Product innovation through module dynamics

Tomoatsu Shibata

Graduate School of Management

Kagawa University

2-1, Saiwai-cho, Takamatsu, Kagawa 760-8523, Japan

Phone: 81-87-832-1942

Email: shibata@gsm.kagawa-u.ac.jp

Pruduct innovation through module dynamics

Abstract:

The modularity of products is progressing in many industries. This study uses a case study to analyze the process of integrating personal computer (PC) and numerical control (NC) machine tools, and conceptualizes as module dynamics the product innovation that works in modular products. Module dynamics refers to the process of product innovation that occur in highest-order modules of modular products and that is achieved through a two-stage process of partitioning and integrating modules.

Keywords:

Modularity, Architecture, Innovation, Partition and Integration

1. Introduction

The open modularity of PCs has brought about a horizontal division of labor in the structure of industry, enabling a business model known as the direct model, such as the one adopted by Dell Computer. Modularity is one factor that has led to commoditization in the digital home electronics industry, including digital cameras and flat panel displays. While advances in modularity have had a substantial impact on many industries, the product strategy governing growing modularity is not clear. In this paper, we identify an evolutionary process within modular products, and we conceptualize this process as module dynamics.

Evolution within modular products is a dynamic of product innovation that only modular products have. One example of this dynamic is the phenomenon that gave birth to the new product concept of PC-NC, or PC-integrated NC, in which PC and NC machine tools are products for different customers in different industries. The phenomenon of module dynamics, however, does not hold true only for PC-NC; rather, it can be applied to every modular product.

In this paper, we review research on shifts in architecture and then explore the dynamics between modular systems, which are referred to as "module dynamics". We then offer an example of how new product systems may be created by modules imported from one modular system and integrated into another, possibly in a different industry. Case studies of NC and PC systems are presented, followed by the conceptualization of module dynamics and a discussion.

2. Previous studies

Previous studies reported that product architecture shifts gradually from integral to modular for the following three reasons.

First, the method by which customers evaluate a product changes with their demands. In response to these demands, companies adapt the architecture accordingly (Christensen and Raynor, 2003). During the introductory phase of a newly developed product, it is not at its most efficient and may not meet the increasing demands of customers. At this primary stage products evolve from integral rather than modular architecture because integral architecture is more suited to achieve a product's best performance. Following improvements in technology, however, the product's efficiency may surpass customers' primary demands and customer evaluation of this product may be more focused on speed and flexibility. Once the first generation has evolved, companies shift to modular architecture, which they regard as more suitable for producing and delivering products rapidly and flexibly.

Second, modularizing products helps companies forge effective relationships with suppliers (Fine, 1998). During the rapid development of a field of technology, companies find it difficult to develop related products independently. Nevertheless, companies should strive to avoid dependence on particular outside suppliers, by standardizing interface specifications before outsourcing(Chen,2005 ; O'Sullivan,2003). Modularizing a product in this manner is therefore advantageous. For this reason, IBM prefers the modular architecture of PCs with standardized interface specifications and therefore chooses to outsource production of many products.

Third, modular architecture is more demanding than integral (Baldwin and Clark, 1997; Shibata, Genba and Kodama, 2002). Designing each module at the same time and getting all of them to work as an integrated whole is difficult. Integrated modules do not always behave as expected and often must be redesigned. For example, IBM faced a great number of difficulties during the modularization of System 360, a family of six compatible computers and forty peripherals (Baldwin and Clark, 2000). To avoid such pitfalls, companies need to accumulate knowledge of and experience with the system through integral architecture before adopting a modular approach. Using this method, product architecture shifts gradually from integral to modular.

The later shift of modular architecture back to integral has been found to occur in response to either internal or external technological innovation. One example of external technological innovation is the introduction of micro-processing units (MPUs) for numerical control machine tools that resulted in an architecture shift from closed-modular back to integral architecture (Shibata, Yano and Kodama, 2005). MPUs were originally developed in the semiconductor industry, in which product hierarchy differs from that of numerical controls. One example of internal technological innovation is the innovation of heads in hard disk drives (HDD) (Chesbrough and Kusunoki, 2001). Changes in the type of head, from ferrite to thin film to magnetoresistive, resulted in architecture shifting from modular back to integral. The innovation was for the same product hierarchy, HDD, and thus can be regarded as an innovation of internal technology.

