ISSN 1346-5597

National Institute of Informatics

NII Technical Report Aug 2002

Dynamic Evolution of Product Architecture
—Alternative view of Technical progress of Numerical Controller —

Tomoatsu Shibata, Masaharu Yano

NII-2002-002E
Aug 2002

URL: http://www.nii.ac.jp/ Tokyo 101-8430, JAPAN +81-3-4212-2000 FAX: +81-3-3556-1916

Dynamic Evolution of Product Architecture
¾ Alternative view of Technical progress of Numerical Controller ¾

Tomoatsu SHIBATA [a] and Masaharu YANO[b]

[a] The Sasakawa Peace Foundation

[b] National Institute of Informatics

Abstract:

The viewpoint, the product architecture, has the possibility to provide the industry analysis a new analyzing angle. However as for the most of the previous studies, it adopts the analysis frame to argue about the convenience of the each architecture and the conformability with the product development style and to compare, by thinking that the each product architecture is given. We will do the case analysis concerning the evolution of the architecture of the NC(Numerical Control) system which lasted 35 years from 1962 to 1997 in order to consider the evolution of the product architecture. Thus we led the following hypotheses as for the evolution of the product architecture. First, the product architecture basically evolves from the integral architecture to the module architecture, and then to the direction of the open architecture. Then 2nd, but the change is not always in one direction, and when and the epoch-making element technique such as the microprocessor is born, the technical problem about the design of the architecture occurs and it can be sometimes retraced from the module to the integral.

Keywords: Product Architecture, Technical Progress, NC(Numerical Control), Integral Architecture, Module Architecture

1. Introduction

Product architecture is the design thinking that is behind individual products, a framework, and is therefore a concept that holds a position above specific individual products (Note 1). Classification by product architecture is not necessarily consistent with existing industrial classifications. Within the same automobile industry, for example, it has been said that the sedan has a closed architecture, but the truck has nearly an open architecture (Note 2). In the computer industry, too, even though the personal computer has an open architecture, general-purpose machines can be considered to have a nearly closed architecture. The issue of open architecture versus closed architecture rests on the question of whether or not the interface specifications of the product are public; it does not depend on the differences of specific technologies itself. If we look at products from this point of view, then the trucks of the automobile industry and the personal computer of the computer industry are both classified in the category of open architecture products. In this way, the concept of architecture has the potential to provide a new angle of view from which to analyze industries, but most existing research on product architectures has adopted the comparative analytical framework of taking individual product architectures as givens and discussing their relative advantages and the suitability to the product development style (Note 3). However, another deeply interesting analytical framework concerning product architecture is the evolution of product architecture. This, at the same time, is a matter of reconsidering the evolution of technology from the viewpoint of product architecture.

On the basis of above recognition, this paper analyzes product architecture from an evolutionary viewpoint. A product architecture is not a fixed thing, but rather a dynamically evolving thing, so we consider in what direction that process is moving. Bringing an evolutionary viewpoint into the discussion can at once both deepen and promote the discussion of product architecture. This paper is aiming at the formulation of a hypothesis concerning the evolution of product architecture.

The perspective from which product architecture is viewed in this paper is how is the product system as a whole implemented by what combination of what kinds of subsystems. The product architecture is therefore analyzed first by how the functional elements are mapped to the structural elements in the product system, and second by to what degree the interfaces between the structural element are defined by explicit rules (Ulrich, 1995).

From the two viewpoints described above, architectures can be broadly classified into two broad conceptual types, one in which rules have not established and the mapping relationship and the interfaces are both complex, and one in which rules have been established and the mapping relationship and interfaces are both simple. The former is referred to as an integral architecture, and the latter is referred to as a modular architecture. No clear line has been drawn to distinguish between an integral architecture and a modular architecture; it is ultimately a matter of degree. However, the establishment of these two concepts is needed for the development of the discussion that follows (Note 4).

Table1. Classification of product architectures

	Integral architecture	Modular architecture
Closed	• Complex mapping of functional elements to structural elements • Complex interfaces between structural elements; no standardization • Interface specifications are generally not public.	• Simple mapping of functional elements to structural elements • Simple interfaces between structural elements with rules established for them • Interface specifications are generally not public.
Open	May exist in theory, but do not exist in actual practice.	• Simple mapping of functional elements to structural elements • Simple interfaces between structural elements with rules established for them • Interface specifications are generally public.

Here, we will introduce the dimension of the range of openness, which is the degree to which the mapping relationship and the interface rules are made public. That is to say, are the specifications open only within the company, or are they broadly open to and shared within the industry. We can also call this the closed-open axis. By introducing this new axis, product architectures can be classified into four categories as is shown in Table 1 (Note 5). Conceptually, there are both open and closed integral architectures as well as both open and closed modular architectures. In actuality, however, products for which the mapping relationship is complex and rules have not been established for the interfaces are virtually nonexistent, because they would have little meaning as widely public architectures, even if they were made public. Accordingly, we may consider open architectures to be modular architectures.

We will therefore refer to open modular architectures as open architectures in this paper, and in the following discussion, classify product architectures into the three categories of integral architectures, modular architectures, and open architectures, and apply that classification to the discussion of the evolution of product architectures.

In the following section, we review the relevant preceding research and clarify the position of this paper within that literature and the features of this paper. In section 3, evolutionary process of NC system architecture over the 35-year period from 1962 to 1997 is analyzed, and it is shown that the product architecture changed toward a constant direction before and after the microprocessor (MPU) was introduced. In section 4, building on the results of that analysis, a hypothesis concerning the evolution of product architecture is studied and several implications are derived.

2. Precedent research

The evolution of product architectures is very closely related to the research topics of technological evolution and industrial evolution. We will therefore review here the main existing research on the directionality of technological evolution, and clarify the positioning and features of this paper. The work of Rosenberg and Abernathy can be considered to represent the main existing research.

Rosenberg, on the basis of a detailed analysis of the history of the machine tool industry, has pointed out that the direction of the evolution of technology is determined by technological imbalance and over-shooting (Rosenberg, 1976). When there is a technological imbalance among elemental components that creates a performance bottleneck for the product as a whole, attention is focused on that point of imbalance and technology development efforts are concentrated there. The concentration of research activities leads to achievement of research results that exceed the original targets (overshooting) and improve the overall performance, at the same time creating a new imbalance in technology. In this way, according to Rosenberg, technological development proceeds through the repetition of the cycle in which technological bottlenecks are revealed and eliminated one after another.

Supporting existence of this kind of inherent of technology are the proposition that the evolution of technology follows a fixed trajectory (i.e., the technological trajectory) and the proposition that it is path dependent (Dosi, 1982). Nelson and Winter point out the effect that accumulated technology by a company through past technology development activities has on search activities for future technological opportunities, and explain the path dependence by means of a learning mechanism, which is the accumulation of technology by that company (Nelson and Winter, 1982). Also, Cohen and Levinthal point to the importance of the ability to absorb and make use of outside knowledge, which they refer to as "absorptive capacity", but according to them, that capability is dependent on pre-existent knowledge concerning the field in question. That is to say, the knowledge that is already possessed by the company determines the company's absorptive capacity, and that capacity affects the company's activities for acquiring knowledge from the outside, and as a result, the evolution of technology is said to be path dependent (Cohen and Levinthal, 1983).

The assertion of the series of research that begins with Rosenberg is that there is a characteristic logic in the directionality of technological evolution and that the direction is determined by that logic. As of yet, however, the essential substance of technological evolution has not been dealt with. In this paper, by taking the viewpoint of product architecture, we will approach the substance of evolution and directionality with emphasis on the technological perspective such as architecture.

