Product Architecture Design

Implications for Modularization and Interface Management

Mikkola, Juliana Hsuan

Document Version
Final published version

Publication date:
2000

Citation for published version (APA):
Mikkola, J. H. (2000). *Product Architecture Design: Implications for Modularization and Interface Management*.

Product Architecture Design: Implications for Modularization and Interface Management

Juliana Hsuan Mikkola
LINK
Copenhagen Business School
Dept. of Industrial Economics and Strategy
Howitzvej 60
DK-2000 Frederiksberg
Denmark
Tel: +45 3815 2941
Fax: +45 3815 2540
Email: jh.ivs@cbs.dk

Paper presented at LINK Workshop
"Organizing Processes of Learning"
Copenhagen Business School, 15 May, 2000
Solbjerg Plads SP 114, Copenhagen Business School

Modularization refers to the scheme by which interfaces shared among components in a given product architecture are specified and standardized to allow for greater reusability and commonality sharing of components among product families. It is also a new product development (NPD) strategy for increasing product variety and customization. When interfaces of components or modules within a system becomes standardized, outsourcing decisions can be made accordingly with respect to a firm's long-term strategic planning of its NPD, manufacturing and supply chain management activities. This paper introduces a mathematical model for analyzing the degree of modularization in a given product architecture by taking into account the following variables: composition of new-to-the-firm (NTF) components in a given product architecture, the degree of substitutability of the product architecture, and interface constraints imposed by the product architecture. The application of the modularization function is illustrated with two design decisions of Chrysler Jeeps' windshield wipers controllers. The case shows that one of the product architectures has a higher degree of modularity than the other attributed by its higher substitutability factor and lower NTF component composition.

Keywords: Modularization; New product development; Interface management; Product architecture; Substitutability

Acknowledgements: I would like to thank Didier Chabaud and Marco Valente for their valuable comments on an earlier version of this paper, presented at the DRUID's 2000 Winter Conference.

1. Introduction

Globalization, deregulation, more demanding customers, the advances in information and transportation technology contribute to the complexity of designing and managing supply chains (van Hoek et al., 1999), and the management of new product development (NPD) activities. A growing number of high-tech firms (e.g., consumer electronics, automotive electronics, and elevator manufacturing firms) have embraced new approaches to the management of their NPD, manufacturing and supply chain management activities. In order to shorten NPD lead time, to introduce multiple product models quickly with new product variants at reduced costs, and to introduce many successive versions of the same product line with increased performance levels, these firms are pursuing modular product architecture development (NPD strategy), *mass customization*[1] (manufacturing and SCM strategy), and *postponement*[2] (SCM strategy). This paper focuses on the issues of modularization as a NPD strategy with the assessment of product architecture designs at the detailed product design level. Modularization is defined as the scheme by which interfaces shared among components in a given product architecture are specified and standardized to allow for greater reusability and commonality sharing of components among product families.

---

[1] In broadest terms, *mass customization* emphasizes the need to provide outstanding service to customers in providing products that meet customers' needs (through maximizing individual customization) at a low cost (through modular components) (Feitzinger and Lee, 1997; Gilmore and Pine, 1997; Kotha, 1995; Pine 1993). It allows companies to penetrate new markets and capture customers whose special or personal needs could not be met by standard products (Lee, 1998). Mass customization is also an outgrowth of the customer-service revolution (Fulkerson, 1997) involving careful coordination of order management, manufacturing, and distribution to provide customers with mass-manufactured products that are made to their exact specifications (Gooley, 1998), available on a timely basis at an acceptable cost (Fulkerson, 1997).

[2] The fundamental principle for designing products and processes so that supply chain efficiency can be optimized is *postponement*, which is about delaying the timing of the crucial processes in which the end products assume their specific functionalities, features, identities or 'personalities'. Such customization process takes place after some key information about the customers' specific needs or requirements is revealed. Hence, postponement can be seen as an information strategy. The delay of the customization steps is only valuable if the information about the customers' needs can be captured quickly and accurately (Lee, 1998).

In assessing modularization at the product architecture level, issues regarding decomposability and integration of components vis-à-vis interface management of these components become an important factor. In a modular design strategy (as opposed to integral design strategy), decomposability of the components and interface compatibility issues must the seriously considered. Consequently, the degree of modularization inherent in a product is highly dependent upon the number of components and the interface constraints shared among the components, modules, sub-systems, and systems. Many studies on modularization are qualitative and exploratory in nature, and there is limited evidence from the literature providing a systematic way to analyze modularization at the detailed engineering level and how it impacts interface management of components in product architecture designs. How can firms manage modularity of its products without understanding the fundamental relationship between components and interfaces at the root of product architecture?

In this paper, I focus on the issue of modularization in new product development at the detailed design level, taking as the unit of analysis a black box of which the functional specification (including planning activities) is set by the buyer while the detailed engineering (including design, purchasing, and manufacturing activities) is the responsibility of the supplier. In addition, a mathematical model is derived for analyzing the degree of modularization in a given product architecture by taking into consideration the following variables: number of components, number of interfaces, NTF component composition, and substitutability factor. The paper is organized as follows. Firstly, a brief literature on modularization, product architecture, and interfaces are reviewed, followed by a brief discussion on the effects of substitutability and components. Secondly, the modularization function is introduced along with the assumptions made for formulating the mathematical model. Finally, the application of the mathematical model is illustrated with two product architectures of Chrysler Jeep’s windshield wipers controller.