Technological innovations, either internal or external, can bring about this architectural shift in several ways. First, with new technological changes, customers come to value the efficiency of the products a second time, so companies respond by shifting back to integral architecture to make the entire product more suitable to consumers (Christensen and Raynor, 2003). Second, the knowledge and know-how that companies had accumulated and used to modularize the architecture becomes redundant upon introduction of innovative technology to the components because interfaces between subsystems have to be re-established (Shibata, Yano and Kodama, 2005). Since most companies have not accumulated enough knowledge and know-how to modularize under conditions associated with new technological systems, they must adopt integral architecture for a second time.

Fine (1998) also suggests a third reason for why architecture shifts from modular back to integral. The structure of some industries, for example the computer industry, is forced to change from vertical to horizontal, and according to Fine, back to vertical. A subsystem that plays a key role in modular architecture may achieve dominance in the market by unifying other subsystems to its own advantage. For example, Intel does not limit itself to designing MPUs; rather, it plans to design motherboards, which are currently designed by package manufacturers. Thus the industry can change in structure from vertical to horizontal, a process called partition, and, at the same time, back from horizontal to vertical, which is called integration.

Product architecture has been shown to shift back and forth between integral and modular. During this dynamic process, modular dynamics are realized using a two-stage process of modular partition and modular integration. The first stage is a process in which appropriate architecture is sought with advancing elemental technology and the dynamically changing modular partition of a given product. Modular partition is the method by which a product is divided into groups of modules, followed by the formulation of interface rules between the modules. For closed modular products, the design rules are proprietary within the company; thus, the cost required to make changes is less than with open modules[1].

In this case, the company has a strong incentive to search for appropriate modular partitions reflecting advances in elemental technology. This process is followed by the second stage, modular integration. If the modular partition achieved during the first stage meets certain conditions, it becomes possible to import specific modules from the modular products of other industries to the given modular product. As a result, the modular product evolves into a product that provides new customer value. Modular dynamics is thus realized through this type of two-stage process.

Also, modular dynamics works to reassemble the highest order modules. As a result, modular dynamics influences the core concept of the product and diversifies customer value. As a concrete example, we present the PC-NC creating process.

3. Case study: Creating process of PC-NC

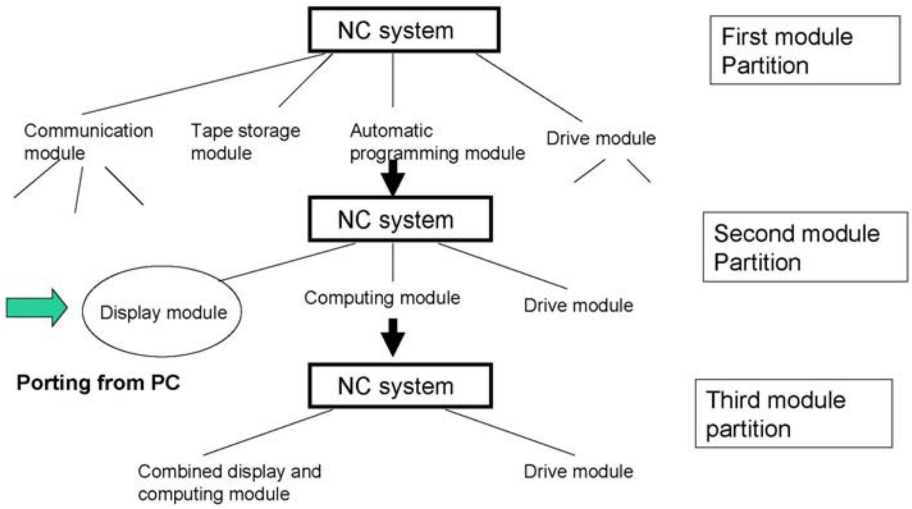
PC-integrated NC (a product resulting from module dynamics, referred to as PC-NC in this paper), the integration of the data processing function of PCs and the control of function by NCs, is becoming more common in the European and American markets for NC machines. In the United States, most of General Motor's (GM) NC systems became PC-NC. Below, we will illustrate an actual case of module dynamics of PCs and NCs to deepen our understanding of module dynamics.

