



RIETI Discussion Paper Series 07-J-047

# 人工物の複雑化とものづくり企業の対応 —制御系の設計とメカ・エレキ・ソフト統合—

藤本 隆宏  
経済産業研究所



Research Institute of Economy, Trade & Industry, IAA

独立行政法人経済産業研究所

<http://www.rieti.go.jp/jp/>

人工物の複雑化とものづくり企業の対応  
—制御系の設計とメカ・エレキ・ソフト統合—

2007年11月  
東京大学大学院経済学研究科  
ものづくり経営研究センター  
RIETI  
ファカルティフェロー

藤本隆宏

---

## 要約

本稿では、現代における「製品の複雑化」という問題を、設計論の観点から探索的に考察する。具体的には、企業が市場に供給する製品を、「人工物」（設計されたもの）と解釈し、それが複雑化・簡素化する諸要因と企業の対応について分析する。

一般に顧客の要求機能や社会的な制約条件(環境・安全対応など)が高度化・複合化すると、モジュラー化による対応は難しくなり、製品はインテグラルかつ複雑なものになりやすい。

こうした「製品＝人工物」の複雑化に対して、企業は複数の補完的なアプローチで対応している。①まず、製品アーキテクチャのモジュラー化が複雑化に対抗する有力な手段とされるが、何らかの理由で徹底したモジュラー化が難しい製品の場合、②従来型の、実物試作により機能検証を行う試行錯誤的な製品開発の能力を高めることに加えて、③開発支援 I T（例えば 3 次元 C A D など）を活用した試行錯誤的なデジタル開発、④機能のばらつきが少ない構造設計を効率的に探索する品質工学、⑤そしてリアルタイムで目標機能の実現を保証する電子制御系、などを補完的に組み合わせることによって、人工物の複雑化に対処しようとしている。⑥さらに、電子制御系そのものが複雑化する場合、それを構成する電気設計（エレキ）系の C A D、⑦あるいは組込みソフトウェアの設計を支援するモデル・ベース開発なども援用される。

このように、複雑化する人工物を正確に機能させるには、電子制御系の発展が必須だ。とりわけ、被制御系の機構部品(メカ)が多く残り、結果としてメカ・エレキ・ソフトが共進化する自動車のような製品の場合、被制御系であるメカ設計と、制御系であるエレキ・ソフト設計の間の相互協調が要求される。しかし上記 3 つの設計系は、歴史的な経緯などから、機能設計重視か構造設計重視か、論理記号重視か物理記号重視かなどに関し、異なる設計風土を持つ傾向がある。また、もともと被制御系の人工物であった自動車の場合、設計者間に「メカ>エレキ>ソフト」という力関係が見られる。

こうした分化傾向に対抗してメカ・エレキ・ソフト設計の統合化を進めるには、設計活動の源流にある「自然言語による人工物の表現」を精緻化し、開発上流におけるメカ・エレキ・ソフトのチームワーク設計を促進する新たな I T を模索するなど、開発上流における統合化の努力が極めて重要である。

---

## 1 はじめに：人工物の複雑化と企業の対応

### 1.1 問題設定：人工物の複雑化

本稿の目的は、現代における「製品の複雑化」という問題を、設計論の観点から探索的に考察することである。具体的には、企業が市場に供給する製品を、「人工物 (artifact)」すなわち「設計されたもの」と解釈し、人工物が複雑化する、あるいは簡素化する諸要因について、おもに人工物の制御という観点から考察する<sup>1</sup>。

**人工物と設計プロセス**：製品、すなわち企業が供給する財・サービスを人工物と見る、ということは、製品を「設計」という観点から見直す、ということの意味する（藤本、2004；藤本・東京大学ものづくり経営研究センター、2007）。そして製品＝人工物の「設計」とは、一般に人工物システムの機能要素と構造要素を分割し連結することを指す。つまり、そのシステムの利用者にとって意味のある機能要素群を特定し、同じシステムを分解可能な構造要素群（部品、モジュールなど；Simon, 1969）に切り分け、しかる後に機能要素間、構造要素間、そして機能要素群と構造要素群の対応関係を決める。以上を、製品の生産・販売・使用に先立って構想することを、設計というのである。それは、一群の要求機能を満たす構造要素パラメータ群の値を求める、一種の連立方程式のようなもので近似的に表現できる（Suh, 1990）。

$$f = A \cdot s$$

$f$ ：機能要素のベクトル  
 $s$ ：構造要素のベクトル  
 $A$ ：機能・構造の対応関係を示すマトリックス

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & & \\ \vdots & & \ddots & \\ a_{m1} & & & a_{mm} \end{bmatrix} \quad f = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_m \end{bmatrix} \quad s = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_m \end{bmatrix}$$