2. Related Literature

2.1. Modularization

The term 'modularization' refers to *modularity* (Baldwin and Clark, 1997; Sanchez and Mahoney, 1996; Meyer and Utterback; 1993), *modular innovation* (Hsuan, 1999a; Christensen and Rosenbloom, 1995; Henderson and Clark, 1990), *modular system* (Baldwin and Clark, 1997; Langlois and Robertson, 1992), *modular components* and *modular product design* (Schaefer, 1999; Sanchez and Mahoney, 1996; Sanchez, 1994), *modular product architecture* (Sanchez and Mahoney, 1996; Lundqvist et al., 1996; Ulrich and Eppinger, 1995), and *remodularization* (Lundqvist et al., 1996). For instance, modular innovation is an innovation that changes only the relationships between core design concepts of a technology without changing the product's architecture (Henderson and Clark, 1990). Langlois and Robertson (1992) defined modular system as a network of sub-products, which form a product that can be treated as an entity, that consumers can arrange into various combinations according to their personal preference. Similarly, Sanchez (1996) highlights how modular product architectures can permit the leveraging of a great number of product variations by mixing-and-matching different combinations of functional components.

Although mixing-and-matching of components is one of the advantages enabled by modularization, its complexities are also dependent on the degree of standardization and customization of the components vis-à-vis respective linkages embedded in product architectures. Mixing-and-matching of components tends to be more prominent at the end of the value chain (e.g., Swatch watches, Sony Walkman). Whereas modular innovation in the form of unique components inserted in product architectures for differentiating a product from that of the competitors' is more critical at the early stages of the value chain (e.g., IWIPE, anti-lock brake systems, air bags, etc.). Changes in product technology and functionality of modular innovations are not as visible and obvious as modularization in the form of mixing-and-matching. In this paper modularity is defined as the scheme by which interfaces shared among components in a given product architecture are specified and standardized to allow for greater reusability and commonality sharing of components among product families.

2.2. Product architecture

Product architecture is the arrangement of the functional elements of a product into several physical building blocks, including the mapping from functional elements to physical components, and the specification of the interfaces among interacting physical components. Its purpose is to define the basic physical building blocks of the product in terms of both what they do and what their interfaces are with the rest of the device (Ulrich, 1995; Ulrich and Eppinger, 1995). Product architecture is often established during the product development process. This takes place during the system-level design phase of the process after the basic technological working principles have been established, but before the design of component and subsystems has begun.

Product architectures can vary from modular to integral. Modular product architectures are used as flexible platforms for leveraging a large number of product variations (Gilmore and Pine, 1997; Meyer et al., 1997; Robertson and Ulrich, 1998; Sanchez, 1996; Sanchez 1999), enabling a firm to gain cost savings through economies of scale from component commonality, inventory, logistics, as well as to introduce technologically improved products more rapidly. Some of the reasons for product change include upgrade, add-ons, adaptation, wear, consumption, flexibility in use, and reuse (Ulrich and Eppinger, 1995). Modular architectures enable firms to minimize the physical changes required to achieve a functional change. Changes to product variants often are achieved through modular product architectures where changes in one component do not lead to changes in other components.

Conversely, in integral product architectures, one-to-one mapping between functional elements and physical components of a product is non-existent, and interfaces shared between the components are coupled (Ulrich, 1995). Changes to one component cannot be made without making changes to other components. With integral product architectures, firms may be able to customize their products to satisfying each customer’s particular needs. Costs of customized components tends to be higher due to the integral nature of product architectures where an improvement in functional performance can not be achieved without making changes to other components. This can be prohibitively costly for complex systems such as computers, automobiles, telephones, elevators, etc. As the interfaces of the customized components become standardized, its costs are significantly reduced as changes to product architecture can be localized and made without incurring costly changes to other components.

2.3. Interfaces

Interfaces are linkages shared among components, modules, sub-systems of a given product architecture. Interface specifications define the protocol for the fundamental interactions across all components and interfaces comprising a technological system. The crystallization and development of interface specifications has a tremendous impact on setting worldwide industry standards (e.g., GSM, TDMA, and AMP). Typical interface specifications for a consumer electronics product at the NPD level, for instance, often includes the tolerance specification of the components with respect to manufacturing processes, operating frequency bandwidths, maximum heat dissipation threshold, voltage and current requirements, housing dimensions, to name a few.

Sanchez and Mahoney (1996) explain how modularity intentionally creates a high degree of independence or a ‘loose coupling’ between component designs by standardizing component interface specifications. Sanchez (1999) furthermore classify seven different types of interfaces:

1. Attachment interfaces – define how one component physically attaches to another

2. Spatial interfaces – define the physical space (dimension and position) that a component occupies in relation to other components

3. Transfer interfaces – define the way one component transfers electrical or mechanical power, fluid, a bistream, or other primary flow to another

4. Control and communication interfaces – define the way one component informs another of its current state and the way that other components communicate a signal to change the original component’s current state

5. Environmental interfaces – define the effects, often unintended, that the presence or functioning of one component can have on the functioning of another (e.g., heat, magnetic fields, corrosive vapors, radiation, etc.)

6. Ambient interfaces – define the range of ambient use conditions (e.g., ambient temperature, humidity, elevation, etc.) in which a component is intended to perform

7. User interfaces – define specific ways in which users will interact with a product

In software platform designs, Meyer and Lehnerd (1997:180-181) identify three essential types of interfaces:

1. Internal program interfaces within the engine itself;

2. Interfaces between the system and the user or between the system and other information systems

3. Interfaces between the platform and the add-in modules attached to it

Interface constraints are restrictions imposed by the components and how interfaces are shared amongst these components in a given product architecture. When a given product architecture is decomposed into sub-circuits, the interface constraints of these sub-circuits can be evaluated in stages. For example, the so-called components of 'closed assembled systems'[3] (e.g., cars, mobile phones, computers, etc.) can often be divided into two groups: electronic (e.g., resistors, capacitors, semiconductors, etc.) and mechanical (e.g., pins, nuts, bolts, housing, etc.). Interface management also

---

[3] A 'closed assemble system' is a system that is enclosed by sub-systems with clear boundaries, and the individual sub-system must be linked together via interface and linkage technologies (Tushman and Rosenkopf, 1992).

deals with the issues of component integration or multiplexing, as opposed to decomposition or de-integration of a system into smaller components[4].