PCs and NCs have followed different technological trajectories, in terms of both industry and customers. PCs are consumer goods aimed at the general public, whereas NCs are producer goods aimed at other machine manufacturers. The product architecture of both PCs and NCs is modular but it is open in the former and closed in the latter. Therefore, many PC modules, such as displays, motherboards and keyboards, can be purchased separately on the open market, whereas NC modules, such as display units, control units and servo units, cannot. Under these circumstances, PC modules have been ported to display units in NC systems and PCs and NCs have been integrated using module dynamics, creating PC-NC.

Searching for Appropriate Module Partition

Ever since 1975, when a microprocessor unit (MPU) was first incorporated into NC equipment architecture, searches have been made for appropriate module partition to accompany the latest advances in elemental technologies. As a result, NC architecture has achieved three different module partitions.

Figure-1 Search process of module partition



After the MPU was adapted into NC equipment, efforts continued to design products with modularity, with hardware modularity achieved for the series 0 (Zero) developed in 1985. As shown in Figure 1, functions such as communications, tape storage, and automatic programming became independent hardware modules, each equipped with an MPU, with a FANUC bus (a proprietary FANUC common interface) used between these modules to form a linked architecture. A printed circuit board known as a hardware module was used to implement each group of function elements. In that sense, the relationship between the function and structure elements is simple. Thanks to this modularity, functions can be freely added and selected in accordance with requests from machine tool manufacturers. This was the first-generation NC architecture and the first module partition.

Subsequently, innovative mounting technology, using printed circuit boards in three dimensions, was adapted for NC equipment, enhancing the ability to mount electronic parts densely. This use of advanced elemental technology influenced the method of module separation and stimulated new module separation in NC equipment. As a result, NC hardware could be divided into three major modular units, display, computing, and drivers. An architecture emerged in which the units were linked with an interface based on proprietary FANUC rules. Series 16, which employs this architecture, was released in 1991. The architecture consisted primarily of the three units of the human interface, display, computing, and drivers, physically linked by fiber-optic cables forming the FANUC Serial Bus proprietary standard interface. This was the second-generation NC architecture and the second module partition.

Further advances in elemental technology led to greater miniaturization. The display and computing units were combined into a single body, and an NC with two main units, the combined display and computing unit and the driver unit, was introduced in 1997 as Series 16i. Surface mounting and other technologies made it possible to mount NC control boards on the rear of LCD devices, combining the two into a single unit. As a result, it was possible to achieve an ultra-thin NC control board, of just 60mm, which reduced the space in a conventional NC unit by about one half. This was the third module partition.

Figure2 Advanced elemental technologies and NC miniaturization

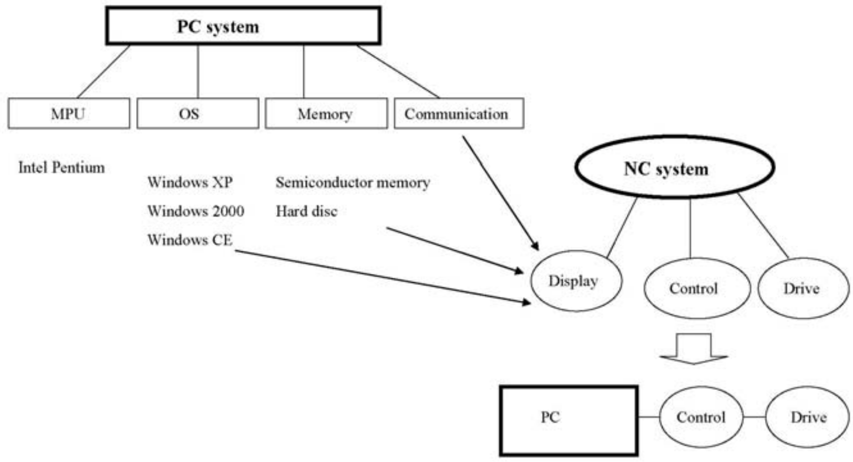
NC system name	Start of sales	Surface area of NC printed wiring board	Main new elemental technologies
FANUC2000C	1975	9.5	First incorporation of a microprocessor
FANUCSYSTEM6	1979	4.6	Adoption of custom LSI Adoption of bubble memory
FANUC Series0	1985	2.7	Large scale custom LSI
FANUC Series15	1987	Not measured	Surface mounting 32 bit bus
FANUC Series16	1991	1.5	3dimensional mounting
FANUC Series16i	1997	1.0	Integration of NC control nit and with LCD