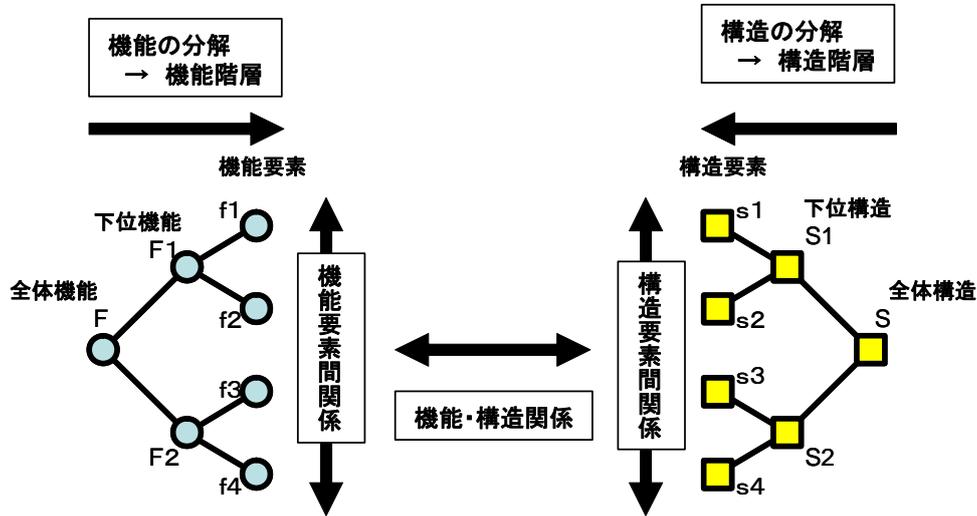
一方、システムの複雑化とは、システムを構成する要素の数と、要素間の関係の数が増加することを意味する。したがって「製品＝人工物」が複雑であるとは、当該製品の機能要素（要求仕様など）が多い、構造要素（部品など）が多い、機能要素間の対応関係が多い、構造要素間の対応関係が多い、そして構造要素と機能要素の間の対応関係が多い、のいずれか、あるいはすべてを意味する。

このように、製品＝人工物の複雑性をその機能・構造の分解・結合関係によって表現することを、アーキテクチャ（基本性能の構想）という（図 1）。

---

<sup>1</sup> その理論的な考察については奥野他（2006）、実証的な応用については、例えば上野・藤本・朴（2007）などを参照されたい。

図1 人工物のアーキテクチャ（複合階層図による表現）



人工物の複雑化とアーキテクチャ：このように、人工物設計の形式的特性を表現するのが「アーキテクチャ」すなわち基本設計構想である。アーキテクチャは、個々の機能・構造要素の内容は捨象して、要素間の形式的な分割と連結の関係だけに着目する概念である。例えば言語システムで言えば、機能・構造要素の内容を見る語彙（ボキャブラリー）に対して、その間のつながり方に注目する文法に相当するのがアーキテクチャである（Chomsky, 1965）。これがシンプルなものモジュラー型、複雑なものをインテグラル型という（Ulrich, 1995；藤本・武石・青島、2001）（図2、図3）。

図2 モジュラー（組み合わせ）型アーキテクチャ

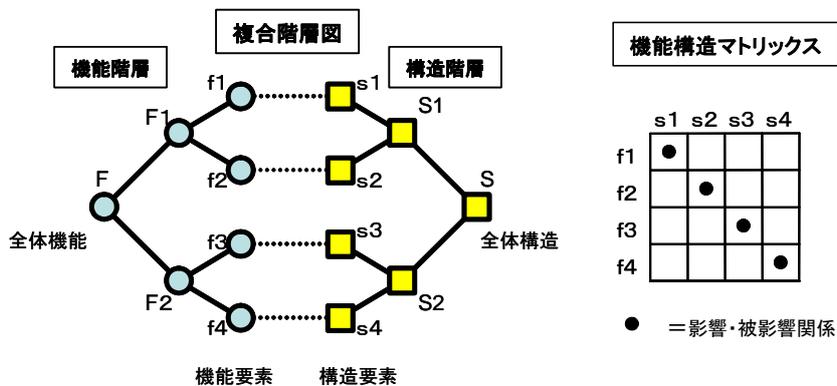
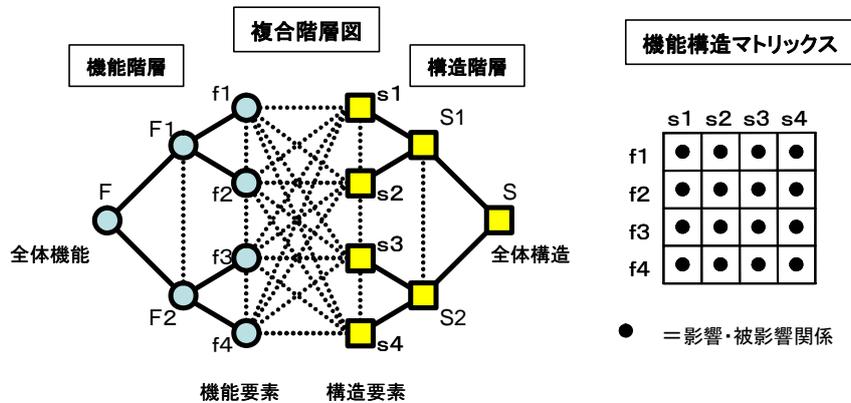