Abernathy have discussed the mutual interaction between the customer and industrial evolution at the market level, and have proposed the concept of dominant design (Abernathy, 1978; Clark, 1983). Dominant design is the design that achieves dominance in the market. In the product evolution that are created with new technology, technology selection of the product begin from a fluid state and gradually move toward the stage in which selection of technology is fixed in a specific form, then finally the a dominant design for the product is established, and the technological evolution of the product converges. Until the dominant design is established, the company is continuously searching through a trial and error process of combining various types of technology (Note 6). Once the dominant design appears, however, if it is decided that the system of technology will be accepted by the market, the company halts its search activities. Then, the source of competitive advantage shifts to the cost reduction by means of the experience curve, and focus of competition shifts from radical innovation to incremental innovation, and from product innovation to process innovation.

Thus, although the emergence of a dominant design has an important effect on industrial evolution, it is not possible to judge whether or not the product features and functions constitute the dominant design. It is not until one looks back on the history of industrial development that one can tell whether or not a design is the dominant design at a particular time. That is because the concept of dominant design is defined only from the viewpoint of the market; it does not involve a definition from the viewpoint of technology. In this paper, we analyze the product architecture from the two viewpoints of the mapping of functional elements to structural elements, and the interfaces between structural elements, thus making it possible to see the evolution of technology from changes in the product architecture.

Certainly, Rosenberg and Abernathy et al have conducted detailed historical analyses concerning technological evolution in the machine tool industry and the automobile industry, respectively. We believe, however, that the main subject of their analyses is clarification of the logic that determines the direction of the evolution of technology in those industries, rather than substance of technological evolution of itself. As a result, their work does not emphasize the question of what is evolving or the evolution itself and its content. This paper attempts to more closely approach the substance of technological evolution by using an analytical framework that has a more strongly technological coloring, product architecture. The positioning of this paper amidst the existing research on the technological evolution and industrial evolution can be obtained from that point.

One of interesting research on product architecture based on evolutionary viewpoint as is this paper is the discussion of a “dynamic shift in architecture” by Chesbrough and Kusunoki (Chesbrough and Kusunoki, 2001). Chesbrough et al develop a very interesting discussion concerning an inherent mechanism in which the technological innovation in certain components, that is to say modular components, promotes changes in the overall product architecture. They further point to the problem that the organization cannot adapt to such a dynamic shift in the product architecture because of inertia. They refer to this kind of problem that is associated with modularization as the “modularity trap”. This paper takes the same viewpoint as Chesbrough and Kusunoki on the point that a shift in product architecture is being discussed, but could be said to hold the position of extending the viewpoint to a case in a different industry and exploring that shift in more detail and more empirically. On the other hand, whereas Chesbrough et al performed their analysis in the context of the adaptability of an organization to changes in product architecture, this paper attempts to analyze product architecture in the context of the evolution of technology. Even though there is a difference in what is stressed between the two, this paper shares the same analytical framework with Chesbrough and Kusunoki, and supplements and deepens their discussion (Note 7).

3. Analysis of the evolutionary process of NC architecture

In this section, we focus on a single product system, the NC system, and analyze in detail the changes in NC system architecture from 1962 to 1997 to see whether or not a direction of change can be observed (Note 8). Again, we will regard the NC system architecture from the two points of view described above, the mapping from functional elements to structural elements and the interfaces between structural elements.

The NC system was selected as the subject of analysis for two reasons. One is that NC system is a typical system product, and so is believed suitable for the purpose of this paper, which is to examine the process of the evolution of product systems. The other reason is that although NC systems are a technology that was first developed in the US, Japan currently has overwhelming competitive strength in this technology, which means that there is a great deal of technical material available in Japan, so that access to the data is easy. Furthermore, because the Fanuc, Ltd. has been a leader in this industry from a very early stage up to today, it is possible to grasp the overall evolution of technology of NC systems by examining the evolution of technology in the Fanuc NC systems.

The changes in the technological structure of NC systems over the 35 year period can be roughly divided into two eras. The first era, from 1962 to 1969, was characterized by hard-wired technology; the second era, beginning in 1975, was centered around MPU technology. To anticipate the conclusion, a change in the decomposition method in the two directions of simplification of the function-to-structure mapping and standardization and simplification of the interfaces, which is to say a change in direction from an integral architecture to a modular architecture, can be observed in both of these technological structures. The modularization of NC systems was achieved at a very early stage, and in 1969 hard-wired modular NC was already in place. After that, however, adoption of the MPU, an epoch-making elemental technology, forced a restructuring of NC system architecture, and although there was a change from modular architecture to integral architecture immediately after introduction of MPU, NC system architecture gradually changed to modularization. In the following sections, we will look at these changes in detail.

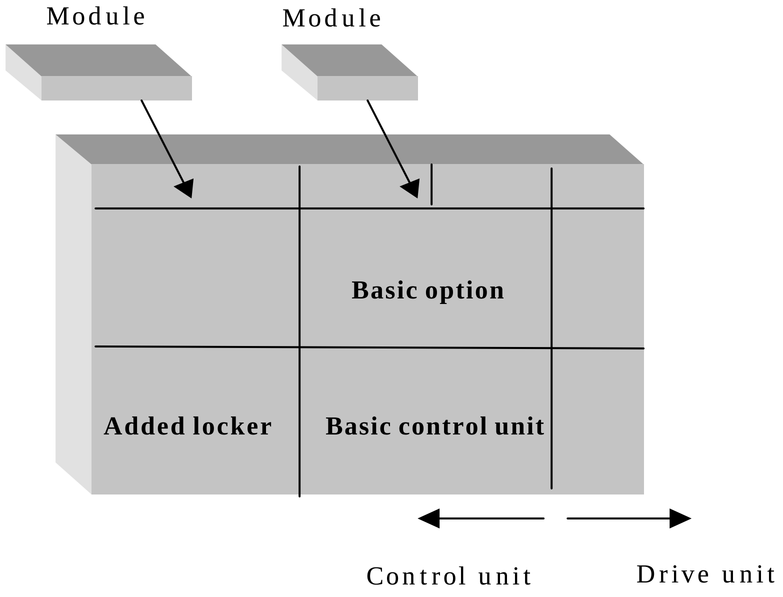
3.1 The appearance of hard-wired module NC

NC technology first appeared in 1952 at the Massachusetts Institute of Technology in application to a numerically controlled milling machine. Research continued after that, and in June 1955 the Cincinnati Machine came out with a numerically controlled milling machine that was called the Hydrotel. At that time, transistors and integrated circuits did not exist, so vacuum tubes were being used as the circuit elements, with the result that the NC equipment alone was larger than the Hydrotel machine (Note 9). In Japan, at an international machine tool exhibition held in Osaka in 1958, Makino Milling Machines and Fujitsu exhibited an NC milling machine. The following year, in 1959, an NC milling machine produced collaboratively by Hitachi Precision Instruments and Fujitsu was delivered to the Nagoya Aircraft Plant of Mitsubishi Heavy Industries.

Then, the Fanuc 220 was completed in 1962. That system employed transistors, thus greatly reducing the number of components used in an NC system. For example, in the vacuum tube era, as many as 2000 vacuum tubes were used, but the use of transistors, diodes, and other such circuit elements resulted in a very small NC unit. Even so, as many as 300 printed circuit boards were used in the NC system (Note 10).

A few years later, integrated circuits (ICs), which contain a circuit that comprises many transistors, resistors, capacitors, etc. in a single package, were being used in NC systems, which reduced the number of boards that were needed to about 40 multilayer printed circuit boards. In 1966, an NC system that made complete use of integrated circuits first appeared. That system, the Fanuc 260, was further improved, and modularization was achieved in 1969. That is to say, if we view these changes in NC systems from the perspective of decomposition into subsystems, NC systems that were implemented with the idea of modularization had already appeared in 1969. However, the modularization of that time was achieved by hard-wired technology, in which various kinds of logic devices and memory devices, etc. are combined so as to implement the required functions in the form of hard-wired modules. Modularization based on MPU technology is same architecture as hard-wired module from the viewpoint of architecture, but the technology itself is greatly different.

