patent requirements of inventiveness and disclosure.

As the drafters of the Japanese trade secrets law indicated, it is intended that reverse engineering of trade secrets be allowed in Japan just as in the United States. The same policy reasons apply, and the same benefits of allowing reverse engineering - promoting the advance of science and technology - can be obtained in Japan as in the United States.

D. Conclusion

As discussed above, the United States Supreme Court has noted several benefits of allowing, and even encouraging, reverse engineering in the context of trade secrets protection law. These same benefits apply to all forms of intellectual property, and in both Ja-

92. *Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470, 476 (1974).

93. *Bonito Boats, Inc. v. Thunder Craft Boats, Inc.*, 109 S. Ct. 971, 980 (1989).

pan and the United States. Therefore, it seems unlikely that a compelling or valid case for prohibiting reverse engineering can be made under copyright law in Japan.

V. CONCLUSION

Intellectual property rights are largely territorial, and the rights given to a copyright, patent, or trademark owner are generally based on the law where a work is created or published or where a discovery or trademark is registered. Even so, the trend toward a single, global market without borders affects intellectual property law, leading naturally to harmonization of law even apart from formal harmonization efforts. This trend is evident in the protection provided to computer programs under copyright laws worldwide, as the legal systems of Japan, the United States, and the European countries converge on harmonized protection of computer programs.

The study of Japanese statutes and court decisions can help determine the most appropriate level of protection for computer programs under copyright law. In particular, the success Japan has had with its express exclusion from copyright protection of programming languages, rules (interfaces and protocols), and algorithms can be instructive to other countries, among them the United States, that have struggled to define the proper scope of protection.

APPENDIX 1

Microsoft Corp. v. Shuuwa System Trading K.K.

Jan. 30, 1987, Tokyo District Court, 1219 Hanrei Jihou 48

ORDER

1. The defendants will not publish or distribute the books described in the Appendix.
2. The defendants will destroy all books described in part one of this Order.
3. The defendants will bear the costs of this litigation.

FACTS

I) RESULT SOUGHT BY THE PARTIES

1. Gist of the Claims
Same as the Order.
2. Defense to the Gist of the Claims
Rejects all of the plaintiff's claims relating to the defendants.
Costs of the litigation should be borne by the plaintiff.

II) ALLEGATIONS OF THE PARTIES

1. Grounds for Claims

(1) The plaintiff is a company organized under the laws of the American State of Washington, with its principal business being the development, production, and selling of software used in personal computers.

(2) The plaintiff is the copyright owner of the work described in the Appendix (hereinafter referred to as "work at issue").

The work at issue was produced and marketed in the manner described below.

(i) The plaintiff was asked on April 10, 1979 by the Nihon Electric Company (hereinafter referred to as "NEC") to develop a BASIC interpreter program for use on the PC-8001, and the plaintiff produced the program that is the work at issue.

(ii) The work at issue was developed, requiring about eight months, by technical people of the plaintiff, other than its officer William H. Gates, as part of their employment.

The copyright notice of the copyright holder, "Copyright 1979 (c) by Microsoft," was programmed to appear on the display when the program is executing.

(iii) The plaintiff recorded the above-mentioned work at issue it had developed on floppy disks and delivered it to NEC, which reproduced it in ROM and sells it as incorporated into the PC-8001 personal computer (hereinafter referred to as the "personal computer at issue").

(iv) The plaintiff has not publicly sold or distributed the work at issue in the United States.

As discussed above, because the work at issue is based on the original ideas of the plaintiff, was created in the course of employment by persons engaged in the plaintiff's business, and was made public under the name of the plaintiff, the plaintiff can be said to own the copyright in it.

(3) As alleged in the preceding paragraph, the works at issue are protected by copyright law in Japan under Article 6(2) and (3) of the Copyright Law.

(4) The work at issue is essentially as summarized below.

Specifically, the work at issue is a source program for the BASIC interpreter which is expressed in a symbolic language (assembly language) and incorporated into the personal computer at issue, and this program is an operating system which interprets commands or programs in the BASIC language which are input into the small-scale microprocessor-based computer system that is the personal computer at issue, and which translates word for word instructions to control the microprocessor and other information.

Moreover, the work at issue has been translated into binary electronic signals (binary code) which the microprocessor can understand and was copied into and stored in the ROM which forms the computer system memory of the personal computer at issue (this program which has been translated into machine language is hereinafter referred to as the "object program at issue").

(5) The work at issue, as discussed below, is within the scope of copyrightability under the copyright law.

(i) In general, computer programs are expressions in the form of combinations of instructions with the purpose of causing a computer to function and achieve a particular result, with these instructions being produced by combining various programming language commands and other information. Moreover, programs are expressions of the advanced scientific thinking of the program's creator and have individual characteristics, and therefore fall within the purview of Article 2(1)(i) of the Copyright Law, which reads "thoughts which are expressed in a creative way, and fall within the . . . scientific . . . domain," and also within the purview of Article 10(1)(i), which reads "other literary works."

(ii) With regard to the work at issue, it provides the ability to easily operate the personal computer at issue in dialogue style using the BASIC language, and is designed to process BASIC commands or programs as written and output the result desired by the person who input those commands or programs.

The work at issue was written using the symbolic language (assembly language) of the Z-80 microprocessor made by the Amer-

ican company Zilog, the microprocessor which is used in the central processing unit (CPU) of the personal computer at issue, and was produced by combining commands and other information in that language in order to match the hardware architecture of the personal computer at issue and to fully utilize its capabilities. Consequently, the overall structure of the program, the structure of each routine, and the combining of the commands and other information at each address were created by calling upon the high degree of knowledge of the program's creator regarding computers and programming languages, and the work at issue can therefore be regarded as a literary work within the scientific domain.

(iii) The work at issue has a total of 229 routines (sets of instructions to execute a particular function), and from those a module called COINIT (a subroutine to initialize the cassette interface) from a routine called the CSAVE routine (which is a set of instructions to save onto cassette tape a program which is in the computer's memory) will be taken as a specific example. When the contents of each instruction are analyzed, and the work at issue explained as the creative expression of thoughts, we have the following.

(The address number, assembly language statement, and its substance are described in that order.)

(1) 1EC0-1EC2 CALL L0C46 (COINIT)

Call the routine COINIT which begins at location 0C46.