2.4. Components and substitutability

Standard components are often off-the-shelf parts, and have well defined technical specifications that are generally accepted as industry standards. These parts are often listed in catalogues with low unit prices varying accordingly with the volume purchased. New-to-the-firm (NTF) components, on the other hand, are components that are usually considered as unique by a firm, as such components often have high technological risks by inducing changes at interfaces shared with other components, thus altering the configuration of a product architecture. Often the risks are well justified by the technical superiority of these components, significantly improving the overall performance of the product. The use of NTF components is strategic in nature because the integration of NTF components into a product architecture are often hard to be imitated by competitors (i.e., modular innovation), thus creating competitive advantages for the firm, at least in the short-run. But too many NTF components hamper innovation due to the increasing complexity in interface compatibility issues with other components in the product.

Product architecture defines the way in which components[5] interact with each other. The substitutability factor of product architecture is a function of the number of product families made possible by the modular component as well as the number of interfaces required for functionality. For example, if a component of a given product architecture can be used in 10 families (or 10 times the same component), and 2 interfaces must be shared with other components/modules/sub-systems for functionality, then the substitutability factor of the product architecture is 5 components per interface. A perfect modular product architecture is comprised of standard components with high substitutability, allowing for high reusability and high commonality sharing of components. Conversely, a perfect integral product

---

[4] For a discussion of the effect of multiplexing of components in a system and its impact on modularization vis-à-vis supplier-buyer relationships, see Hsuan (1999b).

[5] Depending on the level of analysis, a component can be a part, a module, a sub-system, or a system.

architecture is comprised of NTF components with low substitutability, allowing for low reusability and low commonality sharing of components. Hence it is assumed that the degree of modularization in a given product architecture is constraint by the composition of its components (number of standard and NTF components), interfaces shared among the components, and degree of substitutability.

Hence, substitutability factor has implications for the following:

- reusability and commonality sharing of next generation platform designs
- the potential for a high substitutability factor is obtained when components are designed with reusability and commonality sharing in mind

3. The Modularization Function[6]

A simple mathematical model is derived to explain the relationship between the degree of modularization in a given product architecture with respect to the composition of its components (e.g., number of NTF components), and degree of substitutability. The unit of analysis is a black box of which the functional specification (including planning activities) is set by the buyer and the detailed engineering (including design, purchasing, and manufacturing activities) is the responsibility of the supplier. The beauty of a mathematical model is that it allows us to synthesize a complex phenomenon into equations and functions, leading to a wide range of theoretical examinations and simulations of the phenomenon. Although mathematical models are powerful for analyzing dynamic behavior of the variables, it is confined to the limited number of variables and the formulation can become quite complex with increasing number of variables.

In deriving the mathematical model, or the modularization function, following assumptions are made:

1. NPD of a black box[7] is used, implying that the product's functional specifications, including interface specifications, do not change over a period of time. This

---

[6] This section of the paper is a recapture of the mathematical model first presented at the DRUID's 2000 Winter Conference in Hillerød, Denmark, January 6-8, 2000, in Mikkola (2000).

assumption allows the evaluation of the architecture's configuration and components composition independently from other sub-systems.

2. A given product architecture is comprised of a combination of standard and NTF components.

3. It is argued that NTF components impose higher interface constraints. Therefore, the lower the NTF components composition in a product architecture the higher the degree of modularization.

4. Product architectures made entirely of standard components can be equally damaging as product architectures with high-NTF-component composition. It does not protect a product's technological content, and can be easily copied by the competitors. Thus, it is assumed that there should be some amount of NTF components in a product architecture.

5. All standard components are equally critical.

6. All NTF components are equally critical.

7. All interfaces are equally critical.

The assessment of degree of modularization in a given product architecture involves the following steps:

1. Define product architecture and its boundaries.

2. Decompose the product architecture into sub-circuits, so that each one of the sub-circuits can be assessed individually.

3. Assess the substitutability factor of the black box by counting the number of product families enabled by the black box, divided by the number of interfaces

---

[7] Buyers often consider components manufactured by an original equipment manufacturer (OEM) as black boxes, as they are treated as outsourced components.

required by the black box for functionality, in accordance with the level of analysis.

4. Count the total number of components comprising the product architecture. This can be accomplished by looking at the product's bill of materials (BOM).

5. Count the number of NTF components.

6. Compute the interface constraint factor, or the average number of interfaces per component, for each sub-circuit as formulated in Appendix A.

7. Plug these values into the modularization function (Equation 3.1) to find out the degree of modularization inherent in the product architecture.

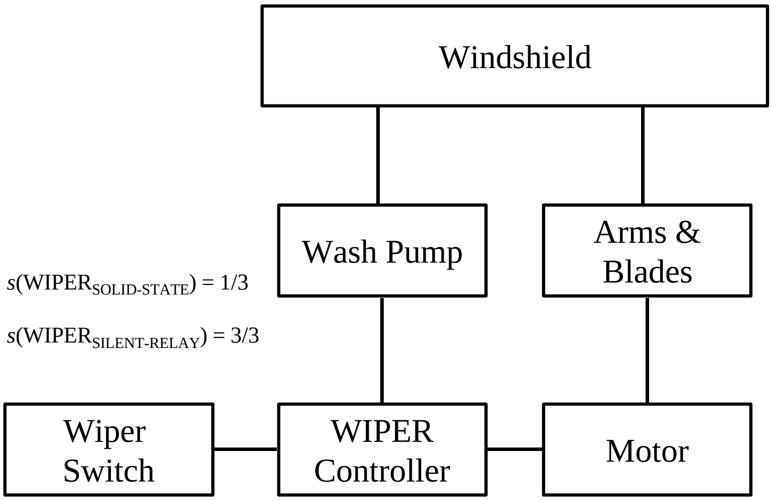
The amount of modularization in a given product architecture is a function of the composition of NTF components, substitutability factor, and interface constraints. The modularization function, *M(u)*, decreases in a non-linear fashion from a perfect-modular architecture (i.e., no NTF components) to a perfect-integral architecture (i.e., no standard components). Refer to Appendix A for the formulation of the modularization function:

$$M(u) = e^{-u^2/2Nsd} \qquad \text{Equation 3.1}$$

$M(u)$ - Modularization function
u - number of NTF components
N - total number of components
s - substitutability factor
d - interface constraint factor