In this way, companies followed a repeated process of recreating module partition in order to find the optimal modularity that met advances in elemental technologies. As shown in Figure 2, this process further stimulated miniaturization as well as simplifying the relationships between functions and structures.

Module Integration between PC and NC[2]

NC equipment gradually became more compact, following a three-fold process of module separation, while module integration was achieved by second generation NC architecture. This architecture was modularized into three main functions: display, computing, and driver units. The display unit showed the locations of tools and machines or NC program screens, the computing unit calculated tool speeds and paths and transferred the data required for manufacturing to the driver unit, and the driver unit controlled the motor based on that data. These three units were linked via the FANUC Serial Bus, a proprietary standard interface. This architecture was made possible with the porting of PC functions to the display module.

Figure-3 Module Integration of PC and NC



Porting a PC function to the display unit of an NC created an NC system with flexible and enhanced PC functions, such as database and networking. The database function, for example, enabled the NC operator to manage tool files, customize operation screens, and freely build human interfaces. The networking function of a PC could also be used to operate the NC from a remote location within the factory via the Internet. The database and networking functions ported from the PC changed the NC system from just a control unit for the tooling machine to an information processing unit capable of analyzing control data or operating the system remotely. The combination of the abundant information processing functions of a PC with the control functions of an NC heralded innovations that turned the NC equipment into a product with diverse value at a more advanced level.

There are two ways by which PC function could be integrated with NC. One way, as on the market, is to port PC to the NC display unit. This requires an Ethernet interface board for communication between the NC and PC. CNC library software, FANUC' s original package software, must also be installed in the PC to share NC data regarding tools and machine operations. The other way is to build a FANUC proprietary display unit embedded with PC functions (called PANEL i) by combining several modules on the market, for example, modules of Intel MPU, Windows XP and semiconductor memory. With those modules, PC functions are ported to the display unit of the NC. PANEL i is the original display unit of FANUC equipped with PC functions. In this case neither an Ethernet interface board nor CNC library software is required. In this way, a PC suitable for specific environmental conditions of factories can be built by combining optimal modules on the market. One example is PANEL I, consisting of Windows CE instead of Window XP and semiconductor memory that can withstand oily and noisy factory environments. Such a feature cannot be obtained simply by porting a PC on the market to the display unit of an NC.

Attention must be paid to the architecture realized in the third module partition. This architecture, which comprises two module units (the display and computing modules in a single unit and the driver module), is more compact and advanced, but is not suited to achieving module integration. To achieve PC-NC, both a computing unit and a driver unit are required, in addition to the information processing functions of a PC. An architecture divided into three modules, the display, computing, and driver units, is needed to transfer PC functions only to the display unit, suggesting that modular dynamics involves a relationship of congruity between module separation and module integration, and that a shift to module integration occurs when those conditions are met.

4. Building the concept of module dynamics

As shown in this historical analysis of the process of PC-NC creation, there is a logical product evolution behind modular dynamics. In this section, we discuss the logical validity of modular dynamics while referring to the case study described above. As mentioned, modular dynamics is realized through a two-stage process of module partition and module integration.

Dynamism of Module Partition

From the aspect of product hierarchy, module partition refers to the highest order division of modules. As clarified in the example of NC module partition, module partition is not stagnant but changes during the search for the optimal module partition. A major factor underlying the dynamism of module partition is the incongruity between progress in the elemental technologies that make up product systems and current module partition. Because of this incongruity, appropriate module partition reflecting advances in elemental technologies needs to be created anew.