(2) 0C46-0C48 CALL LF1B9 (HOOK27)

Call HOOK27 at location F1B9. At that location, there is only an ordinary RET (return) instruction, so control moves to the next address, 0C49, without doing any operation.

Location F1B9 is part of RAM, and because RAM can be freely modified, by putting this CALLnn instruction at strategic points in ROM one can effectively alter ROM simply by altering the contents of the RAM location called.

(3) 0C49-0C4B LD A,(LEA66) (PORT30)

Transfer the data from I/O port 30H at location EA66 to the A register (accumulator). The six bits from bit 0 to bit 5 at I/O 30H are used.

By setting the bit corresponding to I/O port 30H to 1 or 0, and then outputting it, one can cause the output hardware to take the appropriate action desired by the operator.

(4) 0C4C-0C4D AND 0FH

The logical product of each bit (a 1 results only if both are 1) of the contents of register A and the numerical value 0FH is taken, and the result is stored in register A.

The logical effect of AND 0FH is to set the top four bits to

zero, and the practical effect is to change the data in register A in order to use the cassette interface.

(5) 0C4E-0C4F OR 0CH

The logical sum of each bit (a 1 results if either is 1) of the contents of register A and the numerical value 0CH is taken, and the result is stored in register A.

The logical effect of OR 0CH is to set bit 3 and bit 2 to 1, and the practical effect is to change the contents of register A to output a Motor On and space signal (a signal to indicate the beginning of the tape recording).

(6) 0C50-0C52 CALL LOC38 (SET30H)

Call location 0C38 (SET30H).

(7) 0C38-0C3A LD (LEA66),A
 LD (PORT30),A

Store the contents of register A in location EA66. By writing the contents of register A in location EA66, the current status of I/O port 30H can be obtained by other subroutines in ROM as well.

(8) 0C3B-0C3C OUT (30H),A

Output the contents of register A to I/O port 30H. Because the contents of register A have been set to 000011xx by steps four and five, the cassette interface has been selected, the motor relay has been switched on, and the integrated circuit incorporated in I/O port 30H to output the space feeder has been activated.

(9) 0C3D RET

The RET (return) instruction causes a return to location 0C53.

(10) 0C53-0C55 CALL L0D14 (RE8251)

Calls location 0D14 (RE8251). The effect of this is to initialize the LSI chip called 8251 used in the cassette interface.

(11) 0D14-0D15 LD A,0EH

The 8251 is a communications interface which changes parallel data in the CPU to serial data, and serial data to parallel data. When using the 8251, the 8251 must be reset in order to activate it, and the mode word and command word must then be written into the control word register of the 8251. Whenever the 8251 is to be reset by a program, because it is not known whether the 8251 control word input should be treated as setting the mode or the command, dummy data is sent at least once through the control word register, and the 8251 is reset during the next set-command cycle. Bit 6 must be zero in this dummy data.

In this step, the dummy data 0EH is stored in register A through the LD A,0EH command.

(12) 0D16-0D17 OUT (21H),A

The contents of register A (dummy data) are output into I/O port 21H (the 8251's control word register).

(13) 0D18-0D19 LD A,40H

The contents of register A are set to 40H. The value 40H is the reset value for the 8251.

(14) 0D1A-0D1B OUT (21H),A

Output the contents of register A (the reset value 40H) to I/O port 21H, and reset the 8251.

(15) 0D1C RET

The RET (return) instruction causes a return to location 0C56.

(16) 0C56-0C57 LD A,L00CE (0CEH)

The contents of register A are set to the numerical value 0CEH. As mentioned above, because the information read into the control word register immediately after the 8251 is reset is treated as a mode word, the data to set the mode must first be stored in a register.

The meaning of the data 0CEH (11001110B) to set the mode is, from the highest bits, 11 (two stop bits) 00 (no parity) 11 (eight data bits) 10 (asynchronous mode). The LSI chip 8251 has many communication functions, but in this case, data is transferred with a method of communicating called asynchronous communication.

This value 0CEH signifies that two bits are used as stop bits, a parity bit is not used, eight bits are sent as serial data, and it is asynchronous communication.

(17) 0C58-0C59 OUT (21H),A

Output the contents of register A (the value to set the mode) to the control word register of the 8251 (I/O port 21H).

(18) 0C5A-0C5B LD A,11H

The contents of register A are set to the numerical value 11H. The value to set the command word is 11H. 11H means that bit 4 and bit 0 are 1. Bit 4 indicates error reset for the 8251, and bit 0 indicates the transmission capability status.

(19) 0C5C-0C5D OUT (21H),A

Output the contents of register A (the value to set the command) to the control word register of the 8251 (I/O port 21H).

(20) 0C5E-0C60 CALL L0C7C (TMLOP2)

Calls location 0C7C (TMLOP2). TMLOP2 is a time delay subroutine, and calling this routine records the space feeder time period indicating the beginning of the recording in the approximately three seconds before return is made.

(21) 0C7C PUSH DE

Transfer the contents of the DE registers (the 16-bit data value obtained by combining the contents of register D and register E) to the stack. This is done in order to preserve the contents of the DE registers on the stack, since register D and register E are used in the time delay subroutine.

(22) 0C7D PUSH HL

Transfer the contents of the HL registers to the stack. This

step, like step 21, preserves the register contents so they will not be lost.

(23) 0C7E-0C7F LD E,6

Set the contents of register E to 6. The amount of time spent in the time delay is determined by the contents of register E.

(24) 0C80-0C81 JR L0C6D (TMLOOP)

Jump (relative jump) from location 0C82 to the place return was made before byte 0EBH. In other words, this is a jump to location 0C6D (TMLOOP).

(25) 0C6D-0C6F LD HL,0

Set the contents of the HL registers to 0000.

(26) 0C70 DEC L

Subtract 1 from the contents of register L. When 1 is subtracted from a one-byte value of 0, the result is 0FFH (representation of 255 in decimal).

(27) 0C71-0C72 JR NZ,L0C70 (TMLOP1)

Jump to location 0C70 (TMLOP1) when the contents of register L are not zero.

Return is again made to TMLOP1, and through DEC L the contents of register L are made one less. In this way, (26) and (27) become a loop. In other words, the loop will be exited only when register L has been decremented 256 times.

(28) 0C73 DEC H

Subtract 1 from the contents of register H.