4. Case illustration[8]

The Chrysler Jeeps' windshield wipers controller (WIPER)[9] is a black-box module of which the functional specification was set by Chrysler and the detailed engineering including design and manufacturing was the responsibility of a Fortune-100 OEM supplier. The block diagram of the windshield wipers' sub-system linkages is illustrated in Figure 1.



$s(\text{WIPER}_{\text{SOLID-STATE}}) = 1/3$

$s(\text{WIPER}_{\text{SILENT-RELAY}}) = 3/3$

Figure 1. Block diagram of windshield wipers system.

There were two different technological solutions to the design of the module: 'solid-state' approach and 'silent-relay' approach. The WIPER module used by Jeep models prior to the introduction of Grand Cherokee families applied standard-relay-based technology which made annoying 'clicking' noises when switching from one state to another (e.g., ON and OFF), a feature that Chrysler wanted to get rid off with the new family of Jeeps. During the first attempt to defeat the 'clicking noise', a 'solid-state' approach was applied with the use of only transistors and electrical

---

[8] All the information presented in this study are the results of the author's direct involvement as the design team leader responsible for the product design, pre-production, and sourcing tasks of the WIPER. The interpretation of the data is solely the responsibility of the author.

[9] For a more thorough description of this case with respect to technological solutions, modular innovation and supplier-buyer interdependence, see Hsuan (1999a).

components. The product architecture of solid-state WIPER is consisted of the following sub-circuits: power supply, timer and enabling circuitry, oscillator, charge pump, short circuit protection, and driver circuitry (as shown in Figure 2).

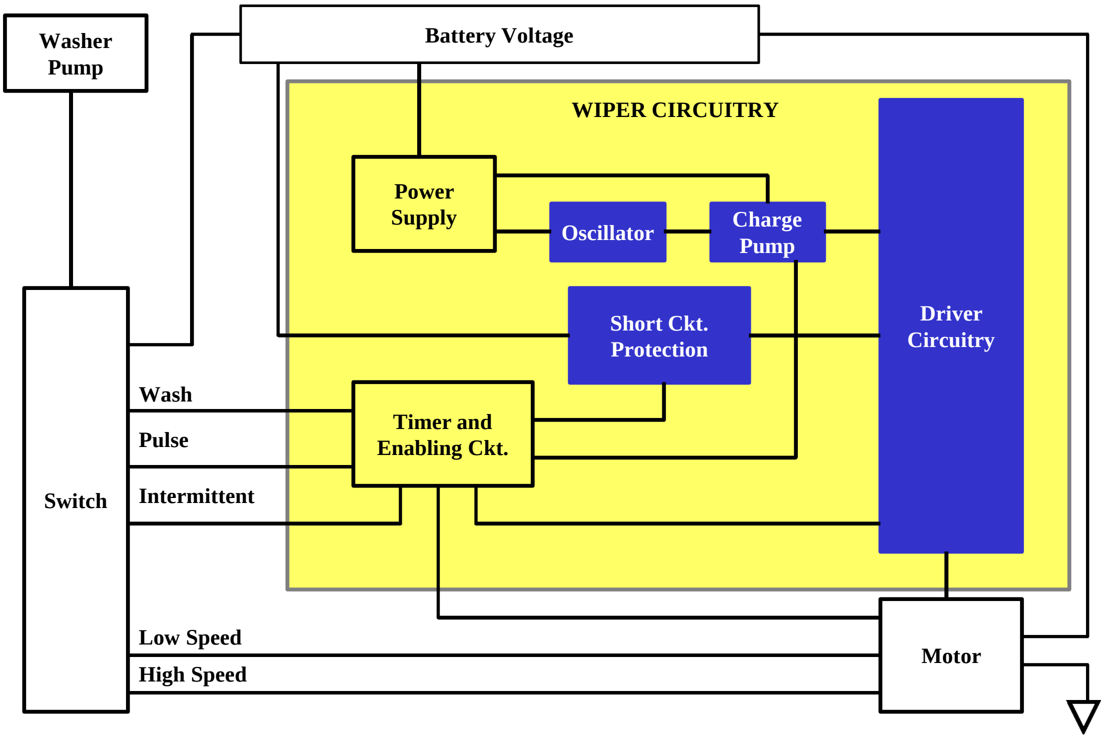


Figure 2. Product architecture of solid-state WIPER.

After almost a year of development, the ‘solid-state’ concept was a failure, contributed by the insufficient knowledge about the interface constraints shared between the WIPER with the rest of the windshield wiper’s system. As Jeep Grand Cherokee was a new family of vehicles with many new technologies incorporated into it, not all the dynamics shared among the components, modules, and sub-systems were well understood. During the second attempt, a totally new innovation was developed to create the ‘silent-relay’ WIPER. In an effort to minimize design and manufacturing changes, ‘silent-relay’ and peripheral circuits replaced a portion of the solid-state WIPER. Although the changes were not drastic, nevertheless the relationships shared among the components and respective sub-circuits and interfaces were altered (as shown in Figure 3).