Technological progress makes it possible for a single device to perform the functions previously performed by multiple devices. One typical example is system LSI, in which multiple LSI functions have been integrated into a single LSI. Surface and 3D mounting technology have enabled high integration and high density mounting, allowing multiple functions to fit within the same space. For example, functions that required 20 printed circuit boards now require just one. Realization of multiple functions within a single device causes a change in the mapping relationship between functions and structure. Incongruity with current module partition also occurs, and a new partition is sought to eliminate this incongruity. As a result, the number of physical modules making up the system declines, and design rules defining the interface between modules become simpler. Both improve product quality and promote miniaturization. Advances in elemental technology bring this sort of dynamism to module partition.

In the case of complex product systems configured of diverse elemental technologies, the dynamism of module partition becomes even more complicated. Since the speed of advances in elemental technology varies with each type of technology, the use of the most advanced technologies exposes a new bottleneck (Burusoni, Prencipe and Pavitt, 2001). For example, the processing speed of semiconductors doubles every year and a half or so following Moore's Law. However, machine technologies do not necessarily advance as rapidly, causing an imbalance between semiconductor technology and machine technology. To eliminate this imbalance, new design rules and module partition are required. However, because the product is configured of diverse elemental technologies, it is not clear whether a smaller number of modules and simple design rules are indeed appropriate. Thus, in the case of complex products, the dynamism of module partition becomes even more complicated.

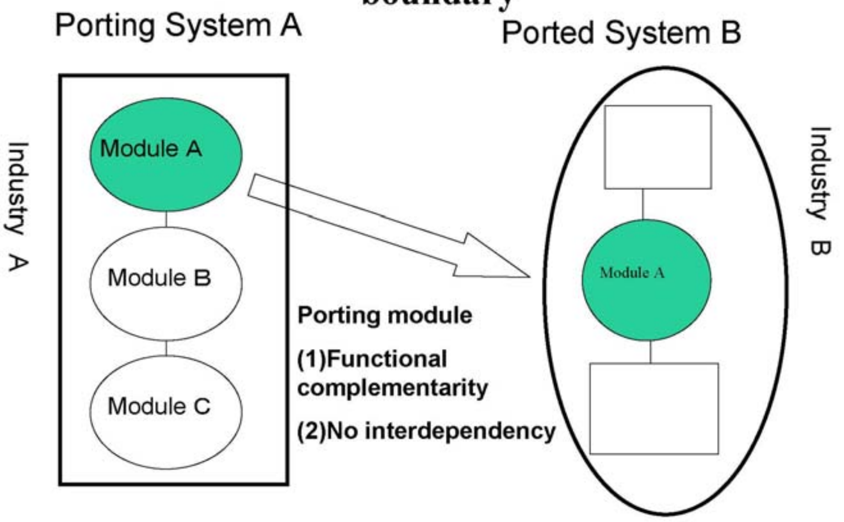
When making decisions on module partition, companies consider not only advances in elemental technology but also value and benefits to customers. Product miniaturization due to advances in elemental technology is a valuable change for customers. However, this change makes it difficult to generate only functions that users really need and presents difficulties in configuring the system, because the reduction in the number of modules forces customers to use modules that include functions they do not need. It is better for module partition to support detailed functions as much as possible. That also enhances its value as an option (Baldwin and Clark, 2000). Such a change, however, would lead architecture in the opposite direction to the one created reflecting the latest elemental technologies.

Thus, although modular architecture may have been established, complex factors drive a company to search repeatedly for more suitable architectures. This may include the redrawing of module boundaries and the recreation of design rules and module partition. Closed modules in particular have proprietary design rules established by the company; because these are not necessarily disclosed or shared, costs for making changes are borne by the company and are therefore lower than those made in open modules. Companies have a strong incentive to find more suitable design rules through trial and error and to redraw module boundaries, which brings about dynamics in module partition.

Module integration beyond industry boundary

If module partition satisfies certain conditions, module porting between products that have evolved along different technology paths occurs across different industries, and module integration indicates a phenomenon in which product architecture is revised. This logical validity can be explained rationally by the function of a porting operator that is also a modularization operator.

Figure-4 Module Integration beyond industry boundary



The other 5 operators are splitting, substituting, augmenting, excluding, and inverting. Baldwin and Clark assert that each of these six operators works on each module independently, causing the rapid technological development of modular architecture (Baldwin and Clark, 2000). The porting operator, as the name suggests, ports modules to other systems and is the only operator that operates on other systems. The other five operators only work within their respective systems. The system to which the module is ported is called the ported product system, while the system from which it is ported is called the porting product system.