For this modular NC system, the results of a detailed analysis of the specifications of various machine tools are referenced in order to design a number of function-specific modules, which are then mass produced and stocked. The NC system is then configured by assembling the appropriate modules according to the users needs. The modules of the Fanuc 260, then, were implemented with hard-wired technology, which is to say that they were electronic circuits that implemented specific functions by combining various kinds of logic elements, memory elements, etc.; specifically, they were independent printed circuit boards and units. As example of a Fanuc 260 configuration is shown in Fig. 1.



Source: Easy Reader in Numerical Contral (1969)

Figure 1. Fanuc 260 configuration example

From the figure we can see that the Fanuc 260 has three types of basic control units, nine basic options, and 20 extended options, which can be combined in over 60 million ways. These kinds of modules are connected by means of machine screws and cables, and so can be assembled with only a screwdriver and wrench, with no need for soldering at all (Note 11).

This is to say that there is a simple mapping of NC functional elements and structural elements in the Fanuc 260 system, with nearly a one-to-one correspondence for certain functions with highly independent modules, such as printed circuit boards and units. Also, although it can be said that rules had been established for the interfaces between the modules that are structural elements, such as the connections being made by means of screws and connectors, those rules are not published and shared within the industry. That is to say, we can conclude that the Fanuc 260 is an NC system that had a modular architecture. (Note 12).

This kind of architecture allowed for easy maintenance and addition of functions. For example, even when function is to be added or a failure occurs after delivery to the user, there is no effect on other modules, and what is more, modules can be replaced on-site one by one with only a screwdriver and wrench and without changing the circuit. It is possible to configure an NC systems that are best suited for types of machine tools by appropriately combining the various modules, with the result that high cost performance can be achieved from the users point of view. In other words, by implementing this modular architecture, the NC system can be understood as a combination of various highly independent functions, which is to say modules.

With this hard-wired modular NC system, the sales of Fanuc NC systems leaped upward. For example, the number of systems shipped was only 60 in 1965 and 388 in 1968, but in 1969, when modularization was implemented, about five times as many systems (1683) were sold as had been sold in the previous year (Mazzoleni, 1997). Because the functions of an NC system are not obtained unless the system is connected to a machine tool, the specifications of the NC have the property of being determined by the specifications of the machine. What that means is that NC systems are basically made to order, and so are a product that is difficult to mass produce. Before modularization, even though there was a demand for NC systems, the number of systems that could be produced was limited by the made to order nature of those systems. However, modularization made it possible to satisfy the seemingly contradictory requirements for both made to order systems and mass production at the same time. By achieving modularization in 1969, Fanuc was able to satisfy the two seemingly contradictory requirements of mass production of diverse types of NC systems, and so was able to greatly increase sales and expand market share.

3.2 Technological issues due to the introduction of the microprocessor — from hard-wired modules to integral architecture

As described above, Fanuc had already been able to secure a high market share with hard-wired modules, aggressively taking on subsequent advances in semiconductor technology. In 1975, Fanuc was successful in developing the world's first NC system that integrated an MPU, which was the Fanuc 2000C system (Note 13).

In the 35 years history of NC systems, it goes without saying that the most epochal elemental technology was microprocessor (MPU). The crossing point for that adoption was the switch from an NC system centered on hard-wired technology to an NC system centered on the MPU, which is to say a switch to a soft-wired system. At that time, the knowledge and expertise accumulated in the hard-wired NC era became useless, so we can infer that technical issues in architecture design arose immediately after introduction of the MPU, complicating the function-structure mapping and the interface design, and shifting the NC system architecture back to an integral architecture again. In this section, we consider whether or not there was a change to an integral architecture just after the introduction of the MPU by verifying whether or not the introduction of the MPU did create technical issues in NC architecture design, and if it did so, what kinds of technical issues were created.

Concerning the design of NC system architecture, one realistic and valid way to consider whether or not technical issues have arisen in design as the result of interdependence of subsystems and lack of stability in the interfaces is to observe the change in the size of the NC systems with adoption of a new elemental technology. As mentioned earlier, because NC system need to be connected to a machine tool, the reduction in NC system size is extremely important technical issue, and MPU is adopted to make NC system more compact. If there are technical issues concerning the design of the NC architecture, however, it is not necessarily true that the overall system can be made more compact, even if miniaturization is achieved at the MPU level. Moreover, if the new elemental technology is an epoch-making one, then will be to some extent necessary to change the existing architecture, thus creating technical issues concerning architecture design that are highly likely to result in an increase in the size of NC system immediately after introduction.

In the case that the existing architecture can basically be used, on the other hand, the adoption of the epoch-making elemental technology should be directly linked to a reduction in product system size. That is because if the new elemental technology can be incorporated into one of the subsystems that constitute the existing architecture, then a reduction in the size of the entire system can be achieved (Henderson and Clark, 1990). That is to say, whether or not major technical issues in the architecture design are created at the time of introduction of MPU can probably be inferred from observation of the size of the NC system. NC systems have various functions such as motor driving functions in addition to the arithmetic and logic operations that are provided by the MPU, and coordinated execution of those functions for the systems as a whole is made possible by integration of those functions. In that sense, the architecture of the system is very important, and the degree of system miniaturization should be sensitive to the effects of the existence of technical issues related to architecture design. If it is observed

Table 2. New elemental technologies and the surface area of CNC printed wiring boards

NC system name	Start of sales	Surface area of NC printed wiring board	Main new elemental technology and characteristics
FANUC 220	1962	60.2	Transistor
FANUC 260	1966	7.5	Complete conversion to integrated circuits. Subsequently, improvement and modularization in 1969
FANUC 200A	1972	Not measured	Built-in dedicated NC minicomputer. Functionality could be changed by replacing software.
FANUC 2000C	1975	9.5	First incorporation of a microprocessor.
FANUC SYSTEM 6	1979	4.6	Adoption of custom LSI. Adoption of bubble memory.
FANUC Series 0	1985	2.7	Large-scale custom LSI.
FANUC Series 15	1987	Not measured	Surface mounting, 32 bit bus.
FANUC Series 16	1991	1.5	Three-dimensional mounting on electronic component printed circuit board.
FANUC Series 16i	1997	1.0	Integration of NC control unit with LCD.

In the above table, the values for the printed wiring board surface areas are normalized to the value for the i series, which is the most recent model of the system.

Source: Created by the author from Fanuc internal documents

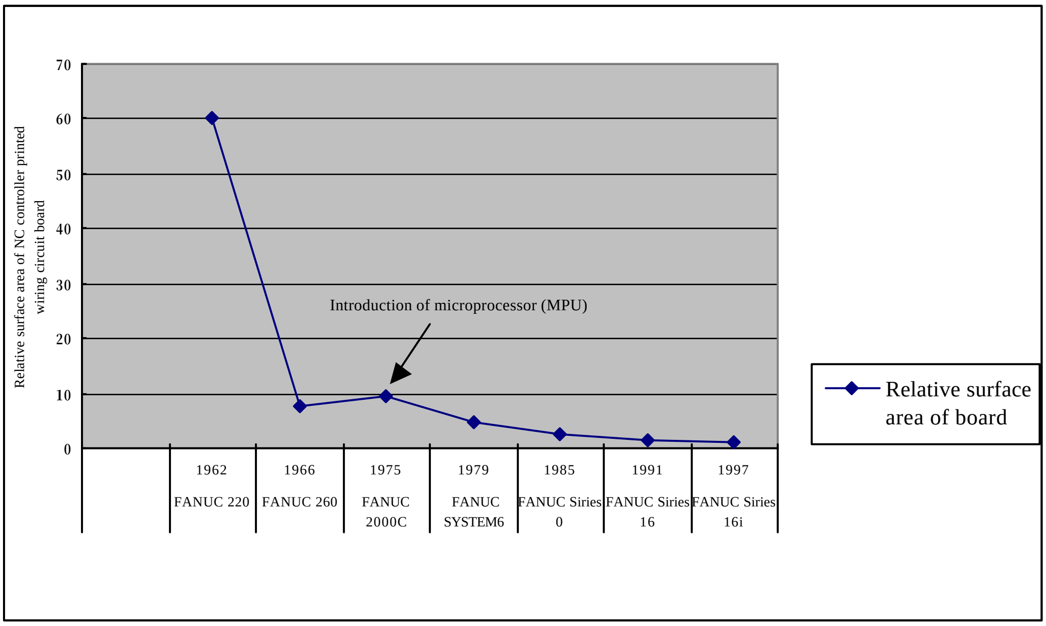


Figure2. Change in printed circuit board area due to introduction of MPU

that there is a temporary increase in system size at that time only, then it is reasonable to say that technical issues concerning architecture design existed.