(29) 0C74-0C75 JR NZ,L0C70 (TMLOP1)

A loop is made, similar to (27). However, because loop (28), (29) is outside loop (26), (27), the loops are nested. When register H is decremented 256 times, loop (29) is exited.

(30) 0C76 DEC E

Subtract 1 from the contents of register E.

(31) 0C77-0C78 JR NZ,L0C70 (TMLOP1)

This is similar to (27) and (29). Here, the three levels of loops—the L-register loop, the H-register loop, and the E-register loop—are executed to create a time delay.

(32) 0C79 POP HL

Restore the contents of the HL registers that were preserved on the stack.

(33) 0C7A POP DE

Restore the contents of the DE registers that were preserved on the stack.

(34) 0C7B RET

Return to location 0C61. The time delay subroutine ends with this instruction.

(35) 0C61-0C63 LD A,(LEA66) (PORT30H)

Finally, the contents of location EA66 (PORT30H) in which the data output to I/O port 30H was stored are stored in register A.

(36) 0C64-0C65 AND 0FBH

The logical product of the contents of register A and the numerical value 0FBH is taken. The effect of this is to set bit 2 to 0 no matter what the prior contents of register A may have been. Because bit 2 of I/O port 30H is used to switch between the space signal and the mark signal, this means that the data to change the output from the space signal to the mark signal is now being prepared.

(37) 0C66-0C68 CALL L0C38 (SET30H)

In (36) the data to output the mark signal was stored in register A, and by calling subroutine SET30H this data is output to I/O port 30H and also stored in location EA66.

(38) 0C69 PUSH DE

Transfer the contents of the DE registers to the stack.

(39) 0C6A PUSH HL

Transfer the contents of the HL registers to the stack.

(40) 0C6B-0C6C LD E,1

Store 1 in register E. This is the initialization for the time delay, and exiting from the loop will require about 0.5 seconds, thereby outputting the mark signal for 0.5 seconds.

(6) Defendants Shuuwa System Trading KK (hereinafter referred to as "defendant Shuuwa") and Tokyo Sugaki Printing KK (hereinafter referred to as "defendant Sugaki") are companies engaged in, respectively, the business of marketing published materials relating to computers and the business of printing published materials.

Defendant Shuuwa, about June, 1982, converted the object program at issue, which had been incorporated into the personal computer at issue, directly into hexadecimal code, reproducing the entire object program at issue, after which it disassembled and interpreted the object program at issue, reproducing without permission the entire work at issue. The former act of reproduction was, therefore, the making of a new reproduction from a reproduction of the work at issue, and the latter act of reproduction was the making of a reproduction of the work at issue itself. Defendant Sugaki printed and reproduced, and defendant Shuuwa distributed, both of the above-mentioned reproductions in the book described in the Appendix.

(7) Accordingly, the plaintiff asks that the defendants be prohibited from publishing and distributing the book at issue under Article 112(1) of the Copyright Law, and seeks the destruction of the book at issue under section (2) of that Article.

2. Response to Grounds for Claims

(1) The truth of Ground for Claim 1 is admitted.

(2) As for the truth of Ground 2, it is admitted that "Copyright 1979 (c) by Microsoft" appears on the display during program execution, but we have no knowledge as to the truth of the remainder.

(3) We have no knowledge as to the truth of Ground 3.

(4) We admit the truth of Ground 4.

(5) We deny the truth of Ground 5.

According to Article 2 of the Copyright Law, a copyrightable work is defined as "a production in which thoughts or sentiments are expressed in a creative way and which falls within the literary, scientific, artistic or musical domain." According to the copyright law, the reason for protecting works is to protect personal profit.

However, the work at issue here is an operating system (basic software), and operating systems have the purpose of managing data more efficiently and quickly and can therefore be regarded as scientific theories.¹ Thoughts and sentiments are completely excluded; pure logic is sought after. In this regard, application programs and game programs, which are expressions of thoughts and sentiments in line with the creator's objective, are completely different.

Because it naturally follows that the work at issue is something to which the laws of science and technology apply and in which the element of personal rights can be ignored, it is not protected under copyright law.

(6) As for the truth of Ground 6, we admit the business of the defendants, that defendant Shuuwa converted the object program at issue directly into hexadecimal code, disassembled and interpreted the object program at issue, and published and distributed the book at issue, and that defendant Sugaki printed the book at issue, and deny the truth of the remainder.

The defendant Shuuwa's acts of interpreting the disassembled listing, attaching labels and comments, and giving a description of each column of the source listing in the book at issue constitute presenting the results of its study of the operating system of the personal computer at issue, and those acts should be regarded as

1. Translator's Note: The word "scientific" used here is a different Japanese word (*kagaku*) than is used in the list "literary, scientific, artistic, or musical" of Article 2 of the Copyright Act. The Japanese word *gakujutsu* is translated in the latter as "scientific," although a better translation in this case is "scholarly" or "academic." The defendant probably intended to contrast the purely scientific nature of the work at issue with the scholarly nature of normal copyrightable works. Since the word was translated as "scientific" in the quasi-official English translation of the Copyright Act, that pattern has also been followed here.

independent creative activity, and not the reproducing of the work at issue.

Over 200,000 units of the personal computer at issue have been sold, and it is necessary that users understand the structure of the computer and, more particularly, understand the substance of its operating system, which is the work at issue, in order to effectively use the personal computer at issue, but since neither the plaintiff nor NEC have made this information public and users have only been able to make partial analyses of it, defendant Shuuwa presented the book at issue in the stead of the plaintiff and NEC as a general explanation of the operating system for the users of the personal computer at issue, and the acts of the defendants were right and proper.

III) EVIDENCE (Omitted)

REASONING

1. There is no dispute between the parties about the truth of Ground for Claim 1.
2. The process of creating the work at issue, the rights attached, etc., will now be examined.

If the [omitted evidence] is generalized, the following facts are evident, with no contrary evidence.

The work at issue was developed and produced by the plaintiff at the request of NEC. Supervision of its development was primarily carried out by the plaintiff's president, William H. Gates, and other engineers of the plaintiff, and about eight months were required for completion. Subsequently, the work at issue was delivered to NEC on about August 3, 1979, and NEC reproduced it in ROM and marketed it as incorporated into the personal computer at issue, although it has never been publicly marketed or distributed in the United States. Furthermore, an indication to the effect that the copyright owner of the work at issue is the plaintiff was programmed to appear on the display during execution of the personal computer at issue.