A typical case of the porting operator at work is the UNIX operating system, whose modular architecture was ported to a wide range of hardware (Baldwin and Clark, 2000). The first version of UNIX (OS) was written for PDP-11s, a series of Digital Equipment Corporation (DEC) minicomputers. Because it was written mainly in assembler language, its use was limited to PDP-11. However, because the OS was rewritten and further developed in C, and because of its advanced modularity, it became easy to adapt for use in a wide range of computer systems.

When a module is ported and integrated with other modules, a new value is created. When the porting system has a modular architecture, a module may be ported, regardless of whether the architecture of the ported system is integral or modular. When the architecture of the ported system meets the following conditions, module dynamics can be brought about.

First, the architecture of the ported system should be modular in order to make substitution possible. However, the architecture of the ported system is not important regarding the porting operator. Module dynamics can only be possible when substituting a specific module in the ported system.

Second, there must be functional complementarity between the porting module and the ported system, functional complementarity meaning that the value of a certain function complements the value of other functions and enhances the value of the entire system[3]. Module dynamics is advantageous to the ported system only when the ported module complements the ported system functionally and porting enhances the value of the entire system. Even so, if the cost of integration is too high, the advantage of module dynamics will be lost. Cost of integration depends upon the degree of interdependency between modules within the ported system, leading to the third condition.

Third, there should be no interdependency between the porting module and the ported system. Otherwise the porting module will require modification, which may reduce its reliability and increase costs. In the absence of interdependence, a new product concept may be produced by module dynamics at very low cost. To eliminate interdependency within a ported system, the porting module should fit the product architecture of the ported system; that is, the way the ported system is modularized should suit the porting module.

Does this sort of module integration also occur in other industries? Theoretically, the porting operator does not work only in specific industries or products. If a product system can be clearly separated into a system-dependent section and a module-specific section, the porting operator can work in the module-specific section (Baldwin and Clark, 2000). Then the modules in which the porting operator works can be separated from the system-dependent section, i.e. away from the design rules of the system, and be incorporated into other system design rules. In that case, a function is required to translate between the new system-dependent section and the module-specific section (Baldwin and Clark, 2000). In the case of the PC-NC mentioned above, the interface board performed that role. In this way, the porting operator itself can function as long as the product system has modular architecture, regardless of differences between products or industries.

Module dynamics and product hierarchy

While modular dynamics works on the highest-order module in a product hierarchy, it is important to determine modular dynamics works from the perspective of product hierarchy.

All complex product systems can be expressed as hierarchical structures with quasi-separation possibilities (Simon, 1981). A system with such possibilities can distinguish between the strengths of interaction, both among and within low-order systems. Interaction among low-order systems is comparatively weak, but cannot be ignored. The quality of quasi-disintegrative possibilities is prominent in modular architecture. In the case of a PC, for example, the monitor, keyboard, main unit and other devices are probably positioned in the highest order of the product hierarchy. Multiple motherboards can then be positioned below the main unit, and the CPU, memory and other components below that. As a feature of quasi-disintegrative possibilities, the interaction within systems, such as the CPU and memory, which make up the motherboard, is much stronger than the interaction among systems, such as the monitor, keyboard, and motherboard. In a product hierarchy such as this, modular dynamics works on the highest-order module.

Accordingly, modular dynamics works on different areas of the product hierarchy than the areas of module components such as CCD, MPU, or system LSI. Unlike modular dynamics, so-called module components indicate modules positioned below these. This difference is significant. The structure of a product hierarchy has a close relationship with a core concept (Clark, 1983). Since modular dynamics ports a new external module to the highest-order module, it influences the product's core concept while also diversifying customer value. The porting of the module component, however, has little impact on the core concept(Nobelius and Sundgren,2002). For example, porting the CCD of another manufacturer may indeed increase the number of pixels of a digital camera, but it does not influence the core concept of the digital camera. In the case of PC-NC, however, customer value was diversified by porting the information processing function into the product, which had not been provided in existing NC equipment. Modular dynamics working on the highest-order module in the product hierarchy thus directly influences customer value.