On the basis of the above discussion, we here look at the changes of size of NC system with adoption of a new elemental technology. Data on the surface area of the control unit printed circuit board of the main NC systems and the elemental technology that was newly adopted at that time is presented in Table 2.

The newly adopted elemental technologies and the relative surface areas of NC control unit printed circuit boards for the main NC systems from 1962 to 1997 are shown in Table 2, and that data is presented in graph form in Fig. 2. What is clear from that table and figure is that in the technological development of the NC system over this 35 year period, there is only one instance in which the printed circuit board surface area increased, and that is when the MPU was introduced for the first time. The relative surface area Fanuc 2000C printed circuit board, to which the MPU was first introduced, is 9.5, which is larger than the 7.5 of the Fanuc 260, which makes full use of integrated circuits. That is to say, the introduction of the MPU, an epoch-making elemental technology, resulted in a temporary increase in the size of the NC control unit printed circuit board compared to the size prior to the introduction. Why did the overall NC system increase in size even though the size decreased at the level of the elemental technology? This fact suggests that the introduction of the MPU created technical issues concerning the architecture design.

Next, we will clarify specifically what kinds of technical issues in architecture design existed (Note 14).

At the time MPU was introduced, the technical issues that Fanuc faced can be broadly categorized as performance issues and reliability issues. The performance problem was whether or not the performance of the interpolation function for calculating the locus of the tool according to the machining instructions and distributing the actuator pulses accordingly is adequate. In the hard-wired NC system, the interpolation function was implemented with a physical combination of integrated circuits, which accounted for a very large amount of the volume of the NC system. If that circuitry could be replaced by an MPU, then a reduction in size could be achieved, which would offer a major advantage to the user. However, the introduction of the MPU meant that much of the processing that had previously been done by hardware had to be done by software, and that created issue of whether or not sufficient performance could be achieved. In particular, the question of whether or not sufficient performance could be achieved by software in the processing interpolation was a very important issue. If it was not possible, the pulses that are distributed to the servo circuits would be late and broken off, so that smooth machining would not be possible. However, obtaining superior performance by software processing is not simply a matter of making software logic efficient by itself; how the functional load is divided between software and hardware and what kind of interface to establish between the two in order to realize smooth exchange of data are important as well. That kind of knowledge and expertise, however, probably differs from the knowledge and expertise accumulated in hard-wired module design. Functions that had previously been implemented by physical combinations of logic elements and memory devices had to be implemented by a division of functions between hardware and software centered around the MPU. This is the issue of what kind of structural element combination and interfaces to use to implement the interpolation, which is indeed related to the design of the NC system architecture.

Table 3. Technical issues in architecture design caused by introduction of MPU

	Technical Problems	Problems in Architecture Design
Performance	Maintaining the same system performance as the hard-wired NC system in a software implementation of the interpolation function	How to divide the functionality between software and hardware How to design the hardware-software interface so as to obtain high performance
Reliability	Maintaining the same level of reliability as that of a hard-wired system in the severe temperature and noise environment of a factory. Maintaining reliability when the NC program is stored in semiconductor memory. Being able to store the program in semiconductor memory without the data being erased or altered.	How to use the MPU and the semiconductor memory in combination in order to maintain the high reliability of the NC system How to do the wiring between ICs and what kind of edging to used in order to achieve robustness against noise and temperature Should a separate interface be designed so as to prevent data from being erased in only some of the IC areas?

Source: Created by the author from the results of personal interviews with Fanuc personnel.

The second type of problem concerned reliability. Maintaining reliability in the severe factory environment in which NC systems are used has been an extremely important issue since that time. In contrast to ordinary indoor environments, the temperature inside a factory can vary from about 0 degrees centigrade to about 45 degrees centigrade, and there is much noise. The NC system must be able to operate normally even in such severe environments. The hard-wired NC systems of that time were able to maintain sufficient reliability in such factory environments, but whether or not an NC system that employs an MPU could achieve the same degree of reliability was a major question. That was just after MPU had appeared as a product, and there was no experience in adopting MPU in an NC system. Moreover, it was commonly recognized at that time that reliability decreased when semiconductor memory was employed. Amidst those circumstances, sufficient knowledge and expertise had not yet been accumulated concerning matters such as how to properly combine the MPU and semiconductor memory, how to do the circuit between ICs on the printed circuit board, how to do the edging, and whether or not the system would be sufficiently reliable to withstand the noise, temperature and vibration conditions of the factory environment. The question of what kind of interface to establish between the MPU and the semiconductor memory in order to maintain high reliability posed issues that could not be dealt with by the knowledge accumulated during the age of the hard-wired NC system; it required the accumulation of knowledge and expertise concerning a new architecture that was based on the MPU (Note 15).

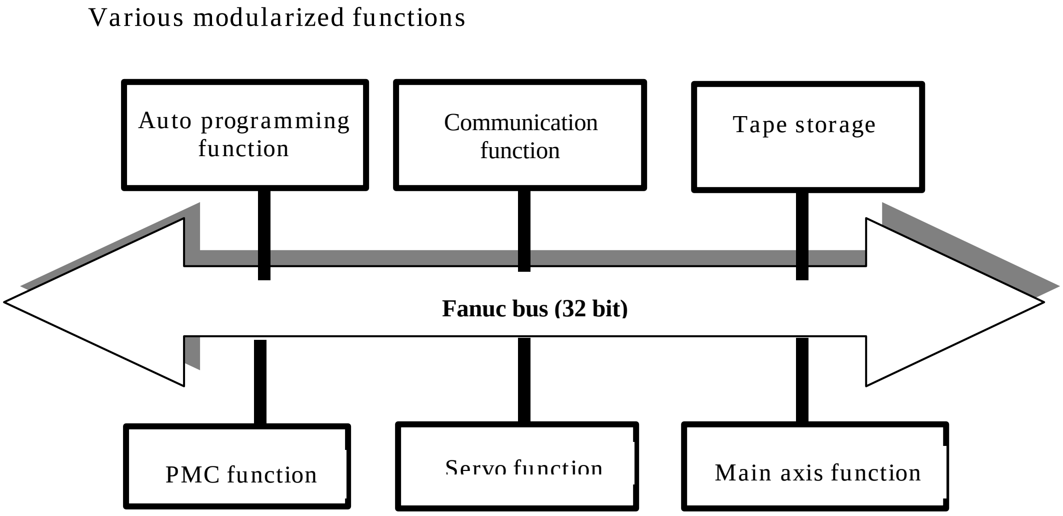
That is to say, just after the introduction of the MPU, sufficient knowledge and expertise concerning the design of the new architecture had not yet been accumulated, and we can believe that, as a result, the company faced the technical issues concerning the architecture design described above, and so the surface area of the printed circuit board of the 2000C increased temporarily.

Four years later, in 1979, the Fanuc system 6 series, which uses the Intel 8086 16-bit MPU, was completed. For the 6 series, as the result of accumulated knowledge and expertise concerning architecture based on an MPU, many custom LSI chips were designed for the hardware, which made it possible to reduce the number of parts by approximately 30 percent, thus reducing the printed circuit board size compared to the 2000C (Note 16). Furthermore, these components were connected via an interface called the Fanuc bus, which was a Fanuc proprietary standard, thus achieving interchangeability on the level of the individual component. Also, the 6 series included a 6T model for lathes and a 6M model for milling machines and machining centers, but those models actually shared the same hardware, and the models were implemented simply by changing the software.