We make the above findings of fact.

In light of the above facts, the work at issue (whether or not it is protected by copyright law will be discussed later) was created on the plaintiff's initiative by persons engaged in the plaintiff's business and as part of their work and can be said to have been made public under the name of the plaintiff, and therefore the copyright should be said to belong to the plaintiff.

Furthermore, the work at issue has never been distributed in the United States, but was reproduced and distributed within Japan through NEC, and because we can say that this is a work to which Japan has the obligation to grant protection under the Universal

Copyright Convention, it should be found to have copyright protection under Article 6(2) and (3) of the Copyright Law.

3. Next we will look at the contents of the work at issue.

From the [omitted evidence], the following facts are found regarding the contents of the work at issue.

(1) The work at issue is the source program for the BASIC interpreter used on the NEC PC-8001, and is written in symbolic language (assembly language). The actual form of expression of the work at issue is as described in the B column of pages 1 to 250 of the program listing which was created by disassembling (which means converting the object program back into an assembly language program, which is a source program), using a disassembling program, the object program which is recorded in the ROM of the personal computer at issue.

(2) Moreover, the purpose of this program is to interpret any commands or programs in the BASIC language that are input into the personal computer at issue, which is a small-scale computer system made by NEC using the Z-80 microprocessor made by the American company Zilog, and to convert them into written instructions and other information to operate the microprocessor.

In other words, when an application program written in the BASIC language is input into the personal computer, we can say that the work at issue is the means to interpret and execute the program.

(3) In order to execute a program, the following kinds of operational steps are necessary, and the work at issue was created so that it could execute these steps without error.

(a) Begin operation when power is turned on.

(b) After beginning operation, initialize the program (initialize each of the various input and output devices, clear the screen, display the sign-on message, set up the various temporary storage areas necessary during execution, etc.).

(c) Whenever a command can be received, display an "OK" message to give notice of that fact.

(d) If an input line begins with numerals, store the line in memory in a condensed form called intermediate code (indirect mode). When the input line does not begin with numerals, interpretation of that line begins immediately, and the appropriate management modules for the commands and execution instructions are given control (direct mode).

(e) The necessary operations are taken by the appropriate execution module for each of the commands and instructions for execution.

(4) Moreover, the work at issue has a total of 229 different routines (for example, a routine to record on cassette tape a pro-

gram recorded in memory and a routine to input a one-line statement) as its structural elements, and it is designed so the proper routine is chosen and used.

4. Accordingly, we will now consider whether or not the work at issue having the contents described in the preceding section is protected under the Copyright Law.

(1) As was found above, the work at issue is the source program for the BASIC interpreter used on the NEC PC-8001, it is expressed in symbolic language (assembly language), and it has the purpose of interpreting any commands or programs in the BASIC language that are input into the personal computer at issue, which is a small-scale computer system made by NEC using the Z-80 microprocessor made by the American company Zilog, and converting them into written instructions and other information to operate the microprocessor.

Moreover, the creator uses his originality and ingenuity with regard to the overall structure of the program, the structure of each routine, and the combining of the commands and other information at each address, in order to match the hardware architecture of the personal computer at issue and to fully utilize its capabilities.

(2) If these points are carefully considered according to all the above evidence, the commands or statements described in each address from address 1EC0 to 0C66 of the work at issue can be understood to have the meaning described in Ground for Claim 5(3) and are understood to be a set of commands to record on cassette tape a program recorded in the computer's internal memory.

Furthermore, the meaning of the commands or statements described in each address from address 3CEC to 3D11 of the work at issue is essentially as follows.

(The address number, assembly language statement, and its substance are described in that order.)

(1) 3CEC CALL L1B7E (INPUTL)

Calls the subroutine named INPUTL which begins at location 1B7E. The INPUTL subroutine performs the action of reading a one-line BASIC program from the console. When the RETURN key is pressed, it recognizes that this is one line and records the string of characters in the area from location EC96, which is labeled as INBUFF, to location ED93.

(2) 3CEF JP C,L3C9F (COMIN2)

If the carry flag is set (C=1), jump (conditional jump instruction) to location 3C9F, which is labeled COMIN2. If the ESCAPE key has been pressed, the carry flag has been set, but because at that time any input is invalid, the contents of the input line are not examined, and the state of waiting for initial input is entered.

(3) 3CF2 RST 10H

This is the same as CALL 0010H. Because the RST command is a one-byte command and the CALL command is a three-byte command, two bytes of memory are saved. At location 10, a jump is made to location 4259, which is labeled CHRG1. The operations carried out at that location consist of putting the ASCII code for the first character of a valid input character string into register A, and setting the carry flag if the initial character is a numeral and resetting the flag if it is not.

```
(4) 3CF3-3CF4      INC   A
                        DEC   A
```

The INC A command adds one to register A and the DEC A command subtracts one from register A. This operation has no effect on the carry flag and is used to check whether or not the contents of register A are zero.

```
(5) 3CF5           JP    Z,L3C9F      (COMIN2)
```

If the Z flag is set (the contents of register A are zero), jump to the address labeled COMIN2 (location 3C9F) and wait for initial input.

```
(6) 3CF8           PUSH  AF
```

The contents of register A and register F are saved on the stack. The information as to whether or not there is a line number (indirect mode or direct mode) is in the carry flag of register F, and it is necessary to preserve the contents of register F.

```
(7) 3CF9           CALL  L44B5      (GETLNO)
```

Call the subroutine at the address labeled GETLNO (location 44B5).

The GETLNO subroutine performs the action of getting the line number from the input line. The line number is converted into a numerical value and stored in the DE register pair, and return is then made.

```
(8) 3CFC           CALL  L4082      (SKPSPC)
```

Call the subroutine at the address labeled SKPSPC (location 4082). The SKPSPC subroutine performs the action of restoring the space character code read over by GETLNO in order to detect the end of the line number.

```
(9) 3CFF           LD    A,(HL)
```

One character from the input line in the INBUFF indicated by the HL register pair is entered in register A.

```
(10) 3D00          CP    20H
```

The contents of register A are compared with the space character (20H). If the contents of register A are a space character, the zero flag is set, and if not, the zero flag is reset.

```
(11) 3D02          CALL  Z,L26C7      (INCHL)
```

If the zero flag is set, the value of the HL register pair is increased by one. When a BASIC LIST command (a command to

display on the screen a program written in the BASIC language) is executed, there is at least one space character immediately following the line number, and since it is not necessary to include this inevitably present space character in the intermediate language, one byte is saved by increasing by one the value of the HL register pair.