Furthermore, since modular dynamics works on the highest-order module, users can port external modules themselves and not be dependent on vendors. For example, Mori Seiki Co., Ltd., a leading Japanese machine tool manufacturer and user of NC systems, developed their own PC-NC by porting consumer PCs to the display module of their NC system.

Mori Seiki had been using standard NC equipment supplied by NC manufacturers such as FANUC and Mitsubishi Electric[4]. Mori Seiki purchased NC equipment from these manufacturers and fitted it to its own machine tools. However, automobile manufacturers and other machine tool users were suffering because the usability of operation panels differs between individual NC equipment manufacturers, complicating user operation. Mori Seiki therefore developed its own operation panel based on PC, and designed its own PC-NC equipment by porting a PC based operation panel to the display unit of the NC equipment of NC manufacturers.

In other words, Mori Seiki ported originally developed PC functions to the display unit, and integrated it with the computing and driver units from NC manufacturers, creating its own PC-NC system by combining these units. Development began in 1997, and the MAPPS (Mori Advanced Programming Production System) I was released in 2000. Mori Seiki has now completed the improved MAPPS III, and is using it in all its models. This has enabled Mori Seiki to produce its own common specifications for operation and display methods independently of the NC equipment manufacturers.

Needless to say, this product strategy assumed the existence of NC architecture configured of three module units: a display unit, a computing unit, and a driver unit. Since the highest-order module of a product hierarchy is thus a module that customers can recognize, users can port it themselves. In the case of modules lower in the product hierarchy, however, users will probably have difficulty porting them on their own. This advantage derives from the fact that module dynamics works on the highest-order module in the product hierarchy.

5. Implications of Module Dynamics

Based on the above discussion, we conceptualize module dynamics as follows. Module dynamics works on the highest-order module in product architecture and is logically configured from a two-stage process of module partition and integration, leading to the diversification of customer value and influencing the core concept of the product. The logic of module dynamics is not limited to a specific industry but should be applicable to other industries developing modularity. What are the practical implications of this concept of module dynamics? In this section we look at recently developed commoditization and study the implications of module dynamics on it.

The commoditization of digital home electronics such as DVD players and thin-panel TVs in recent years has been remarkable. Since industries facing commoditization have found it difficult to differentiate themselves from other companies in non-price areas, they have been forced to engage in price competition. In this situation, companies are having trouble generating sufficient profit despite achieving major technology innovations. Module dynamics offers a direction for companies to break away from commoditization by diversifying customer value of the product.

By porting specific modules of modular products from other industries, module dynamics can expand the value dimension of the products, thus contributing to product differentiation. For example, the most important customer values of NC equipment are accuracy and speed, since the role of NC equipment is to control machine tools. Thus, efforts have been made to achieve technological innovations in NC equipment in the areas of high-speed and high-accuracy cutting functions. In contrast, PC-NC adapted a new value dimension of information processing functions, which diversified customer value of the products. The coexisting values of controlling and information processing diversified the value dimension that NC provides to customers and created product concepts such as PC-NC. By diversifying the value dimension, module dynamics suggests possibilities for differentiating products and breaking away from commoditization[5].

In this way, module dynamics may indicate possible product strategies for commoditization, since one of the causes of commoditization is modularity[6]. Many products undergoing commoditization are already being modularized at the parts level. Module dynamics exploits this process. Creating great cohesiveness at the highest-order module level and porting new modules from outside promotes the diversity of customer value. Since modularity at the parts level has already been achieved, larger module partition and the recreation of design rules that support it may be advantageous. To that end, the focus is not on existing module partition but on the larger perspective of the product system as a whole. New module partition can be flexibly designed and recreated while creative combinations are achieved with external modules.

In this paper, we have conceptualized the logic of product innovation, which functions in modular products beyond industry boundaries as modular dynamics. In the future, it is important to further elaborate on and verify this concept as well as to consider the capabilities required to achieve module dynamics.

References

Aoki, Masahiko and Masahiro Okuno (eds.) (1996), "Comparative Analysis of Economic System", The University of Tokyo Publishing.

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

Baldwin, CarlisssY, and Kim B. Clark (2000), *Design Rules: The Power of Modularity, Vol.1*, Cambridge, MA: MIT Press.