Thus, great improvement in terms of architecture can be seen in the 6 series with respect to the 2000C, but no conclusion had been formed for the component units, much less had rules been established for the interfaces between the component units. For example, as with the display function and the control function, component units had not been formed for each similar functional element. Also, the software was all created by Fanuc, including the software that was used by the machine tool makers, and was stored in ROM. Accordingly, there had been no clear separation into a user-referenced part and a Fanuc-referenced part, nor had rules been established for the software user interface. That is to say, the architecture of the 6 series involved a complex mapping of functional elements to structural elements and the establishment of interface rules was not sufficient, so it is probably reasonable to consider it as an integral architecture. Thus, from this fact, too, we can infer that the architecture of the 2000C system was an integral architecture.

3.3 From soft-wired modules to a multivendor environment

If we classify the NC systems that are typified by the 6 series as the first generation NC systems, then the characteristic of the second generation NC systems that appeared subsequently is the implementation of a modular architecture based on the MPU. However, because the relative importance is shifted from the hardware to the software, this modular architecture is referred to as soft-wired modules as opposed to hard-wired module (Note 17).



Source: Created by the author from the Fanuc catalog.

Figure 3. Conceptual diagram of hardware modules

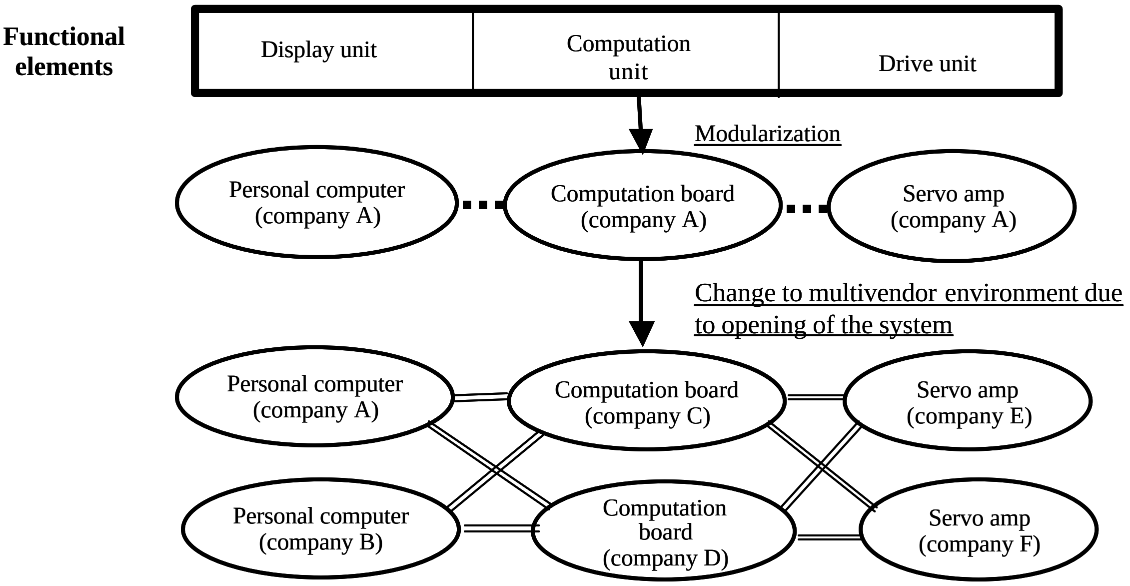
If we look at this second generation CNC system from the viewpoint of architecture, we can see that modularization had already been realized for both hardware and software. Speaking first of the hardware, there is one corresponding printed circuit board, called a module, for each functional element, as shown in Fig. 3, which means that the mapping of functional elements and structural elements is very simple. Also, each hardware module is connected to a common interface, the Fanuc bus, so rules had been established for the interface of each module. For the software, too, there was division into a user part and a Fanuc part by means of a clear specification of a user interface for the exchange of data between the user part and the Fanuc part.

The characteristic of the third generation NC that appeared next was an even greater simplification of the function-structure mapping than that of second generation

NC, with decomposition of the hardware into three major component unit, which were a display unit, a computation unit, and a drive unit. As a result, multivendor environment became possible.

The software followed suit with the customization function that was implemented in the second generation CNC, but the hardware configuration was decomposed into larger component units rather than the hardware modules for each individual functional unit of second generation NC. Then, because the interface between component units was an in-house Fanuc standard, individual component units could be independently added or modified.

The result of decomposing the NC system into a display unit, a computation unit, and a drive unit in this way was that it became possible to use an IBM-compatible personal computer for display function of Fanuc NC system rather that necessarily using specialized display equipment provided by Fanuc. That is to say, it became possible to construct an NC system by combining the computation unit and drive unit that were supplied by Fanuc together with an IBM-compatible personal computer that was available on the open market. In the sense that an open-market personal computer could be used, it can be said that the third generation CNC architecture made possible a multivendor environment for the display unit.

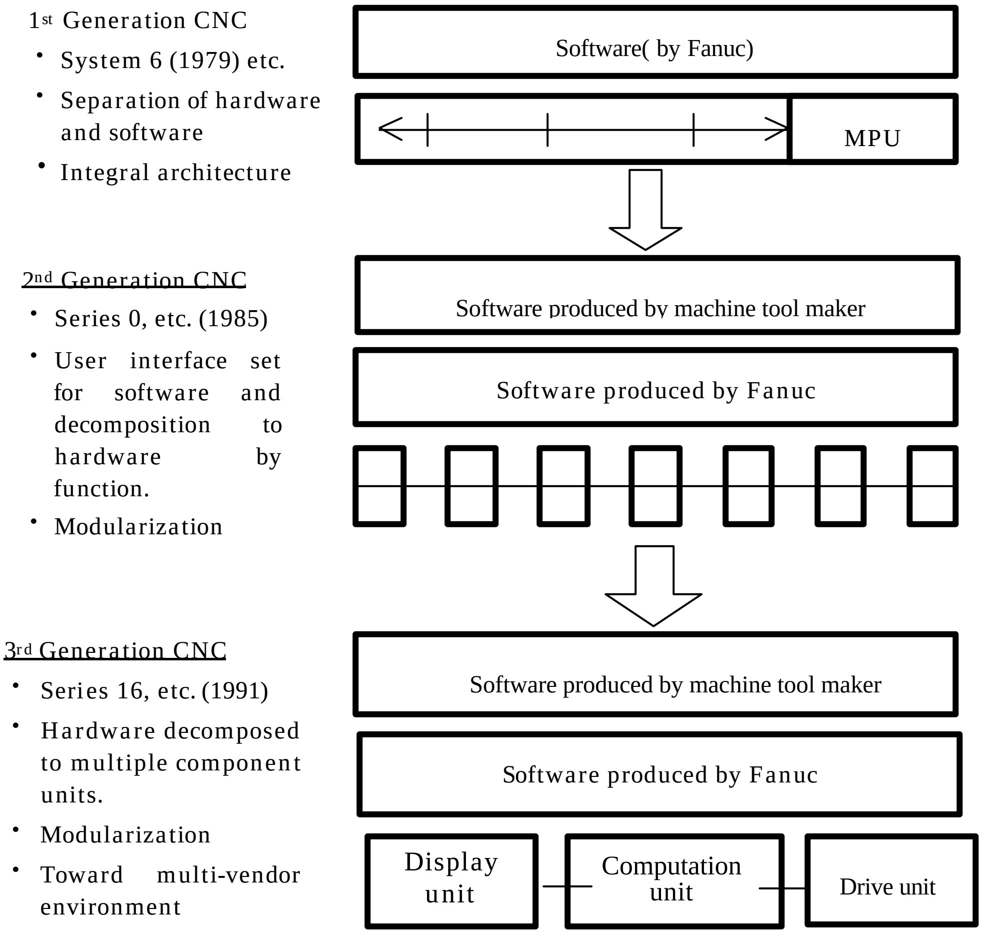


Source: Created from the Japan Development Bank Survey, Towards Further Development of Japan's Machine Tool Industry, May, 1999, No.257