(12) 3D05 PUSH DE

The contents of the DE register pair (the line number) are saved on the stack. This is to prevent the contents of the DE register pair from being destroyed by the next instruction.

(13) 3D06 CALL L3E65 (CHGINT)

Call the subroutine at the address labeled CHGINT (location 3E65). This subroutine converts into intermediate language the input that is recorded in INBUFF from the character pointed to by the HL register pair to the end of the line, and records it in the 315-byte area from location EB57, labeled TEXTBUF, to location EC91. When the conversion is complete, EB56, which is one byte before the beginning address (location EB57) of TEXTBUF, is put in the HL register pair and return is made.

(14) 3D09 POP DE

Restore the values of the AF register pair (the numerical value of the line number) from the stack.

(15) 3D0A POP AF

Restore the values of the AF register pair (the information as to whether the line being operated on is in indirect mode or direct mode is in the carry flag of register F) from the stack.

(16) 3D0B LD (LEF8D),HL
TD (SAVETP),HL

Transfer the value of the HL register pair (the address of one byte before TEXTBUF) to the address labeled SAVETP (location EF8D) and save the value of the text pointer, the HL register pair.

(17) 3D0E CALL LF174 (HOOK4)

Call the subroutine at the address labeled HOOK4 (location F174). Because a RETURN command is at this location, return is made without any action being taken.

(18) 3D11 JP NC,L423B (PRACT3)

If the carry flag has been reset (0), go to the address labeled PRACT3 (location 423B). In that case, because the mode is direct with no line number, the input command is immediately executed by PRACT3.

If the carry flag is set (1), the mode is indirect with a line number, so go to the next editor section (here the line number and intermediate code are stored in the BASIC program area) without executing the input data.

As discussed above, the set of instructions written in the addresses from address 3CEC to 3D11 can be seen to be routines to

determine whether the mode for the commands and statements input from the keyboard or elsewhere is indirect or direct, and to then go on to the next step.

(3) According to the facts found above, the work at issue was created, even down to the program structure, routines, and use and combining of the subroutines, using a high degree of technical knowledge concerning programming languages so that commands and programs input into the personal computer at issue through the BASIC language are executed as written, and the result desired by the person inputting the command is produced, and it is clear that the work is the expression of the scientific thoughts of the program creator and therefore can be regarded as a work within the scientific domain.²

(4) However, the defendants claim that the work at issue is an operating system (basic software) and thus is different from application programs and the like because it does not include the author's thoughts and sentiments and its only purpose is to manage data efficiently and quickly, and therefore it does not receive copyright protection. We will examine this point.

Generally with regard to computer programs there is not just one solution to achieve a particular purpose, but it is possible to choose from a variety of solutions. Similarly, when the objective is to produce a BASIC interpreter to put into a personal computer, as in this case, the various problems in achieving that purpose must be individually and carefully analyzed and a solution found for each, as was previously examined in detail, and the program is completed by writing these solutions in assembly language as a combination of commands and other information. All of these processes are unique, and not only does each process differ in that it reflects the creator's personality and thoughts, but there is also value in that individuality. There is not the least bit of difference regarding this point whether the program is a game program, an application program, or as in this case an operating system.

Therefore, there is no validity to the defendants' claim that the work at issue is not protected under copyright law. 5. We will next examine whether the defendants committed copyright infringement.

(1) There is no dispute between the parties that defendant Shuuwa converted directly into hexadecimal code the object program at issue that was incorporated into the personal computer at issue, and then after disassembling the object program at issue, interpreted the program and attached labels and comments.

It is clear that the former of these acts of the defendant

2. Translator's Note: The Japanese word translated in this sentence as "scientific" and "science" is *gakujutsu*, which is probably better translated as "scholarly." See *supra* note 1.

Shuuwa can be considered to be the reproduction of the object program at issue, which is itself a reproduction of the work at issue.

We will now consider the latter of these acts of the defendant Shuuwa.

According to [omitted evidence], we find the following facts.

The work at issue (the B-column of each page of Exhibit A-1) and the *ro*-column of each page of the book at issue have external differences.

For example, if we look at address 0004, JP L003B is written in the work at issue, while JP WAMCHK;CHECK COLD START OR WARM START is written in the *ro*-column of the book at issue, and they differ in that regard. However, while the former means an unconditional jump to location 3B (the WARM CHECK subroutine begins at location 003B), and the latter means an unconditional jump to the address labeled WAMCHK, the latter is just an explanation in English of the instruction's meaning and an English abbreviation label indicating the function performed at the place the jump is made to, and the former and the latter both have the same meaning.

The other differences in expression between the two texts largely arise from the difference in explanatory labels.

We find the above facts.

From this, because we can say that both versions are the same program, the acts of the defendant Shuuwa in disassembling and then interpreting the object program at issue and attaching labels and comments to the interpreted version can be considered to be the act of reproducing the work at issue.

Moreover, there is no dispute that defendant Sugaki reproduced and printed the book at issue, and that the defendant Shuuwa distributed it.

(2) The defendants claim that the defendant Shuuwa's actions in interpreting the disassembled listing, attaching labels and comments, and giving descriptions of the items in the source list column of the book at issue should be viewed as the presenting of the fruits of its research and therefore as independent creative activity, and that these actions were thus not copyright infringement. Moreover, they argue that because the defendants' presenting of the book at issue was for the benefit of the users of the personal computer at issue, the defendants' actions were fair and just.

However, whether or not something newly created based on a preexisting work possesses creativity or individuality should have no bearing on whether or not the copyright in the preexisting work has been infringed. Moreover, it seems natural that the act of presenting the work at issue, which was not made public by its author as a source program, against the wishes of the author cannot be

justified simply because it was done for the convenience of users, and the defendants' argument is therefore inapplicable and cannot be adopted.