図3 インテグラル（擦り合わせ）型アーキテクチャ



設計プロセスを「機能要素ベクトル＝機能・構造マトリックス×構造要素ベクトル」という連立方程式体系で近似する公理的設計（Suh, 1990）の表現法で見ると、純粋なモジュラー型は機能・構造マトリックスが対角行列になるようなもの、純粋なインテグラル型はマトリックスが非ゼロの値で埋まるようなものを言う（図2、図3）。

モジュラー型

$$A = \begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & & \\ \vdots & & \ddots & \\ 0 & & & a_{mm} \end{bmatrix}$$

インテグラル型

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & & \\ \vdots & & \ddots & \\ a_{ml} & & & a_{mm} \end{bmatrix}$$

**人工物複雑化の諸条件**：以上の観点から、現代において、製品＝人工物が複雑化するのには、以下のようなケースと考えられる。

- (1) 使用者がある製品に多くの機能を同時に要求する（→機能要素数が増加する）。
- (2) 使用者が製品機能間のバランスを厳しく要求する（→機能要素間の関係が緊密化する）。
- (3) 要求機能群を満たすために必要な分解可能な構造要素の数が多い（→構造要素数が増加する）。
- (4) 構造機能間に対するサイズや重量に関する制約が厳しい（→構造要素間の関係が緊密化する）。
- (5) 使用者がある製品に要求する機能のレベルが高い（→機能・構造要素間の関係の数が増加する）。
- (6) ある製品に課される制約条件のレベルが高い（→機能・構造要素間の関係の数が増加する）。

たとえば自動車、とりわけ先進国の小型乗用車は、以上の条件から見ると、「複雑化する人工

---

物」の条件を多く備えていることがわかる。すなわち：

- (1) 使用者は自動車に移動、自己表現（記号）、遊戯、住宅代替といった複合的な機能を期待する。
- (2) 経験を積んだ使用者は走行安定性、乗り心地、居住性、燃費などの高度なバランスを要求する。
- (3) 自動車を構成する構造要素（部品）の点数は減少せず、依然1台あたり3万点程度ある。
- (4) 小型自動車は燃費や利便性の面から重量や容積の制約が大きく、部品間の相互干渉が著しい。
- (5) 先進国の自動車購入者は製品使用経験が長く、高水準の機能・性能を要求する傾向がある。
- (6) 自動車は安全規制、排気ガス規制、燃費規制などの制約が厳しく、この傾向には際限がない。

この結果、自動車は、機能要素（要求仕様）が多く、構造要素（部品）も多く、しかも機能要素間の相互関係、構造要素間の相互干渉、そして構造要素と機能要素の対応関係は複雑である。

**複雑なインテグラル製品としての自動車：**以上のように、自動車は、前述の製品アーキテクチャ論の観点から言うならば、「複雑なインテグラル型アーキテクチャの製品」と分類される。

「製品＝人工物」に要求される機能要素や構造要素の数が増加し、その意味で複雑化せざるをえない場合、その間の相互依存関係をできるだけ切断し、機能・構造関係を1対1のすっきりした形に近づけることが出来れば、人工物の複雑化による作業負荷の増加という問題をある程度緩和できる（Langlois and Robertson, 1992；Baldwin and Clark, 2000；青木・安藤編、2002；奥野・龍澤・渡邊、2006）。例えばN個の機能要求があるならば、N個の機能完結的な構造要素（部品）を対応させるのが、純粋な形でのモジュラー化である。

多様な機能や品種が要求される製品であっても、顧客の要求水準が厳しくなく、形がなく、容量の制約が少なく、軽く、社会空間で消費されず（他人に迷惑をかけず）、エネルギーや環境負荷の問題の少ない製品は、相対的にモジュラー化しやすい。例えば、比較的シンプルなデジタル情報機器はモジュラー型になりやすい。大容量の半導体に支えられ、リアルタイム性を要求されないパソコン・ソフトなどもモジュラー化しやすい。しかし、先進国市場で販売されている小型乗用車は、こうした条件を満たさない。したがって、事前には設計者が最大限のモジュラー・アーキテクチャ化の努力をすとしても、事後的には、市場や技術の要求や制約ゆえに、インテグラル型に落ち着かざるを得ない。21世紀初頭の段階における、自動車に課される安全・環境・エネルギーの制約が厳しくなる現状では、自動車が、デジタル電子機器のように急激にモジュラー・アーキテクチャ化することは考えられない。

かくして、社会的制約や物理的制約から、アーキテクチャのモジュラー化に限界がある製品において、製品への機能的要求が多様化、高度化する場合、その人工物を創造する設計作業が非常

---

に複雑化する可能性がある。複雑な人工物は、複数の設計者が組織として開発するのが通常であるから（Clark and Fujimoto, 1991；藤本・安本編、2000）、人工物の複雑化は、設計組織や設計プロセスの複雑化をも意味する。人間を安全に運ぶための自動車、同じく人体を対象とする医療機器、繊細な製品を加工する生産設備、扱いにくい紙を扱う複合事務機器、極限的な小型化を要求される電子機器、リアルタイムの複合制御を必要とする組込みソフトなどが、本稿で問題になる。