Figure 4. Conceptual diagram of the change to a multivendor environment for NC systems

Multivender environment is a matter of implementing one product system by combining subsystems that are supplied by multiple vendors. In other words, this is nothing other than an attempt to provide high flexibility and freedom in product selection by dividing a single product system into multiple subsystems, and having those subsystems supplied independently by multiple vendors. To achieve that, it is necessary, first of all, to have a method of decomposition into subsystems in which the subsystems are highly independent and functionally unified. In that sense, the change to a multivendor environment is intimately related to the decomposition method of the product system, and that change cannot be realized without adopting an appropriate decomposition method. Actually, although the second generation CNC systems had a modular architecture, a multivendor environment was not possible because the decomposition method was not appropriate. The second generation CNC systems were decomposed into individual functions, so it was not possible to replace one part of the system with a commercial personal computer. The third generation CNC systems, on the other hand, the NC system was decomposed into three subsystems, namely the display unit, computation unit, and the control unit. If the interfaces and technical specifications of those subsystems are made public, that is to say if an open system architecture is implemented, then a full multivendor environment is possible in principle. A conceptual diagram of that is shown in Fig. 4 (Note 18).

One point to be noted here is that, as shown in Fig. 5, both the second generation and the third generation systems have a product structure that can be classified as a modular architecture, but the decomposition methods are different; the decomposition method does not end with that of the second generation CNC systems, but attainment of still further decomposition is seen in the third generation CNC systems. This can probably be understood as the existence of architectures that are still immature with respect to the simplification of the mapping from functional elements to structural elements and the establishment of simplified and rule-based interfaces even among the architectures that are classified as modular in form, and that there was a change in the decomposition method towards simplification and standardization.



MPU: Microprocessor

Source: Created by the author with reference to the Fanuc catalog.

Figure 5. Changes in NC architecture after introduction of MPU

4. Conclusion and implications

In this section, we briefly summarize what has been discussed here and consider what kind of hypothesis concerning the evolution of product architectures can be derived from that discussion. Then, we consider the validity of that hypothesis and finally conclude with an examination of the implications of the hypothesis.

First, to summarize the changes in NC system architecture, the Fanuc 220 system, which employed transistors was developed in 1962, complete conversion to integrated circuitry occurred in 1966, and then complete modularization was achieved in 1969 with the development of the Fanuc 260. That is to say, an NC system that adopted a modular architecture with hard-wired technology first appeared in 1969.

Following that, the Fanuc 2000C, which was the world’s first CNC to adopt an MPU, was developed in 1975. And then, it was confirmed that the overall size of the NC system increased, in spite of having adopted the MPU, an epoch-making elemental technology. From that fact, we conjectured that there might have been technical issues related to architecture design, and from the results of analyzing technical issues by means of interviews, we concluded that there was a change from a modular architecture to an integral architecture. Subsequently, the results of analyzing the NC architecture made it clear that after the introduction of the MPU two basic technical tendencies could be observed, which were the simplification of the mapping of the functional elements to the structural elements and the standardization and simplification of the interfaces. That is to say, there had been a change back from an integral architecture to a modular architecture. What is of profound interest there is that a modular architecture had already been achieved in the second generation CNC system of about 1985. The third generation CNC possessed the same modular architecture, but the decomposition method evolved along the two directions described above, and at the same time signs of a movement toward a multivendor environment were observed. This evolution of architecture is illustrated in Fig. 6.

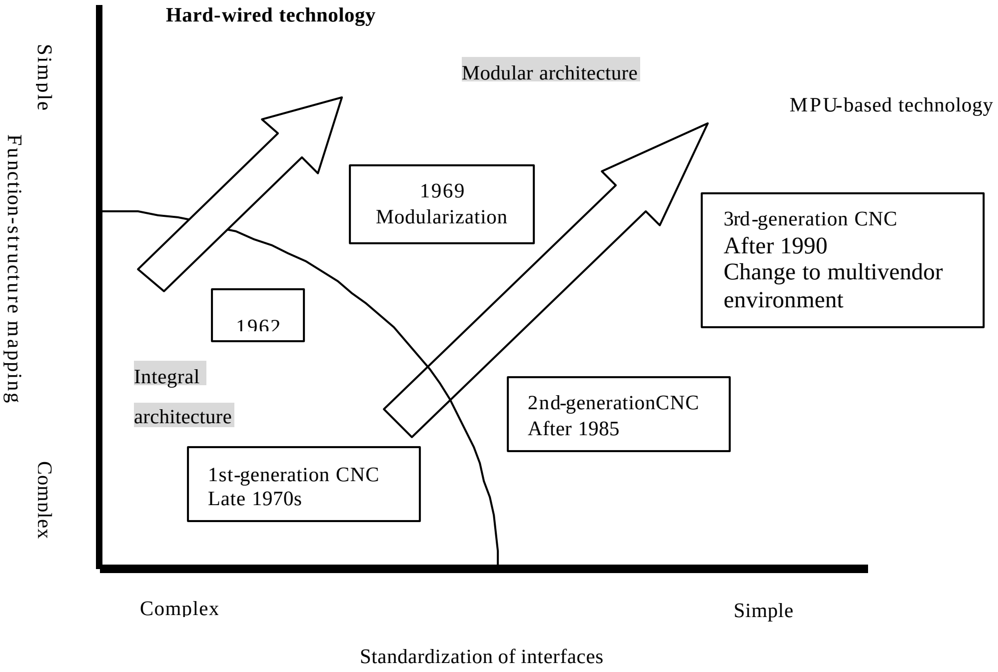


Figure 6. Evolution of NC system architecture

With this knowledge serving as a base, we can think of the evolution of technology in terms of product architecture rather than in terms of elemental technology. The product architecture can be considered to evolve according to the following two points of view. The first is that product system architecture evolves from an integral architecture to a modular architecture, and then again evolves toward an open architecture. The second viewpoint is that this change is not always in one direction, but when an epoch-making elemental technology is adopted the direction of change can reverse from a modular architecture to an integral architecture. What that means, in other words, is a return to a product system that has a complex mapping of function to structure and a complex interface. This kind of evolution of architecture can be considered.

Next, we would like to give a little more consideration to what factors and driving forces might result in evolution along those lines. First of all, there is the existence of a strong demand from the market. As described earlier, NC systems should be basically thought of as made to order systems, and first the NC system specifications are created to match existing mechanical specifications, an estimate is made and price negotiations are conducted. If an agreement is reached, then design is begun and components are prepared, and not a single NC machine tool is produced before this process is completed. In this way, however, mass production of a large number of NC systems is not possible, and costs cannot be reduced. If, however, all NC systems were ready-made, then it would not be possible to satisfy the requirements of diverse machine tools. This characteristic of NC systems, we can easily infer, strongly promoted modularization. The degree of market demand depends on the characteristics of the industry, which should have some effect on the speed of modularization. For example, one factor in the failure of modularization to develop so early in the automobile industry can be considered to be that the market demand for modularization is not as strong in that industry as it is in the NC industry.

Secondly, we can take the viewpoint of an inherent theory of technology. Comparing to the inherent logic of technical evolution clarified by Rosenberg, which is to say that the directionality of the evolution of technology is achieved through repeated conditions of technological imbalance and overshooting, how is this model proposed in this paper evaluated? The level of a hypothetical conjecture has not yet been reached, but we believe that the directionality of the evolution of product architecture that has been presented in this paper is a directionality for which a technological imbalance among the subsystems of the product system is actually easily actualized and for which Rosenberg's proposed mechanism for the evolution of technology easily works. That is to say, the inherent logic of the technology can bring about the evolution of the product architecture.