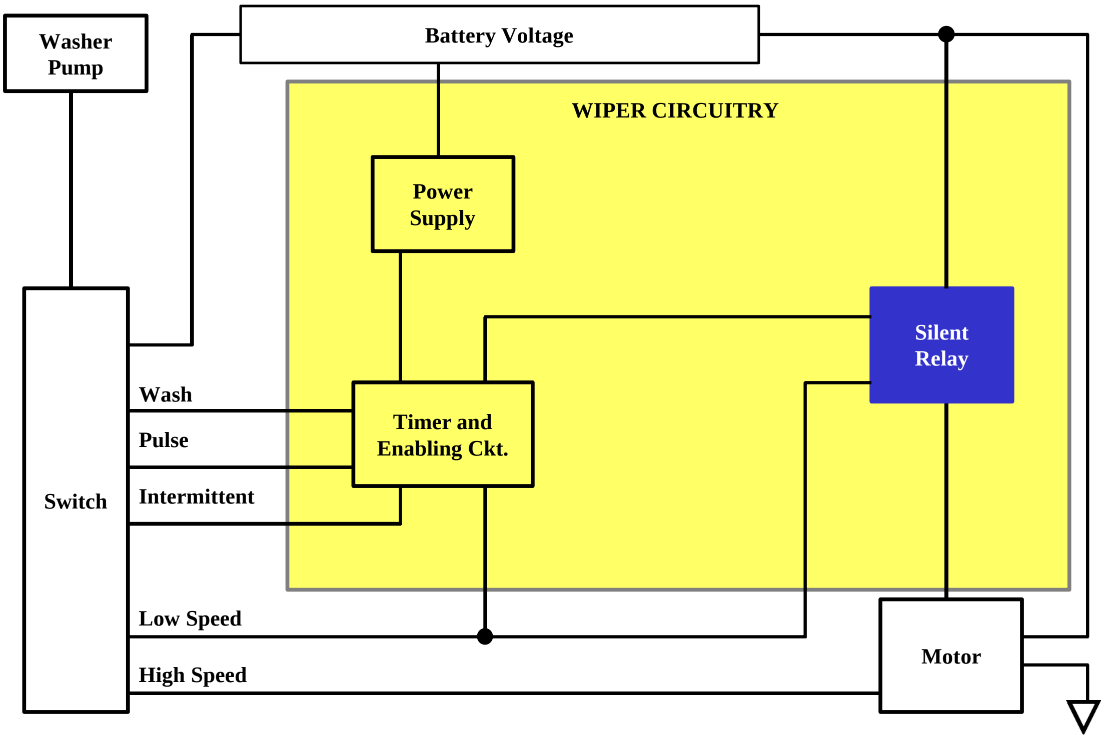


Figure 3. Product architecture of silent-relay WIPER.

In order to get a 'feel' for how components and respective linkages interact with one another to form the sub-circuits, we need to take a closer look at the constituents of individual sub-circuits, at the detailed product architecture level. In doing so, we will find that technical functionality of each sub-circuit is enabled by the discrete components and respective linkages. For example, the power supply sub-circuit[10] is comprised of three standard components (R1, C1 and VR1) with specific interfaces (as illustrated in Figure 7) in order to deliver proper oscillator and charge pump output signals from a common battery voltage. Notice that the Power Supply sub-circuit requires three linkages for solid-state WIPER versus two linkages for silent-relay WIPER.

---

[10] The configuration of such sub-circuit is considered a standardized design with high reusability across other circuit designs.

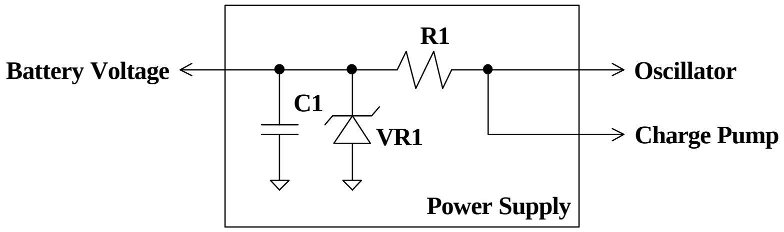


Figure 4. Schematic of power supply circuit.

The electronic portion of the WIPER architecture (Level 1), for both the solid-state and silent-relay modules, share the following relationship with mechanical components (Level 2), as shown in Figure 5. Following the analysis of interface constraints described in Appendix B, and applying to all sub-circuits of both solid-state and silent-relay WIPERs, we find that $d_{solid\text{-}state}$ and $d_{silent\text{-}relay}$ values are 9,85 and 9,94 respectively (see Appendices C.1 and C.2 for the computations).

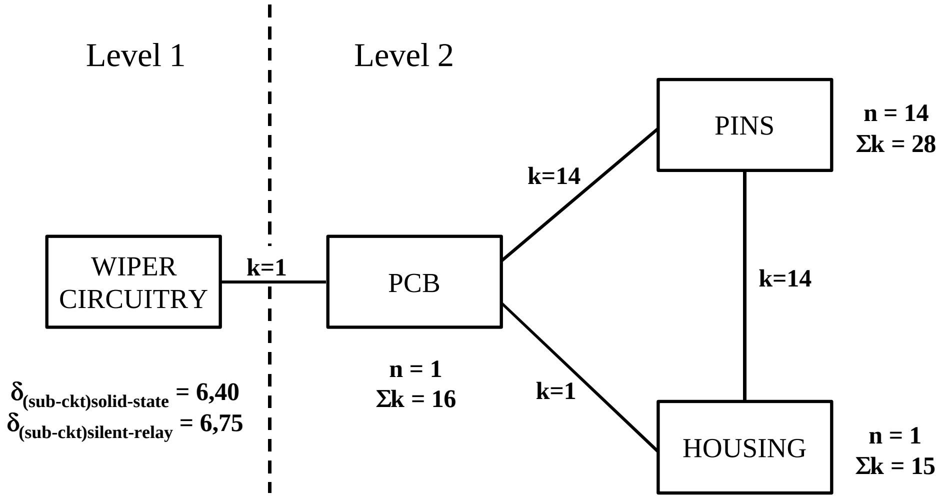


Figure 5. WIPER's relationship with other components.