6. As discussed above, each of the above-mentioned acts of the defendants should be considered to be infringements of the copyright at issue belonging to the plaintiff, and because we can say that the book at issue was created by the above-mentioned acts of infringement, the plaintiff can properly demand that the defendants cease publishing and distributing the book at issue and destroy the book at issue.

Therefore, each of the plaintiff's claims against the defendants is granted, and with regard to the litigation costs, we rule as was stated in the Order based on the various provisions of Articles 89 and 93(1) of the Code of Civil Procedure.

Tokyo District Court, Twenty-ninth Civil Division

Chief Judge
Judge
Judge

Shin Motoki
Toshiaki Imura
Eiji Tomioka

APPENDIX: THE WORK AT ISSUE

Source Program of the Basic Interpreter used on the NEC PC-8001

The listing (page 1) made by disassembling the program recorded in the NEC PC-8001 personal computer made by Nihon Electric Company is as shown in Exhibit 1, and the work at issue is shown in the B-column of that listing.

THE SUBJECT OF SUIT

Title: PC-8001 BASIC SOURCE PROGRAM LISTINGS: THE WHOLE ANALYSIS OF Ver. 1.0 and 1.1 (Publisher: Shuuwa System Trading K.K.)

The first page of the program listing of the book at issue is as shown in Exhibit 2.

EXHIBIT 1

		ORG		O
0000	F3	L0000:	DI	
0001	31FFFF		LD	SP,LFFFF
0004	C33800		JP	L003B
0007	00		NOP	
0008	C3336A00	JP	L006A	
000B	C35717	L000B:	JP	L1757
000E	AB		XOR	E
000F	FO		RET	P
0010	C35942		JP	L4259
0013	C36A00	L0013:	JP	L006A
0016	DA0CC3		JP	C,LC30C
0019	A6		AND	(HL)
001A	40		LD	B,B
001B	F3		DI	
001C	0B		DEC	BC
001D	C37E50		JP	L507E
0020	C3DAF1		JP	LF1DA
0023	C39C27		JP	L279C
0026	88		ADC	A,B
0027	0C		INC	C
0028	C3DDF1		JP	LF1DD
002B	C3600D		JP	L0D60
0021E	46		LD	B,(HL)
002F	0C		INC	C
0030	C3E0F1		JP	LF1EO
0033	9F		SBC	A,A
0034	0F		RRCA	
0035	C35702		JP	L0257
0038	C3E3F1		JP	LF1E3
003B	AF		L003B:	XOR,A
003C	3275EA		LD	(LEA75),A
003F	CDF10C		CALL	L0CF1
0042	30C7		JR	NC,L000B
0044	18CD		JR	L0013
0046	00		NOP	
0047	00		NOP	

0048	00	NOP
0049	00	NOP
004A	00	NOP
004B	00	NOP
004C	00	NOP
004D	00	NOP
004E	00	NOP
004F	00	NOP
0050	00	NOP
0051	00	NOP
0052	00	NOP
	A	S

EXHIBIT 2

NPC-8001 Basic Source Program Listings

		ORG		O	
N0000	F3		DI		
0001	31FFFF		LD	SP,0FFFFH	
0004	C33B00		JP	WAMCHK	; CHECK COLD START OR WARM START
0007	00		DB	0	
0008	C36A00		JP	BASHOT	; RESTART 08H TO BSC WARM START
000B	C35717	RECOLD:	JP	COLDST	; BSC COLD START
000E	ABFO		DW	FACC+3	
0010	C35942		JP	CHRG EI	; RESTART 10H TO GET CHARACTER
0013	C36A00	REWARM:	JP	BASHOT	; BSC WARM START
0016	DA0C		DW	CCHROT	
0018	C3A640	JP	CHROUT		; RESTART 18H TO OUTPUT CHAR.
001B	F30B		DW	CHNIT	
001D	C37E50		JP	POS2	
0020	C3DAF1		JP	HOOK38	; RESTART 20H FOR USER
0023	C39C27		JP	CINT3	; REGISTER HL TO FLTNG. ACCUM.
0026	880C		DW	CCHRIN	

0028	C3DDF1	;	JP	HOOK39	; RESTART 28H FOR USER
002B	C3600D	;	JP	LPTCHR	; OUTPUT CHAR. TO PRINTER
002E	460C	;	DW	COINIT	
0030	C3E0F1	;	JP	HOOK40	; RESTART 30H FOR USER
0033	9F0F	;	DW	KSCAN0	
0035	C35702	;	JP	CONOUT	; DISPLAY CHARACTER
0038	C3E3F1	;	JP	HOOK41	; RESTART 38H FOR USER
		;			

APPENDIX 2

System Science K.K. v. Toyo Sokuki K.K.

June 20, 1989, Tokyo High Court, 1322 Hanrei Jihou 138

Case No. Heisei 1 (ra)-327 - A *kokoku* appeal from denial of a petition for a provisional injunction to cease infringement of copyright.

Original decision - Tokyo District Court Case Nos. Showa 63 (yo)-2531 and Showa 63 (yo)-2551

ORDER

I) The original decision is modified as follows:

1. The Appellees may not reproduce or adapt the programs listed in sections 1, 2, and 4 of Appendix III.

2. Appellee Toyo Sokuki K.K. may not distribute the devices listed in sections 1-4 and 9 of Appendix II that incorporate the programs listed in sections 1, 2, and 4 of Appendix III, nor may it advertise or exhibit them for the purpose of distributing them.

3. Appellee K.K. Nihon Technart may not distribute the devices listed in sections 1-4 and 9 of Appendix II that incorporate the programs listed in sections 1, 2, and 4 of Appendix III.

4. The Appellant's other requests are denied.

II) The costs of the petition and of the appeal shall be divided in half, with the Appellant bearing one half and the Appellees bearing the remainder.

FACTS

I) GIST OF THE APPEAL

1. The original decision should be stricken.

2. Same as part I, paragraphs 1 to 3 of the Order.

3. Appellees may not reproduce or adapt the programs listed in section 3 of Appendix III.

4. Appellee Toyo Sokuki K.K. may not distribute the devices listed in sections 5-8 of Appendix II that incorporate the programs listed in section 3 of Appendix III and Appendix IV, nor may it advertise or exhibit them for the purpose of distributing them.

5. Appellee K.K. Nihon Technart may not distribute the devices listed in sections 5-8 of Appendix II that incorporate the programs listed in section 3 of Appendix III and Appendix IV.