## 1.2 複雑性のもう一つの源泉：人工物の制御

**制御系と制御対象**：人工物の複雑化についての基本的な構図は以上の通りであるが、複雑な人工物システムが変動する環境の中で機能を発揮するためには、「制御」（control）というシステムの動作が必要になる。制御とは、人工物が目的とする機能を発揮するように、その構造を変化させることである。

ここで「制御する側」（制御主体）は、「制御される側」（制御対象）である当該人工物のあるべき機能の姿を記憶し、現在の機能や構造の状態、あるいは環境の状態を認識した上で、情報やエネルギーの形で制御対象に指令を送る。制御主体が人間であり、外から人工物に働きかけるのであれば手動制御、制御系自体が人工物に埋め込まれたものであれば自動制御である（緒方、1970）。現代における製品＝人工物の複雑化は、しばしば、制御対象の挙動の複雑化と連動した、制御系の複雑化を伴うのである（奥野・龍澤他、2006）。

**メカ・エレキ・ソフトによる制御**：製品＝人工物が複雑化し、要求機能が複合化するとき、これを人間による手動制御だけで行うことは難しくなる。そこで、自動制御が発達し、自動制御系そのものが複雑化してくる。

制御系の中には、制御対象であるメカの動作エネルギーを利用する、機構に埋め込まれた「メカ制御系」（いわゆる「からくり」など）、電気回路を用いた「エレキ制御系」、そしてソフトウェアを電子回路に読み込ませることによる「ソフト制御系」がある。自動車のように高度で複合的な制御を要求される人工物の場合、その制御は、人間系に加えて、メカ・エレキ・ソフトが連動する複雑な自動制御系を必要とする。

このように、自動車のような製品が複雑化するということは、制御対象（制御される側）である機構部分の複雑化のみならず、制御系（制御する側）の複雑化を伴う。そして、その制御系・被制御系が、メカ（機構）、エレキ（電気）、ソフトという、ことなる工学的伝統で育った3系統のサブシステムからなる、という点が人工物の複雑化に拍車をかける。仮に2つのシステムの要素数や要素の相互依存度が同等であったとしても、同質の要素からなるシステムと、異質な要素からなるシステムでは、複雑さが異なってくるのである。

**自動車の制御系**：このことは、自動車のケースにも当てはまる。自動車の開発負荷が膨大化しつ

---

つあることが、先進国における自動車ビジネスの大きな挑戦課題となっているが、その原因は、単なる「メカ的な制御対象としての自動車の複雑化」ではない。現代の自動車の制御系が、「メカ・エレキ・ソフト」という、もともと異質な開発風土、技術体系、開発プロセスのもとで育ってきた開発人材の緊密な相互関係・統合化を必要としていることも、21世紀の自動車製品開発組織にとって、大きなチャレンジとなっているのである。

自動車はもともと、メカニカルな製品であった。このため、自動車の製品開発プロセスは、エンジン・ボディ・シャシーを中心に、メカニカルな（機械工学系の）エンジニアリングが主流であり、開発プロセスも、メカ主導を暗黙の前提にしたものであった。

しかしいまや、エレキ・ソフト系のエンジニアリングにかかるコストが、高級車では製造原価の30%に達するといわれる。自動車という人工物の制御における、メカ・エレキ・ソフトのコーディネーション（連携調整）は、今後ますます重要になる。

ところが、メカ・エレキ・ソフト技術者の連携調整に関しては、多くの企業で不協和音が聞かれる。この問題をどう考えるべきか。不整合の背後にあるロジックは何か。安全・環境・エネルギーなどに関する社会の厳しい要求が続く限り、世界の自動車産業にとって、「複雑化する製品・工程・事業への対応」は、最も重要な課題であり続けると思われる。

**開発組織にとっての課題:** このように、消費者のニーズが不確実性・多様性・洗練性を増す中で、現代の製品はますます複雑性を高める傾向がある。かくして、複雑化する製品を短期間に設計・開発することは、現代企業にとってますます困難な作業となりつつある。それでもなお、その開発期間は短縮化を必要とする。

すなわち、前述のように顧客が要求する製品機能の数、それに応じる部品など構造的要素の数、そしてこれら機能要素・構造要素間の相互関係の数が増えることにより、開発作業の手順も増加し、しかもそれらの間の同期化・重複化が要求され、全体として、プロダクト・プロセス双方における複雑化を引き起こしているのである。

とりわけ、多くの機構部品〔メカ〕から構成され、それらが多数の電気回路〔エレキ〕によって統合的に制御され、しかもそうした制御系にソフトウェアが深くかかわるタイプの製品において、制御系・被制御系の双方を含む「製品＝人工物」の複雑化が著しい。