Thirdly, there is the viewpoint of the learning process of the enterprise. The reason for this is that it is the enterprise that actually implements a modular architecture, and advanced knowledge and expertise concerning the product system are required for that purpose. The customer, however, does not possess that knowledge and expertise. For example, it has been pointed out that a product design that has a modular architecture requires a higher degree of organizational ability than does an integral architecture (Baldwin Carlisss and Kim B. Clark, 1997). Accordingly, it is necessary to assume that a learning process at the design stage exists in the background of the evolution of an architecture from an integral form to a modular form (Note 19). The directionality and speed of product architecture evolution is probably determined by these multiply factors, but further research is required obtain a deeper understanding in that area.

Finally, we would like to touch upon the implications of this research on management strategy. Against the background of the open architecture of personal computer and the change to a horizontal corporate structure that accompanied it, the current mainstream thinking on management strategy is the strategy of concentrating on core competence and commissioning out other work outside the company. Certainly, in the case of a product system that has an open architecture, such as the personal computer, because the components are standardized and the interfaces are public, it can easily be inferred that the strategy can be effective. One wonders, however, how long such a strategy will remain effective. The model of architecture evolution proposed in this paper suggests that a product system that adopts a modular architecture does not stay in that state forever. When a new elemental technology is taken on in the face of technological change, there may be a reversion to an integral architecture. In the case that an interface is not made standardized to public, it is difficult to procure components from market efficiently, and so a relatively large weight on in-house development cannot be avoided. The question then arises how to retain diversity in research and development (Rosenberg, 1994).

If, in the field of investment in research and development, there is excessive solidification and loss of diversity, the result will be a loss of diversity in technological capability, and a high probability that the company will not be able to cope with an integral architecture (Note 20). That is to say, even if the current architecture is modular, if we take its evolution into consideration, this model suggests that it is necessary to retain a certain degree of technological diversity. In other words, This model of technical evolution sounds a note of caution concerning the danger of simply riding on the current of the times and employing a management strategy of concentrating on core competence without long-term strategy.

Acknowledgements

We are very much indebted to the Fanuc, Ltd. for the technical data and interviews they provided. The responsibility for the factual content presented here is ours alone.

Note 1 According to Masatoshi Shima (1987), who was involved in the design of the world's first MPU, the 4004, an architecture encompasses the "structure and framework and the approach and specifications concerning a computer". Furthermore, "architecture is an idea, and an idea is a concept from which individuality gushes forth."

Note 2 According to Fujimoto for example, "While the sedan passenger vehicle is a typical closed design, the truck has a rather open approach to design." (Nihon Keizai Shinbun, March 23, 1998)

Note 3 For recent discussion of modularization, see Baldwin and Clark (1997), Langlois and Robertson (1992), Robertson and Ulrich (1998), Ulrich (1995), etc. All of such discussion is premised on a modular product structure, and focuses on the effect on the product development process and the advantages of modularization.

Note 4 Theoretically, products for which the mapping relationship has been simplified, but the interfaces are complex and rules have not been established for them may also exist. The actual problem, however, is that the number of components decreases as the mapping relationship is simplified, and the interfaces should become simpler and standardized as the number of components decreases. That is to say, in actuality, these two axes are not entirely orthogonal, and a correlation between them exists, so these two categories are adequate for the following discussion.

Note 5 The kind of definition of architecture given above is also a concept that specifies the degree of product distinctiveness or the degree of component commonality (Robertson and Ulrich 1998). That is to say, for products that have an integral architecture, to the extent that they share common components, it is difficult to distinguish products. To the extent that one tries to achieve product distinctiveness, the more different parts the better. That is to say, the degree of component commonality and the degree of product distinctiveness can be said to have a negative correlation. In the case of modular architecture and open architecture, on the other hand, both product distinctiveness and component commonality are

---

possible. That is to say, even if component commonality is promoted, it is still possible to achieve product distinctiveness by changing the relationships among the components.

Note 6 Behind this kind of industrial evolution, there is a process in which users and companies are learning about the product while interacting with each other. On a basis of similar recognitions of issues, the work of Clark focuses more on the product concept. Clark focuses on the product concept hierarchy, and understands the process of industrial evolution as a process of descending through the product hierarchy as the company and the customer learn about the product (Clark, 1983). In the case of the automobile, for example, once the gasoline internal combustion system has been selected, the next step is to turn attention to items that are lower on the hierarchy, such as the body, etc..

Note 7 Another difference between this paper and the paper by Kusunoki and Chesbrough could be said to be whether the shift in product architecture is due to endogenous factors or to exogenous factors. Whereas Kusunoki and Chesbrough discuss a case in which the process of modularization itself, beginning from an integral system, promotes the next integration (i.e., the case of the hard disk drive industry), this paper discusses a case (i.e., the NC industry) in which the integration of the architecture is brought about by adopting the external technological innovation of the MPU.

Note 8 Concerning the significance and utility of the methodology of the case study in social science, there are pros and cons, and many arguments have been presented on both sides. One argument of the defenders is that case studies allow heavy interaction with the data and the phenomena, and are thus suitable for developing a useful theory. The critics, on the other hand, question the extent to which knowledge obtained from case studies has external validity, that is to say if generalization from the results is possible. Needless to say, this paper assumes the former position. For consideration of the case study methodology, see In (1996), Numagami (2000), etc.

Note 9 Inaba (1970) Easy Reader in Numerical Control

Note 10 Inaba (1980) Easy Reader in Numerical Control (4th Edition)

Note 11 Inaba (1970) Easy Reader in Numerical Control

Note 12 In the Fanuc 260 catalog, the term “fully modular” is used.

Note 13 An understanding of the technological circumstances of that time is beneficial to the development of the discussion that follows. Looking back from 1975, the world’s first MPU, the Intel 4004, was developed in 1971 by Intel. That, however, was not the single-chip MPU that we see nowadays; rather, it was a bit-slice MPU that comprised a chip set of four functional components, which were an arithmetic processing unit, a ROM unit, a RAM unit, and a shift register. Of those components, it is the bit slice cut off as the computation unit called the 4004 that should be called the progenitor of the MPU. After that, Intel developed the 8008 8-bit MPU in 1972, and an improved version of it, the 8080 was ready in 1974. Intel further came out with the 8086 16-bit MPU in 1978, which was incorporated into the Fanuc Series 6 systems that shipped in 1979, as is described later in this paper. That time period was the dawning of semiconductor technology, and it was a chaotic time of promise. Looking at the circumstances of Fanuc at that time, on the other hand, Fanuc already had a large share of the hard-wired NC system market, and system performance and reliability were both quite stable. It was under those circumstances that Fanuc developed the world’s first NC system that employed the MPU and semiconductor memory devices. Considering that semiconductor technology was immature at that time, adopting that technology for NC systems was a decision that involved a large risk. However, it can be considered that the early recognition of the technological limitations of the hard-wired technology that was the main-stay

---

technology at that time turned attention to the possibility of using semiconductor computers in NC systems at an early stage..

Note 14 This description of the technical issues that occurred at that time the MPU was introduced in 1975 is based on the results of a meeting with the company executives that was held on December 10, 2000. Concerning the joint development between Fanuc and Intel at the time the 8086 processor was adopted for the NC system four years after the MPU was first introduced, see Okuda (2000).

Note 15 Another issue that was related to reliability was the reliability of backing up the NC program after power to the NC system has been cut off. One short-coming of a CNC system relative to a hard-wired system is that with a hard-wired system, the NC program that describes the machining locus could be read into the computer memory from a paper tape. Thus, once the program was input to memory, the same program could be used for machining any number of times in subsequent days. To implement that kind of operation, however, the NC program must be kept accurately in memory even after the power to the NC system is turned off. If the NC program is erased or altered when the NC system power is turned off, then such repeated machining with the same program is not possible. At the time, SRAM (static RAM) was used as the semiconductor memory for program storage, so that the NC program can be retained in memory by maintaining a 2 V power supply to the memory unit alone, even if the system is powered down. That implementation, however, was technologically a very high hurdle. After the NC system is powered down, the IC circuits go to 0 V, but the particular area in which the NC program is stored must be continuously maintained at 2 V by batteries, which was a technically difficult problem. The entire memory space of the NC system comprises three memory areas: a ROM area for storing the basic NC software, a battery-powered SRAM area for storing a back-up of the NC program, and a DRAM area that is used as a buffer, and each of those areas has an established interface with the MPU. However, there was at that time no accumulated knowledge and expertise concerning how to implement the NC system with what combination of the three memory areas and with what interfaces so as to attain high reliability. This kind of problem concerning reliability stemmed from the fact that sufficient knowledge about what combination of subsystems and, furthermore, what kinds of interfaces between them, should be used to implement the NC system had not been accumulated at that time.