The WIPER controller module requires three immediate linkages for functionality: wiper switch, wash pump, and motor. While the solid-state WIPER is only compatible with Grand Cherokee Jeeps (substitutability factor, $s = 1/3 = 0{,}33$), all three families of Jeeps (Grand Cherokee, Cherokee, and Wrangler) can use the silent-relay WIPER ($s = 3/3 = 1$). The solid-state WIPER has 60 components (N=60), of which 19 (u=19) are NTF components, yielding a NTF component ratio b of 0,317 (b=19/60=0,317). Similarly, silent-relay WIPER has 57 components with 17 NTF components, translating to a value of 0,298 for b.

Now we are able to find the values for the modularization functions:

Solid-State WIPER	**Silent-Relay WIPER**
u = 19 components	u = 17 components
N = 60 components	N = 57 components
s = 0,33 components/interface	s = 1,00 components/interface
d = 9,85 interfaces/component	d = 9,94 interfaces/component
b = 31,7 %	b = 29,8%
$M_{solid\text{-}state} = 0{,}40$	$M_{silent\text{-}relay} = 0{,}77$

Graphically, the modularization functions for both WIPERs are shown in Figure 6.

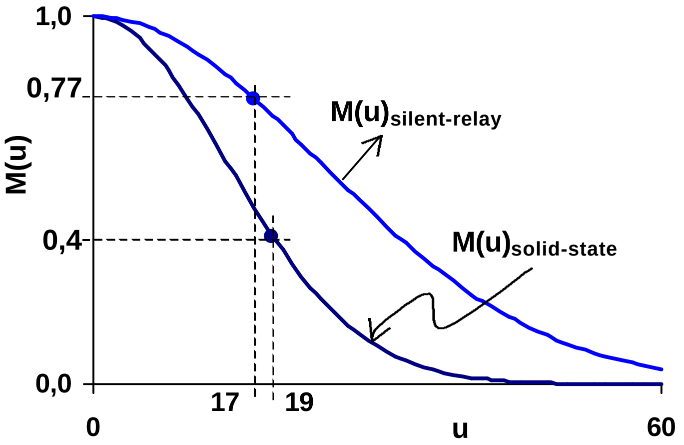


Figure 6. Modularization functions for solid-state and silent-relay WIPERs.

The silent-relay WIPER has a higher degree of modularization ($M_{silent\text{-}relay}$ = 0,77) than the solid-state WIPER ($M_{solid\text{-}state}$ = 0,4). Given the relatively similar values of interface constraints ($d_{solid\text{-}state}$ = 9,85; $d_{silent\text{-}relay}$ = 9,94), the main factor that made the silent-relay WIPER more modular is attributed to its higher substitutability factor and lower NTF component composition. Notice how the modularization gap increases as the number of NTF component increases, implying that product architectures can achieve higher levels of modularity by reducing the number of NTF components. Similarly, modularity can also be improved by designing product architectures with higher substitutability factor, if the NTF component composition remains constant.

5. Conclusion and discussions for future research

This paper discussed product architecture design in new product development and its impacts on modularization and interface management, at the detailed engineering design level. Issues related to modularization focused on the interfaces shared among components in a given product architecture, the specification and standardization of these linkages to allow for greater substitutability, reusability and commonality sharing of components among product families. It was argued that certain degree of complexity of modularization of product architectures could be captured by looking at the composition of new-to-the-firm (NTF) components and how these components are linked to the rest of the components, modules and sub-systems. The relationships shared among NTF components and standard components define the degree of modularity of a given product architecture, assuming that any product architecture range from being perfect modular (no NTF components) to being perfect integral (no standard components). Substitutability factor also plays an important role in the degree of modularization of product architectures, as they are a function of the number of product families made possible by the modular components as well as the number o f interfaces required for functionality.

A simple mathematical model, termed modularization function, was formulated and derived to estimate the degree of modularization in a given product architecture. The modularization function indicated the degree of modularity of a product architecture with respect to the composition of unique components, substitutability factor, and interface constraints shared among components. From a system level's perspective, the modularization function also implied that degree of modularization of a given product architecture can be leveraged with the number of unique components and the degree of substitutability of modules and sub-systems. The case illustration and brief validation of the modularization function with Chrysler Jeeps windshield wipers controller revealed that the main factor that made the silent-relay WIPER more modular is attributed to its higher substitutability factor and lower NTF component composition.

As the majority of products sold in the market place involve many suppliers with distinctive knowledge and expertise, the design of product architectures should also take into consideration how it impacts the organizational design of NPD tasks vis-à-vis manufacturing design and inter- versus intra-firm learning and knowledge management. Moreover, it has been debated that outsourcing of non-core technical activities are enabled by the standardization of these non-core components with respect to the core technology. Can decisions regarding to product architecture designs provide us insights to strategic decisions regarding outsourcing, manufacturing, and supply chain management? If so, how should firms design its organization to match such strategies with respect to its suppliers and customers? Other areas of great interest for research include, for example, the impacts of product architecture design choices (e.g., multiplexing and de-integration of components) with respect to postponement and mass customization strategies.