自動車はおよそ3万点の機構部品からなる複雑な製品であるが、近年はその多くが自動制御され、先進国の高級車の場合、回路基盤は数十を超え、組み込みソフトウェアは1千万ステップを超える規模に達する。自動車以外でも、医療機器、半導体製造装置、携帯端末、高級小型デジタル家電、複合型事務機などの中に、同様の複雑な製品が出現しつつある。こうしたタイプの製品においては、単に製品の機能や構造の数や相互作用だけでなく、メカ・エレキ・ソフトという異なるタイプの設計論理が複合化して絡み合っており、このことが、製品および製品開発プロセス

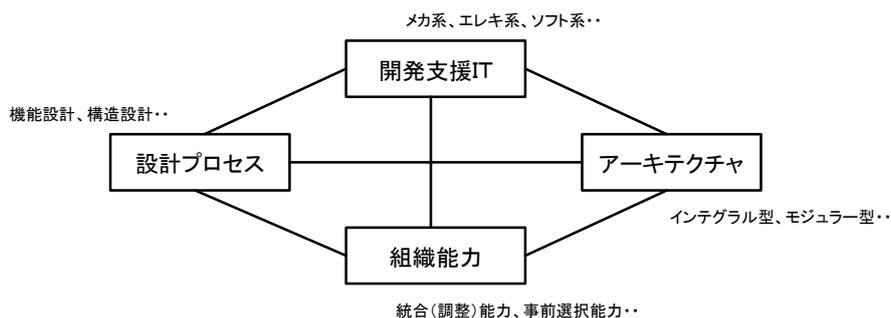
の複雑化に拍車をかけているのである。

そこで本稿では、こうした制御系の複雑化を伴う人工物の分析に関する枠組みを考えてみる。言い換えれば本稿は、精密な実証分析でも解析でもなく、実証分析の前提となる分析枠組みを探る、対策的な試論である。

## 2. 分析枠組—アーキテクチャ・プロセス・IT・組織能力の相互適合—

本稿での分析の前提には、これまでの研究を踏まえた、「広義のものづくり」に関するおおまかな分析枠組が存在する（藤本・東京大学ものづくり経営研究センター、2007）。それは、製品の設計思想（アーキテクチャ）、その製品の設計（開発）プロセス、開発支援 IT、ものづくりの組織能力の間には、一種のダイナミックな相互適合関係（フィット）が存在する、というものである（図 4）<sup>2</sup> 以下において、その構成要素を簡単に説明しておこう。

図 4 分析の枠組み



### 2.1 ものづくり・設計情報・製品

「ものづくり」とは、人工物の設計・開発・生産・購買・販売を通じて顧客満足を生み出す、企業全体・産業全体の取り組みを指す。それは顧客へ向かう「製品設計情報の流れ」を司る汎用技術であり、生産（＝設計情報の素材への転写）のみならず製品開発（＝設計情報の創造）や販売（設計情報の顧客への伝達）も含む<sup>3</sup>。

「製品」は本質的に、顧客を満足させるために企業が作る「人工物」である。「人工物」とは、あらかじめ「設計」されたもの全てを指す。言い換えればそれは、顧客にとって有用な設計情報が、しかるべき媒体に転写されたものである。「人工物」はまた、その媒体の有形、無形を問わ

<sup>2</sup> 藤本（2003）、藤本（2004）、他。

<sup>3</sup> こうした意味における広義の「ものづくり技術」は、2006年に開始された第3次科学技術基本計画における8本の柱の一つに加えられた。

---

ない。従って、「製品＝人工物」とは、物財のみならずサービスも包含する広義の概念である。

このように、広義の、あるいは開かれた「ものづくり」の概念とは、設計情報を創造し、媒体に転写し、出来上がった「製品＝人工物」を顧客に伝達するまでの、「設計情報の流れ」を作る企業活動全体のことである（藤本他、2007）。言い換えれば、「ものづくり」の本質は、「ものをつくる」ことではなく、設計情報を「もの（媒体）につくりこむ」ことである。このように「もの」から「設計情報」へ発想転換することにより、「ものづくり」は工場の生産現場だけに閉じたプロセスではなくなり、むしろ開発・購買・生産・販売の現場が連携し、本社部門も経営トップも、サプライヤーも販売店も顧客も巻き込む、一つの開かれたプロセスとなる。設計情報を創造するのが開発の仕事、創造された設計情報を媒体（もの）に転写するのが生産の仕事、転写する媒体を確保するのが購買の仕事、転写された設計情報を顧客に向けて発信するのが販売の仕事である。そして、顧客はそうした設計情報を企業から受け取り、いわば設計情報を消費するのである（藤本, 2006）。以上のように、ものづくりにおいて最も本質的なのは、「もの＝媒体」ではなく「設計」であるといえる。

## 2.2 設計プロセス

前述のように「設計活動」とは、ある人工物について、その機能と構造の対応関係を、現物の制作に先立って事前に構想することである。

既に図1で示したように、ある「製品＝人工物」を設計する場合、顧客ニーズを反映する製品機能は、複数の機能要素群に分解され、多段階の階層構造として示される。これが機能設計である。同様に製品の構造形状・寸法・材質なども、複数の構造要素群、例えば集成部品・単体部品に逐次的に分解され、多段階の階層構造（例えば部品表＝BOM）として記述される。各階層の部品本体の設計は、製品特殊的、社内共通、業界標準のいずれでもありうる。そして、こうした分解＝階層化によって特定された機能要素数、構造要素数、そして機能要素間、構造要素間、機能・構造要素間の相互依存関係が、「製品＝人工物」の複雑性と関連するのである。