Brusoni, Stefano, Andrea Prencipe and Keith Pavitt (2001), "Knowledge Specialization, Organizational Coupling, and the Boundaries of the firm: Why Do Firms Know More Than They Make?" *Administrative Science Quarterly, Vol. 46, December,* 597-621.

Chen, Stephen(2005),"Task partitioning in new product development teams" *Journal of Engineering and Technology Management*, Vol.22, No.4, pp.291-314

Christensen, Clayton and Raynor (2003), *The Innovator's Solution*, Harvard Business School Press, Boston, Massachusetts.

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

Fine, Charles (1998), *ClockSpeed: Winning Industry Control in the Age of Temporary Advantage*, Reading, Massachusetts: Perseus Books.

Fujimoto, Takahiro, Akira Takeishi, Yaichi Aoshima (eds.) (2001), *Business Architecture*, Yuhikaku, Tokyo.(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, pp 9-30.

Iansiti, Marco, and Jonathan West (1997), "Technology Integration: Turning Great Research into Great Products", *Harvard Business Review*, May-June.

Kusunoki, Ken, and Satoshi Akutsu (2006), "Category Innovation: Theory of Commoditization Breakaway", *Organizational Science*, Vol. 39, No. 3.(in japanese)

Langlois, Richard N., and Paul L. Robertson (1992), "Networks and innovation in a modular system: Lessons from the microcomputer and stereo component industry", *Research Policy,* 21, pp 297-313.

Nobelius, O. and Sundgren, N.(2002)," Managerial issues in parts sharing among product development projects", *Journal of Engineering and Technology Management*, Vol.19, No.1, pp.59-73.

Nobeoka, Kentaro, Munehiko Ito, and Hirokazu Morita (2006), "Failure to Capture Value Because of Commoditization – the Case of Digital Home Electronics", RIETI Discussion Paper Series 06-J-017.

O'Sullivan, Alan(2003)," Dispersed collaboration in a multi-firm, multi-team product-development project", *Journal of Engineering and Technology Management*, Vol.20, No.1, pp.93-116.

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

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.

Shibata, Tomoatsu, Masaharu Yano, and Fumio Kodama(2005), "Empirical Analysis of Evolution of Product Architecture", *Research Policy,* Vol. 34, pp.13-31.

Simon, HA (1981), *The Science of the Artificial (2nd Edition)*, Cambridge, Mass; MIT Press (translated by Motoyoshi Inaba and Hideki Yoshihara (1987), *Science of System (New Edition)*, Personal media, Tokyo).

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

---

[1] It is important to distinguish between closed and open modules. In the case of closed modules, the interface between modules is shared internally within a company, limiting their range to divisions within a company. Closed modules do not have great influence on the structure of an industry. In the case of open modules, however, the interface between modules is shared with the larger society, thus extending their range beyond an individual company. As a result, open modules may greatly influence the structure of an industry.

[2] The case study on module integration is based on the product catalog of FANUC Limited and an interview conducted on September 15, 2005, with Mr. Shuichi Okamune of the Manufacturing Science and Technology Center (MSTC).

[3] Complementarity refers to a relationship in which one existence becomes the reason for another existence. For example, computer hardware and software are in a complementary relationship. Also, according to Aoki (1996), institutional

complementarity exists in systems and frameworks. In addition, because of complementarity, the diverse systems and frameworks that exist within a single economic system serve to further strengthen that system.

[4] The case study on Mori Seiki is based on an interview conducted on May 16, 2006, with Mr. Makoto Fujishima, then an Executive Director and manager of the Development and Manufacturing Division, and Mr. Nobuyuki Yamaoka, then Management Division Leader, as well as on Mori Seiki product catalogs.

[5] Kusunoki et al (2006) describe a real solution to breaking away from commoditization, showing that it is necessary to create a situation in which a comparison of competing products is difficult, by deliberately lowering the visibility of value dimension. Although module dynamics does not lower the visibility of value dimensions, it does promote diversity of value.

[6] Nobeoka et al (2006) describes three factors as causes of commoditization: modularization, the appearance of intermediate materials on the market, and the peaking of customer value.