APPENDIX A - MODULARIZATION FUNCTION FORMULATION

The NTF component composition of a given product architecture, *b*, can be represented by:

$$b = \frac{n_{NTF}}{N} = \frac{u}{N} \qquad ; \qquad 0 \leq b \leq 1 \qquad \text{Equation A.1.}$$

b = 0 represents a perfect-modular product architecture

b = 1 represents a perfect-integral product architecture

Given the range of component composition defined by Equation A.1., it is reasonable to assume that there is a relationship between modularization and the number of NTF components. In other words, it is expected that the degree of modularization, *M*, decreases at a rate, *r*, that is proportional to the amount of modularization present with each set of NTF components, *u*. If *M* is amount of modularization present in a given product architecture with any set of NTF components *u*, then as the number of NTF components vary, the amount of modularization will have changed by the amount of $\Delta M = rM$. In other words, for any unit change of NTF components (Δu=1), the corresponding amount of modularization change ΔM is proportional to the initial amount of modularization. From this, it seems plausible that a similar relation should hold for the decrease in any the amount of modularization in any set of NTF components; that is, the decrease of modularization should be proportional to the change in the number of NTF components as well as the initial amount of modularization.

$$\Delta M = (-rM)\Delta u \qquad \text{or} \qquad \frac{\Delta M}{\Delta u} = -rM$$

The factor *r* is the NTF component ratio per the total interface constraints in a given product architecture. Since a given product architecture may generate many family variations, the interface constraint factor is magnified by substitutability factor, s.

Thus, the factor *r* is represented as:

$$r = \frac{b}{sd} = \frac{u/N}{sd} \qquad \text{Equation A.2}$$

Thus,

$$\Delta M = (-rM)\Delta u = \left(-\frac{u/N}{sd}\right)M\Delta u$$

In differential equation form,

$$\frac{dM}{du} = -\frac{u}{Nsd}M \qquad \text{or} \qquad \frac{dM}{M} = -\frac{u}{Nsd}du$$

For any constant *r*, the solutions to the above differential equation are of the form:

$$M(u) = M_0 e^{-u^2/2Nsd}$$

It is assumed that the amount of modularization is constraint by interface compatibility factors introduced by the NTF components in a given product architecture, thus the amount of modularization *M* in a perfect modular product architecture is when there are no NTF components (*u*=0), hence the initial condition of $M(0) = M_0 = 1.0$.

Consequently, the modularization function is represented as:

$$M(u) = e^{-u^2/2Nsd} \qquad \text{Equation A.3.}$$

Variables:

M(*u*)	-	Modularization function
u	-	number of NTF components
N	-	total number of components
s	-	substitutability factor
d	-	interface constraint factor

APPENDIX B - INTERFACE CONSTRAINT FACTOR ESTIMATION

Interface constraints of a given product architecture are represented in terms of the number of interfaces shared per component, interfaces shared per module, or interfaces shared per sub-system. The analysis can furthermore be carried out at two levels of analysis[11]. Level 1 analyzes the modularization of in the electronic portion of the product architecture (or the circuit design), and Level 2 analyzes the modularization of the circuit design in relation to mechanical portion of the product architecture.

Level 1: A given product architecture is decomposed into I number of sub-circuits so that components and respective interfaces can be analyzed individually at each sub-circuit levels. Then, an interface constraint value, d_i, defined as the number of interfaces per number of components in a sub-circuit, can be obtained:

$$d_i = \frac{\sum k_c}{n_c}$$

a) With I sub-circuits, the aggregate value of all interface constraints from sub-circuit components, $d_{components}$, can be approximated as the average of all d_i, that is,

$$d_{components} = d_{average} = \frac{\sum_{i=1}^{I} d_i}{I} \qquad \text{I = number of sub-circuits}$$

d_c represents aggregate interface constraint value of components *within* sub-circuits (e.g., components within modules). The next step is to evaluate the interface constraints shared *among* the sub-circuits (e.g., modules within sub-systems), $d_{sub\text{-}ckt}$, represented by the number of interfaces shared by a sub-circuit ($k_{sub\text{-}ckt}$) per the number of sub-circuits, I, or

---

[11] This type of analysis fits best for electrical products of which electronic and mechanical components are clearly delineated such as coffee machines, mobile phones, automotive components, personal computers, etc.

$$d_{sub-ckt} = \frac{\sum k_{sub-ckt}}{I}$$

b) The interface constraint factor of the electronic portion of the product architecture is, then, the sum of the interface constraints created by the components within the sub-circuits and interface constraint existent among the sub-circuits.

$$d_{level1} = d_{components} + d_{sub\text{-}ckt}$$

Level 2: The modularization of the mechanical portion of the product architecture is evaluated in the same manner as Level 1. In Level 2 analysis, d_{level1} is treated as an input to the final interface constraint factor calculation of the product architecture.

APPENDIX C.1. - INTERFACE CONSTRAINT FACTOR FOR SOLID STATE WIPER, $d_{SOLID\text{-}STATE}$.

SOLID-STATE WIPER

Component Level 1

Sub-Circuit	**Component**	k_c	Σk_c	n_c	$d_i = \Sigma k_c / n_c$	$k_{sub\text{-}ckt}$	l	$d_{sub\text{-}ckt} = \Sigma k_{sub\text{-}ckt} / l$
	R1	2						
Power Supply	VR1	2	6	3	2	3		
	C1	2						
Oscillator			16	4,50	3,56	2	6	3,83
Charge Pump			10	4,00	2,50	4		
Short Circuit			20	7,75	2,58	3		
Driver Circuit			16	7,00	2,29	4		
Enabling Circuit			44	17,75	2,48	7		
		$N_{electronic}$ =		44				
		$\delta_{component} = \delta_{avg}$ =			**2,57**			
		$\delta_{sub\text{-}ckt}$ =						**3,83**
		$\delta_{level1} = \delta_{component} + \delta_{sub\text{-}ckt}$ =						**6,40**

Component Level 2

		k	**n**	**δ**
Sub-Circuit				6,40
PCB		16	1	16
Pins		28	14	2
Housing		15	1	15
	$N_{mechanical}$ =		16	
	$N_{solid\text{-}state}$ =		**60**	
	$\delta_{solid\text{-}state}$ = avg(δ)=			**9,85**

APPENDIX C.2. - INTERFACE CONSTRAINT FACTOR FOR SILENT-RELAY WIPER, $d_{SILENT\text{-}RELAY}$.