6. The costs of the petition and of the appeal shall be borne by the Appellees.

II) ALLEGATIONS OF THE PARTIES

1. In addition to the supplement that follows, we adopt the

description of page 4, line 4 through page 17, line 7 of the original decision. (However, "Petitioner" is changed to read "Appellant" and "Respondents" to read "Appellees." Moreover, "*gijutsu shiryō*" on page 14, line 3 of the original decision is changed to "*gijutsu shiryō*."¹)

2. Claims of the Appellant

That program CA-9 is an adaptation of program CA-7 II is clear from the following facts:

1. The expression of a program is a combination of instructions, and so even if each of the control codes, output addresses, and output instructions, for example, were constrained by the hardware and the flow of processing were identical, it would not be essential that the combination of instructions be identical.

Moreover, although program CA-9 is an extraction of the basic functions of program CA-7 II, even a program with basic functions can embody creativity. In other words, a program's creativity is determined by the degree of freedom of expression available for its combination of instructions, but because the degree of combinatorial freedom has no relation to whether the functions of the program are basic or specialized, it cannot be said that program CA-9 corresponds to the portions of program CA-7 II without embodying creativity simply because they are programs carrying out basic functions.

2. Concerning the "processing routine for after data entry from the processor":

If we assume that the degree of creativity in a program expressed as "a simple manifestation of individuality" should be sufficient, then many efficient ways of expressing a "processing routine for after data entry from the processor" are possible, and because this same routine in program CA-7 II was made with redundancies in order that changes could later be easily made, it must be said that individuality is sufficiently manifest and there is creativity.

Although the original decision found that from the perspective of trying to control the length of the program, no matter who created it the subroutine in the "processing routine for after data entry from the processor" would be identical, it cannot be said that getting results with maximum speed by having the shortest program is necessarily the most practical program. Whether a program is practical is determined by considering its efficiency and economy in view of the particular development environment - such as development time, memory requirements, the demand for processing time, and ease of maintenance - and therefore, individuality can be mani-

1. Translator's Note: The term "*gijutsu shiryō*" means "technical materials." The original decision contained an apparent typographical error, using an incorrect *kanji* character for "shī" in the word "*shiryō*."

fest in the combination of instructions, and because program CA-7 II has a great deal of leeway in memory size and processing time and it is not necessary to control the length of the program, it is absolutely not necessary for it to be identical no matter who creates it.

3. Concerning "processing routine when printer operation is disabled":

A "processing routine when printer operation is disabled" is given as the Example 3 in Exhibit B-42, but each one of these combinations of instructions is completely different from the combinations of instructions of program CA-7 II. Accordingly, because it is clear that the combination of instructions can easily not be identical even if the algorithm is identical, program CA-7 II must be said to embody creativity even in the "processing routine when printer operation is disabled."

REASONING

D) We find from Exhibits A-7, A-23, A-26, and A-71 (all of which are reports from the Appellant's representative) and Exhibit A-55 (report of Makoto Karasawa), which provide evidentiary support for what was presented in the oral proceedings, that at the initiative of the Appellant, Representative Director Kiyoshi Sugano and Director Makoto Karasawa, who were engaged in the business of the Appellant, created in the course of their duties the program described in section 4 of Appendix III (the MIC program) by about March of 1981, the program described in section 1 or Appendix III (the temporary edition program ZA-FM II) and the program described in section 2 of Appendix III (the temporary edition program ZA-FX II) by about September of 1985, and the program described in section 3 of Appendix III (the program CA-7 II) by about March of 1986, and in addition, that the Appellant was selling the device described in section 4 of Appendix I that has ROM installed on its circuit boards incorporating a copy of the MIC program in about December of 1981 and the device described in sections 1 and 2 of Appendix I that has ROM installed on its circuit boards incorporating a copy of the temporary edition program ZA-FM II and the temporary edition program ZA-FX II in about February of 1986, each program being made public under their own name as being of their authorship. It is then clear that the copyright in the programs described in sections 1 to 4 of Appendix III belongs to the Appellant.

With regard to this point, the Appellees argue that because the programs described in sections 1 to 4 of Appendix III were created in the course of their duties by the Appellant's engineers in the course of the business of Appellee Toyo Sokuki K.K. and at the

initiative of Appellee Toyo Sokuki K.K., their copyright belongs to Appellee Toyo Sokuki K.K., and that even were that not found to be true, because the devices described in sections 1 to 4 of Appendix I were developed with Appellee Toyo Sokuki K.K. bearing the expense and with the Appellant doing nothing more than supervising their production based on the directions of Appellee Toyo Sokuki K.K., and because the programs described in sections 1 to 4 of Appendix III were created exclusively for the devices described in sections 1 to 4 of Appendix I, Appellee Toyo Sokuki K.K. should be said to have been conveyed the copyright to the programs described in sections 1 to 4 of Appendix III through the conveyance from the Appellant of the devices described in sections 1 to 4 of Appendix I. However, because we are not convinced even from Exhibit B-7 (a document entitled "Request for Manufacturing Study") and Exhibits B-32(1) and B-32(2) (a document entitled "Development Material Provided to SSO from Toyo Sokuki K.K. at the Time of the Request for Study of Zone Analyzer ZA-FX" and its contents) that the Appellant's Representative Director Kiyoshi Sugano and Director Makoto Karasawa created each of the above-mentioned programs as the business of Appellee Toyo Sokuki K.K. or that an agreement had been made, even impliedly, between the Appellant and Appellee Toyo Sokuki K.K. to convey the copyright in the programs described in sections 1 to 4 of Appendix III, we cannot accept the above argument of the Appellees.

II) In the past (according to p.12 of Appellee Toyo Sokuki K.K.'s Brief), the Appellees distributed each of the devices described in Appendix I, which have ROM installed on their circuit boards incorporating a copy of each of the programs described in Appendix III, and the Appellees themselves recognize and do not dispute that advertising and exhibiting for the purposes of distribution were carried out. That alone makes it difficult to deny that there is a danger that the Appellees would in the future reproduce or adapt any or all of the programs described in sections 1, 2, and 4 of Appendix III, and a danger that they would distribute, or advertise or exhibit for the purpose of distributing, the devices described in sections 1 to 4 and 9 of Appendix II incorporating them.