さらに、構造要素間の相互依存関係では、これら構造要素部品・モジュールをつなげる結合部分、すなわち「インターフェース」が定義される。インターフェースの設計は、本体と同様、製品特殊的、社内共通、業界標準のいずれもありうる。

## 2.3 アーキテクチャ〔基本設計構想〕

既に示したように、製品＝人工物における設計情報の諸要素の「切り分け方・つなぎ方」に関する基本設計構想を「アーキテクチャ」という（Ulrich, 1995; 青島・武石, 2001; 藤本, 2001a、他）。その最も基礎的な分類は、前述の「モジュラー型」と「インテグラル型」の区別、また「オープン型」と「クローズ型」の区別である（Ulrich, 1995; Fine, 1998; Baldwin and Clark,

---

2000; 藤本、2001a)。

既に見たように、機能要素と構造要素（部品）の関係が1対1対応に近く、インターフェースも共通化されている結果、既に設計済みの部品を組み合わせれば全体製品の機能を保証できる、というタイプの製品＝人工物が「モジュラー型」、逆に、製品機能要素と製品構造要素の関係が多対多対応で錯綜し、インターフェースも製品特殊である結果、製品ごとに部品を新規に最適設計しないと全体性能が出ない、というタイプの製品＝人工物が「インテグラル型（擦り合わせ型）」である。また、モジュラー型のうち、社内共通部品の組み合わせで全体機能を実現する人工物を「クローズド・モジュラー型」、業界標準インターフェースによって異なる企業の既設計部品を組み合わせることができる人工物を「オープン・モジュラー型」という（藤本・武石・青島編、2001）。

むろん実際の製品は、そう単純には割り切れないことが多く、機能階層・構造階層の中における部位や層によって異なるアーキテクチャ特性を持ちうる（藤本、2003）。実際の製品は、純粋インテグラル型と純粋モジュラー型を両極端とするスペクトル上に連続的に分布すると考えるべきだろう（大鹿・藤本、2006；新宅、2003；新宅・善本・加藤、2004；延岡他、2006）。例えば大鹿・藤本（2006）は、簡単な尺度で製品ごとのインテグラル（モジュラー）アーキテクチャ度を測定した結果として、最もインテグラル度の高い領域に自動車・同部品、最も低い（モジュラー度の高い）領域に電気機器類が分布する傾向を見出している。

一般に、インテグラル型製品は、要求性能や構造・機能のトレードオフなど、制約条件が厳しい中でぎりぎりの構造設計を行う製品である。例えば、市場ニーズを所与とすれば、重量・強度・容積などの制約の厳しい小型のメカ製品や、構造設計・機能設計を連続量で表現する必要のあるアナログ型製品（西村、2004）が、インテグラル（擦り合わせ）寄りである傾向があるとされる。

製品のアーキテクチャ特性は、構成部品の特性によっても推定できる。例えば、インテグラルな組み立て製品は、製品特殊なカスタム設計部品の比率、社内設計基準による部品の比率、アナログ系の部品、そしてメカ系の部品が多い傾向がある。モジュラー型製品はその逆で、業界標準的な汎用部品、社内共通部品、デジタル系の部品、そしてエレキ・電子系の部品の比率が高い傾向があると言われる。例えば、製品原価に占めるエレキ・電子系部品の比率は、PCやデジタル家電では50%以上、高級自動車では30%、低価格自動車では10%とみられる。また同様に、製品原価に占める汎用部品の比率は、PCでは50%以上、白物家電で30%、高級自動車では10%以下といわれる。

## 2.4 設計・開発支援 IT

一方、開発や生産の現場において「良い設計情報の流れ」を作るツールとして、近年存在感を

---

増しているのが、デジタル情報技術、いわゆる IT である。とりわけ、CAD・CAM・CAE といった、いわゆる製品開発支援 IT である。

設計プロセスとは一般に、問題に対する解決案（設計案）を決めるプロセスであり、その主たる要素は、代替的設計案の探索（サーチ）、および各設計案の問題解決力の事前検証（シミュレーション）である。そして、主に設計案のサーチを支援するデジタル情報技術を CAD（Computer-Aided-Design）、設計案の問題解決シミュレーションを支援するデジタル情報技術を CAE（Computer-Aided-Engineering）と呼ぶ。また、特に金型設計を支援する IT を CAM（Computer-Aided-Manufacturing）と呼ぶが、これは金型設計用の CAD と考えてもよいだろう。

ただし、後述のように、同じ開発支援 IT であっても、メカ系、エレキ系、ソフト系の IT は、開発対象の違いを反映して、進化の経緯も基本的な発想も相当に異なるので注意を要する（新木、2005；上野、2005）。