SILENT-RELAY WIPER

Component Level 1

Sub-Circuit	**Component**	k_c	Σk_c	n_c	$d_i = \Sigma k_c / n_c$	$k_{sub\text{-}ckt}$	l	$d_{sub\text{-}ckt} = \Sigma k_{sub\text{-}ckt} / l$
Power Supply	R1 VR1 C1	2 2 2	6	3	2	3	3	4,33
Timer & Enabling Circuit			79	35	2,26	6		
Silent Relay			9	3	3,00	4		

$N_{electronic}$ = 41

$\delta_{component}$ = avg(δ_c) = 2,42

$\delta_{sub\text{-}ckt}$ = 4,33

$\delta_{level1} = \delta_{component} + \delta_{sub\text{-}ckt}$ = 6,75

Component Level 2

	k	**n**	δ
Sub-Circuit			6,75
PCB	16	1	16
Pins	28	14	2
Housing	15	1	15

$N_{mechanical}$ = 16

$N_{silent\text{-}relav}$ = 57

$\delta_{solid\text{-}state}$ = avg(δ) = 9,94

REFERENCES

Baldwin, C.Y. and Clark, K.B. (1997) “Managing in an Age of Modularity,” Harvard Business Review, (September-October), 84-93.

Christensen, C.M. and Rosenbloom, R.S. (1995) “Explaining the attacker’s advantage: technological paradigms, organizational dynamics, and the value network,” *Research Policy*, **24**, 233-257.

Feitzinger, E. and Lee, H.L. (1997) “Mass customization at Hewlett-Packard: the power of postponement,” *Harvard Business Review*, (Jan-Feb), 116-121.

Fulkerson, Bill (1997) “A response to dynamic change in the market place,” *Decision Support Systems*, **21**, pp. 199-214.

Gilmore, J.H. and Pine, B.J. (1997) “The Four Faces of Mass Customization,” *Harvard Business Review*, (Jan-Feb), 91-101.

Gooley, Toby B. (1998) “Mass Customization: How logistics makes it happen,” *Logistics Management & Distribution Report*, (April 1st).

Henderson, R.M. and Clark, K.B. (1990) “Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms,” *Administrative Science Quarterly*, **35**, 9-30.

Hsuan, J. (1999a) “Impacts of supplier-buyer relationships on modularization in new product development,” *European Journal of Purchasing and Supply Management*, **5** (3/4), 197-209.

Hsuan, J. (1999b) “Modularization in New Product Development: A Mathematical Modeling Approach,” *Copenhagen Business School Working Paper 99-4,* June.

Kotha, S. (1995) “Mass Customization: Implementing the Emerging Paradigm for Competitive Advantage,” *Strategic Management Journal*, **16**, 21-42.

Langlois, R.N. and Robertson, P.L. (1992) “Networks and innovation in a modular system: Lessons from the microcomputer and stereo component industries,” *Research Policy*, **21**, 297-313.

Lee, H. (1998) “Postponement for mass customization : Satisfying customer demands for tailor-made products,” in *Strategic Supply Chain Alignment: Best practice in supply chain management*, Gattorna, J. (ed.), Gower Publishing Limited, Chapter 5, pp. 77-91.

Lundqvist, M., Sundgren, N. and Trygg, L. (1996) “Remodularization of a Product Line: Adding Complexity to Project Management,” *Journal of Product Innovations Management*, **13**, 311-324.

Meyer, M. H. and Lehnerd, A. P. (1997) *The Power of Product Platform: Building Value and Cost Leadership*. New York, NY: The Free Press.

Meyer, M.H. and Utterback, J.M. (1993) “The Product Family and the Dynamics of Core Capability,” *Sloan Management Review,* (Spring), 29-47.

Mikkola, J.H. (2000) “Modularization Assessment of Product Architecture,” in the *Proceedings of DRUID’s Winter Conference 2000*, Hillerød, Denmark, January 6-8, Vol.2.

Pine, J. (1993) *Mass Customization – The New Frontier in Business Competition*. Boston, MA: Harvard Business School Press.

Robertson, D. and Ulrich, K. (1998) “Planning for Product Platforms,” *Sloan Management Review,* (Summer), 19-31.

Sanchez, R. (1994) “Towards a Science of Strategic Product Design: System Design, Component Modularity, and Product Leveraging Strategies,” University of Illinois at *Urbana-Champaign Working Paper No. 94-0124*.

Sanchez, R. (1996) “Strategic Product Creation: Managing New Interactions of Technology, Markets, and Organizations,” *European Management Journal*, **14** (2), 121-138.

Sanchez, R. and Mahoney, J.T. (1996) “Modularity, Flexibility, and Knowledge Management in Product and Organisation Design,” *Strategic Management Journal*, **17**, (Winter Special Issue), 63-76.

Schaefer. S. (1999) “Product design partitions with complementary components,” *Journal of Economic Behavior & Organization*, **38**, 311-330.

Tushman, M.L. and Rosenkopf, L. (1992) “Organizational determinants of technological change: toward a sociology of technological evolution,” *Research in Organizational Behavior*, JAI Press, Greenwich, CT, 311-347.

Ulrich, K. (1995) “The role of product architecture in the manufacturing firm,” *Research Policy*, **24**, pp. 419-440.

Ulrich, K.T. and Eppinger, S.D. (1995) *Product Design and Development*. McGraw-Hill, New York.

van Hoek, R.I., Vos, B. and Commandeur, H.R. (1999) “Restructuring European Supply Chains by Implementing Postponement Strategies,” *Long Range Planning*, **32** (5), 505-518.