With regard to this point, the Appellees argue that reproduction of the programs described in Appendix III had already ceased and there were plans to change to new programs, and Appellees have submitted a development commission agreement (Exhibit B-37) dated December 20, 1988, indicating that a development agreement was entered into by the Appellees for programs to replace the temporary edition program ZA-FM II and the temporary edition program ZA-FX II. However, as it has not been demonstrated at all that a program which can be used in the devices described in sections 1 to 4 and 9 of Appendix II has been created by now by the

Appellees and that these programs do not infringe the copyrights in the temporary edition program ZA-FM II, the temporary edition program ZA-FX II, or the MIC program, it would be premature to decide that the danger of the Appellees reproducing or adapting any of the programs of sections 1, 2, and 4 of Appendix had been eliminated simply because a program development agreement was entered into between the Appellees.

III) Although there is no dispute between the parties that the Appellees are distributing each of the devices described in sections 5 to 8 of Appendix II that have ROM installed on their circuit boards incorporating a copy of the program described in Appendix IV (the CA-9 program) or that they are advertising or exhibiting for the purposes of distribution, the Appellant argues that program CA-9 is an adaptation of program CA-7 II.

However, in order to be able to decide that a program infringes the copyright in a program work, it is of course necessary that there be a part of the combination of instructions of the program work that can be found to be creative, and that the combination of instructions of the later-created program be similar to the part of the program work which can be found to be creative, but we must say that we are not convinced that the portion of program CA-7 II which the Appellant has indicated can be found to be creative in its method of combining instructions.

In other words, because for any program the symbols by which it is expressed are extremely limited and the relevant system (grammar) is also strict, if one attempts to cause a computer to function and obtain a certain result in a more effective manner, there will be more than a few portions where one cannot avoid the combination of instructions necessarily being similar. Therefore, finding copyright infringement with regard to program works must be done with careful consideration, and according to Exhibits B-39 (report of the representative of Appellee Toyo Sokuki K.K.) and B-41 to B-43 (report of the representative of Appellee K.K. Nihon Technart), which provide evidentiary support for what was presented in the oral proceedings, we find that the functions in the devices described in sections 5 to 8 of Appendix II of changing measuring modes, keyboard input, setting the measurement area, measuring, and writing in common memory, are all accomplished by hardware, and the appropriate operations for programs CA-7 II or CA-9 are only the printer portion (waiting for the measurement data, etc., to be written into common memory, reading it out and converting it to printer code, and outputting it); that because the combination of instructions for the "processing routine for after data entry from the processor" is constrained by hardware, it must be essentially the same combination; that both program CA-7 II and program CA-9 adopt a very common combination of instructions for the "process-

ing routine when printer operation is disabled" (in other words, the processing routine for waiting for the printer); and that because in the devices described in sections 5 to 8 of Appendix II 4000H and above is RAM area, it is common sense to set the subroutine stack at the ideal partition of 4100H. Of course, because the "flow of processing" itself in a program is an "*arugorizumu*" (algorithm), or in other words a "*kaiho*" (solution) as provided for in Article 10(3)(iii) of the Copyright Law, and is a portion which does not receive protection as a copyrightable work, it has no relevance to the creativity of the program.

As stated above, in addition to it being difficult to recognize any creativity in the combination of instructions in the portion identified by the Appellant in program CA-7 II, if we also consider that program CA-7 II is 12 kilobytes compared to 763 bytes for program CA-9, and yet the similar portions of the two programs put forth by the Appellant are no more than a very small number of bytes, we simply cannot be convinced that program CA-9 is an adaptation of program CA-7 II. Exhibits A-81 (report of Makoto Karasawa) and A-82 (report of the representative of the Appellant) cited by the Appellant are insufficient to sway us from that decision.

Moreover, since as discussed above the parties do not dispute that the Appellees are distributing, or advertising or exhibiting for the purposes of distribution, each of the devices described in sections 5 to 8 of Appendix II that have ROM installed on their circuit boards incorporating a copy of the program CA-9, it is appropriate to consider the danger that the Appellees will reproduce or adapt program CA-7 II, and the danger that they will distribute, or advertise or exhibit for the purposes of distribution, the devices described in sections 5 to 8 of Appendix II that incorporate that program, to have been eliminated.

IV) Because of the above, there are grounds for the portion of the Appellant's petition in this case that relate to the programs described in sections 1, 2, and 4 of Appendix III, but there are no grounds for the portions that relate to section 3 of that Appendix. Accordingly, we modify the original decision, which differs in part in its conclusion from the above, as set forth in part 1 of the Order, and with regard to the cost of the petition and of the appeal, applying Articles 96, 89, 92, and 93 of the Code of Civil Procedure, we decide as set forth in the Order.

June 20, 1989

Tokyo High Court, Sixth Civil Division

Chief Judge
Judge
Judge

Toshihiko Fujii
Tamio Kasuga
Yoshihiko Iwata

APPENDIX I

1. Zone Analyzer System ZA-FM II (temporary edition)
2. Zone Analyzer System ZA-FX II (temporary edition)
3. Colony Analyzer System CA-7 II
4. Image Processing Minimum Growth Impeding Density (MIC) Measuring Device

All are manufactured by the Appellant.

APPENDIX II

1. Zone Analyzer System ZA-FM II
2. Zone Analyzer System ZA-FX II
3. Zone Analyzer System ZA-FM III
4. Zone Analyzer System ZA-FX III
5. Colony Analyzer CA-9A
6. Colony Analyzer CA-9M
7. Colony Analyzer CA-9F
8. Colony Analyzer CA-9D
9. Image Processing Minimum Growth Impeding Density (MIC) Measuring Device

All have a "TOYO" nameplate attached.

APPENDIX III

1. Zone Analyzer System ZA-FM II (temporary edition)
Object Program Appendix I
2. Zone Analyzer System ZA-FX II (temporary edition)
Object Program Appendix II
3. Colony Analyzer System CA-7 II
Object Program Appendix III
4. Image Processing Minimum Growth Impeding Density (MIC) Measuring Device

Object Program Appendix IV

Note that items 1 to 4 are references to the object program dump lists appended to the decision of the original court.

Appendix IV

The program work described in the appended object dump list, which the Respondent Toyo Sokuki K.K. claims is currently used in the Colony Analyzer System CA-9.