いずれにせよ、開発支援 IT は、単に設計を支援するだけではない。それは、蓄積された電子媒体の設計情報を用いて、製品性能や製造性のシミュレーションも行なう。これにより、実物の試作品による機能や製造性の評価に先立って、バーチャルな問題解決サイクル（設計・試作・実験サイクル）を回すことが出来るようになる。これを、問題解決の「前倒し」という意味で「フロント・ローディング」といい、今日、開発の競争優位を築く決め手の一つと言われている（藤本、1997；Thomke and Fujimoto, 2000）。

## 2.5 ものづくり組織能力

組織能力（Organizational Capability）とは、ある企業あるいは組織に独特の組織ルーチン（繰り返し行動のパターン）であって、他の企業に対して競争力や収益力の持続的優位をもたらすものを指す。それは他社には真似しにくく、継続的に蓄積され、創発的に進化するものである。

これを、前述の広義の「ものづくり」概念に適用するならば、「ものづくりの組織能力」とは、「設計情報の良い流れ」を作る組織能力のことである。すなわち、顧客へ向かう設計情報の創造・転写・発信のプロセスを、競合他社よりも常に正確に（高品質で）、効率良く（低コストで）、迅速に（短いリードタイムで）遂行する組織ルーチンの体系を指す。つまり、いわゆる QCD の同時達成・同時改善を行う能力である。そこでは、開発・購買・生産・販売の現場の組織能力が一体となって緊密に絡み合っている（藤本、2001）。

いわゆるトヨタ生産方式は、こうした「ものづくりの組織能力」の典型である（藤本、1997；藤本、2003；藤本、2004）。すなわち「設計情報の創造・転写が行われぬ時間」を最小化し、顧客へ向かう設計情報の、淀みない「流れ」をつくることがその要諦である。

---

また、製品の設計・開発における組織能力とは、前述のような「問題解決としての設計プロセス」において、他社に対する持続的優位性を持つ組織能力であり、すなわち「組織的問題発見・問題解決の能力」である（Clark and Fujimoto, 1991）。

## 2.6 プロセス・アーキテクチャ・IT・組織能力の相互適合

製品特性、開発プロセス、IT とものづくり組織能力には、短期的には不適合があるとしても、長期的には互いに適合的な方向に共進化する傾向がある。そうした相互適合が競争力、すなわち表の競争力（製品の存続力）および裏の競争力（現場の存続力）をもたらすからである。例えば、戦後の日本企業では、歴史的な経緯により、多能工のチームワークを強みとする「統合型ものづくりの組織能力」が偏在する傾向があったが、これと相性のよい設計思想は設計要素間の複雑な相互調整を要する「擦り合わせ型アーキテクチャ」であり、適合的な IT は協調環境でチームワークを支援する IT であり、フィットしやすい開発プロセスは部門間・部門内のチームワークを要する同時並行開発（サイマルエンジニアリング）・前倒し開発（フロントローディング）・大部屋型開発（コロケーション）などである傾向がある<sup>4</sup>。これらは相互適合的である<sup>5</sup>。

## 2.7 インテグラル型アーキテクチャと設計プロセス

既に述べたように、「製品＝人工物」複雑化の開発負荷へのインパクトがより大きいのは、他の条件を一定とすれば、インテグラル型アーキテクチャの製品である。例えば、自動車などインテグラル型の開発の場合、顧客が要求する機能要素群に対して、新たに構造設計を行う部品が多くなる。しかも、そうした部品間の機能的相互依存性（複数部品が協調してひとつの機能を達成すること）、逆機能的相互依存性（電磁干渉など）、構造的相互依存性（部品配置の相互干渉など）があり、まして、設計の途中で事後的に発覚する相互干渉も少なくない。したがって、個々の部品の構造設計や、部品間の相互依存性のチェックを事前に周到に行うことが重要である。

このように、インテグラル型製品の設計プロセスは、個別部品の構造設計や部品設計間の相互調整に力点が置かれる傾向がある。また、「メカ設計」の開発風土は、こうしたインテグラル型設計プロセスの特徴を反映する傾向がある。

---

<sup>4</sup> 藤本・東京大学ものづくり経営研究センター（2007）他。

<sup>5</sup> また、過去十数年の実証研究では、以上のような枠組を製品設計・開発を支援するデジタル情報技術（IT）に応用することで、以下のような結論が得られた。すなわち、90年代における3次元ソリッドモデルCADなど、先端的な開発支援ITの導入そのものは、産業の競争優位確立にとって必要十分条件とはならない。言い換えれば、ITは、それを使いこなす「ものづくり組織能力」、例えばチームワークによる問題発見・問題解決能力が伴わない限り、競争優位をもたらさない（藤本、1997；藤本・延岡・青島・竹田・呉、2002；Thomke-Fujimoto, 2000）。例えば、まったく同じCADのパッケージを使いながら、結果としての「裏の競争力」では、企業によって大きな差が出ることもある。現場において、電子媒体とヒトの集団は、密接に絡み合いながら一つの設計情報システムを形作っているからである。