Note 16 Inaba (1980) Easy Reader in Numerical Control, 4th Ed.

Note 17 For details concerning the decomposition of the second-generation CNC systems and the third generation CNC systems, see Shibata (2000) and Shibata (2001a).

Note 18 There are various viewpoints regarding the concept of an open system, and there are various uses that depend on the actual circumstances. Distinguishing the following three viewpoints is probably effective for discussing open systems, that is to say the viewpoints of open interface (or open network), open architecture (technical specifications), and open data.

Note 19 From the viewpoint knowledge and expertise concerning a decomposition method for the product as a whole are required in order to achieve an appropriate modular architecture, a learning process that is behind the decomposition method is considered in Shibata (2000) and Shibata (2001a).

Note 20 From the viewpoint of the incongruence of product architecture and organization, Kusunoki and Chesbrough (2001) called the case in which the organization is not compatible with the integral architecture the “modularity trap”, and developed a interesting discussion for the case of the hard-disk drive industry. Also, from the viewpoint of compatibility between the product system and the business system, Shibata (2001b) conducted a questionnaire survey of design engineers in an attempt

---

to clearly demonstrate the existence of compatibility.

References

Abernathy, William (1978) *The Productivity Dilemma*, The Johns Hopkins University Press.

Abernathy, William, Kim Clark and Alan Kantrow (1983) *Industrial Renaissance*, Basic Books.

Anderson, Phllipp and Michael, Tushman (1990) Technological Discontinuies and Dominant Designs: A Cyclical Model of Technological Change, *Administrative Science Quarterly*, 35, pp604-633.

Arrow, Kenneth (1962) "The Economic Implication of Learning by Doing," *Review of Economic Studies*, Vol. 29, pp. 166-170.

Baldwin, Carlisss Y,and Kim B. Clark (1997) "Managing in the Age of Modularity," *Harvard Business Review*, Vol. 75 , No. 5.

Clark, Kim B. (1983) "The Interaction of Design Hierarchies and Market Concepts In Technological Evolution," *Research Policy*, No. 14, pp. 235-251.

Cohen, Wesley M., and Daniel A. Levinthal (1983) "Absorptive Capacity: A New Perspective on learning and Innovation," *Administrative Science Quarterly*, Vol. 35, April, pp. 147-160.

Chesbroughage, Henry W. and Ken Kusunoki (2001) "The Modularity Trap: Innovation, Technological Phase Shifts and the Resulting Limits of Virtual Organization," I. Nonaka and D. Teece, Eds., *Managing Industrial Knowledge*, London: Sage Press.

Christensen, Clayton (1998) *The Innovator's Dilemma*, Harvard Business School Press, Boston, Massachusetts.

Dosi, Giovanni (1982) "Technological Paradigms and Technological Trajectories," *Research Policy*, Vol. 11, pp147-162.

Dong-Jae, Kim and Bruce Kogut (1996) "Technological Platforms and Diversification", *Organization Science* Vol. 7, No. 3, May-June.

Doi, Yasushiro, Honda, Yogo, and Doi, Kuniko, Eds., NC System Dictionary, Asakura Shoten(in Japanese).

Iansiti, Marco and Jonathan West (1997) "Technology Integration: Turning Great

---

Research into Great Products," *Harvard Business Review*, May-June.

Inaba, Seiuemon, Ed. (1970), Easy Reader in Numerical Control, Japan Management Association(in Japanese).

Inaba, Seiuemon (1980), Easy Reader in Numerical Control (4th Revised Edition), Japan Management Association(in Japanese).

Inaba, Seiuemon (1991), The Yellow Robot, Japan Industrial Journal(in Japanese).

Henderson, R and K. Clark (1990) "Architectural innovation: The reconfiguration of existing product technologies and the failure of established firms," *Administrative Science Quarterly*, March, pp9-30.

Japan Society for the Promotion of Machine Industry Economics Research Institute, Superiority in International Competitiveness of the Machine Tool Industry and Mutual Innovation Mechanisms, May, 1993 (in Japanese).

Langlois, Richard and Paul L. Robertson (1992) "Networks and Innovation in a modular system: Lessons from the microcomputer and stereo component industry," *Research Policy*, 21, pp297-313.

Mazzoleni, Roberto (1997) "Learning and path-dependence in the diffusion of innovation: comparative evidence on numerically controlled machine tools," *Research Policy*, 26, pp.405－428.

Matsuzaki, Kazuhisa, Yamada, Toshiyuki (1997), "Organizational Relationships and Mutual Innovation in the Development of New Machine Tool Products," Kikai Keizai Kenyu, No. 28, pp. 71-86.

Numagami, Kan (2000) "Behavioral Management Science—Search for Unintended Results in Management Science," Hakuto Shobo(in Japanese).

Nelson, Richard R. and Winter, Sidney G. (1982) *An Evolutionary theory of Economic Change*, Harvard University Press, Cambridge, MA.

Okuda, Koji (2000) "Denda, Nobuyuki, When Intel was Still Small," Daily Industrial Times (in Japanese).

Robertson, David and Ulrich, Karl (1998) "Planning for Product Platform," *Sloan Management Review*, Summer.

Rosenberg, Nathan (1976) *Perspectives on Technology*, Cambridge: Cambridge University Press.

Rosenberg, Nathan (1982) *Inside the Black Box: Technology and Economics,* Cambridge: Cambridge University Press.

Rosenberg, Nathan (1994) *Exploring the Black Box: Technology, Economics and History,* Cambridge: Cambridge University Press.

Sanchez, R. and J. Mahoney (1996) "Modularity, flexibility, and knowledge management In product and organization design," *Strategic management journal,* Vol. 17(Winter special issue), pp. 63-76.

Simon, HA. (1981), *The Sciences of the Artificial (2nd Edition),* Cambridge, Mass; MIT Press.

Shima, Masatoshi, (1987) The Birth of the Microprocessor—the 4004, Iwanami Shoten(in Japanese).

Shibata, Tomoatsu, (2000), "Building a Concept of Learning by Decomposition—the case of numerical control architecture," Organization Science, Vol. 34, No.1, pp. 76-94(in Japanese).

Shibata, Tomoatsu, (2001 a) "Research on Leaning Processes in Product Design—Analysis of decomposition processes for numerical control systems," Doctoral Dissertation for the University of Tokyo (in Japanese).

Shibata, Tomoatsu, (2001 b) "Product Systems and Business Systems from the Viewpoint of Division into Subsystems," Research and Technology Planning.

Tushman, Michael and Phillip Anderson (1986) "Technological Discontinuities and Organizational Environment," *Administrative Science Quarterly,* Vol. 31, pp. 439-465.

Ulrich, Karl (1995) "The Role of Product Architecture in the Manufacturing Firm," *Research Policy,* 24, pp. 419-440.

Utterback, James (1994) Mastering the Dynamics of Innovation- how companies can seize opportunities in the face of technological change, Harvard Business School Press, Boston.

Yin, Robert. (1994) *Case study research: design and methods, second edition,* Sage publications Inc.

Japan Machine Tool Industry Association (1994), United Nations Industrial Development Organization (UNIDO), Machine Tool Industries of the

---

World(in Japanese).

Development Bank of Japan (1999), “Toward Further Development of Japan’s Machine Industry: Future Development as Seen from Technological Seeds in the Machine Tool Industry” Research Report, MAY 1999, No. 257(in Japanese).