---

しかし、これらの新規部品設計をゼロから行うことは、複雑なインテグラル製品の場合、莫大な設計工数につながる恐れがある。そこで、工数節約のために、いくつかの手段が採用される。

例えば部品構造の「編集設計」である。機構部品の場合、新規設計といっても、既存設計に新たな要求水準や制約条件を加えて修正すればよい場合が多い。編集設計は新しいアイデアを抑圧する逆機能も指摘されるが、インテグラル型開発には不可欠なツールである。

一方、「機能要件の絞り込み」も一つの方法である。この場合顧客は、製品差別化にとって必須の機能要件のみを提示する。構造設計者は、その機能要件を満たしつつ、指定されていない「隠れた機能要件」を補完し、顧客に形（構造設計）で示す。それを顧客が承認することで、事後的にすべての機能要件が確定する。つまり、事実上、全機能要件の確定タイミングを遅らせることによって複雑性を部分的に吸収するのである。

このように、製品全体機能が最適設計された部品群に依存する擦り合わせ（インテグラル）型アーキテクチャの場合、製品ごとに新規設計部品を「起こす」こと、つまり最適化プロセスが重要になる。したがって、設計プロセスにおける代替設計案のサーチでは、既存の設計案の探索よりはむしろ、新しい設計案を「起こす」ことが、インテグラル型製品の設計プロセスの特徴になる。

したがって、組織能力としては、リアルタイムで部品設計間の相互調整を行う「統合力」「調整力」がポイントになる。そして、開発支援 IT としては、「新規部品設計の最適化」に力点を置いた CAD が重要になる。後述のような、構造設計重視のメカ系 CAD はこの色彩が強い。

## 2.8 モジュラー型アーキテクチャと設計プロセス

他方、モジュラー型アーキテクチャ製品は、製品を構成する各構造要素（部品）間の機能的・構造的な相互依存性が低いので、仮に、機能向上のために新しい部品を設計しなければならない場合でも、それぞれの部品の設計を他から独立して進行できるので、部品間の設計パラメータの調整負荷はあまりかからない。

また、従前どおりの要求機能で十分な場合は、設計済みの部品を社内・社外の既存部品のリスト（ライブラリー）から選べば済むので、部品の新規設計作業は発生しない。一方、まったく新しい一群のモジュラー型製品の設計プロセスでは、長く陳腐化しない、拡張性の高いインターフェース（デザイン・ルール）を周到に構想することが重要になるので、インテグラル型製品に比べれば、こうした事前の骨格作りにより多くの設計工数が割かれる傾向があろう。

このように、製品の要求機能ごとに機能完結的な部品が 1 対 1 で対応する「モジュラー型アーキテクチャ」の製品は、既に設計済みの部品をカタログから選んで購入し、それらを組み合わせる傾向が強い。つまり、設計代替案のサーチにおいて、既存設計部品の探索が優先される。言い換えれば既存設計部品を「捨てる」こと、つまり選択プロセスが重要になる。組織能力とすれば、

---

製品をあらかじめ機能完結部品群に切り分ける構想力、あるいは素性の良い既存部品を選ぶ「事前の目利き能力」がポイントになる。そして、開発支援 IT としては、「最適の既存設計部品の選択」に力点を置いた CAD が重要になる。電子部品のデータベース（ライブラリ）の整備に力点を置くエレキ系 CAD はこの色彩が強いといえよう。

このように、製品ごとのアーキテクチャ特性によって、開発プロセスや適合的な IT が異なるとすれば、その間の相性（フィット）を考える必要がある。そして、製品ごとのアーキテクチャは、その製品に対する市場ニーズの特性（例えば極限性能が要求されるか、機能間のバランスを重視するか、軽量化・小型化が重要か）、あるいは製品に含まれる技術の特性に影響される。

## 2.9 メカ・エレキ・ソフト設計とアーキテクチャ特性

アーキテクチャ特性と、メカ・エレキ・ソフトの区別は、異なる系統の概念ではあるが、相互に連動する傾向は見られる。つまり、当該製品が機構設計中心のいわゆる「メカ製品」であるか、あるいは電気設計中心のいわゆる「エレキ製品」であるか、あるいはコンピュータが読み込んだソフトが機能実現の鍵を握る「ソフト製品」であるかは、アーキテクチャや設計プロセスに少なからぬ影響を与えるとみられる。

例えば、他の条件を一定とすれば、プリント基板をベースとするエレキ設計は、設計済みのカタログ部品を「拾う」製品設計プロセスを通じて、モジュラー型の製品になりやすい。一方、メカ設計は、新規設計部品を「起こす」製品設計プロセスを通じて、インテグラル型の製品になりやすいと予想される。

ソフト設計については、書き方次第でモジュラーにもインテグラルにも振れる可能性があり、個々のソフトウェア・エンジニアの構想力・設計力に左右されるところが大きい。一般には、制御対象がリアルタイムの制御を要求する、つまり制御の時間制約が厳しい場合はインテグラル寄りの制御ソフト、そうした制約が厳しくない場合には機能分割によるモジュラーよりのソフト設計になりやすいといわれる。

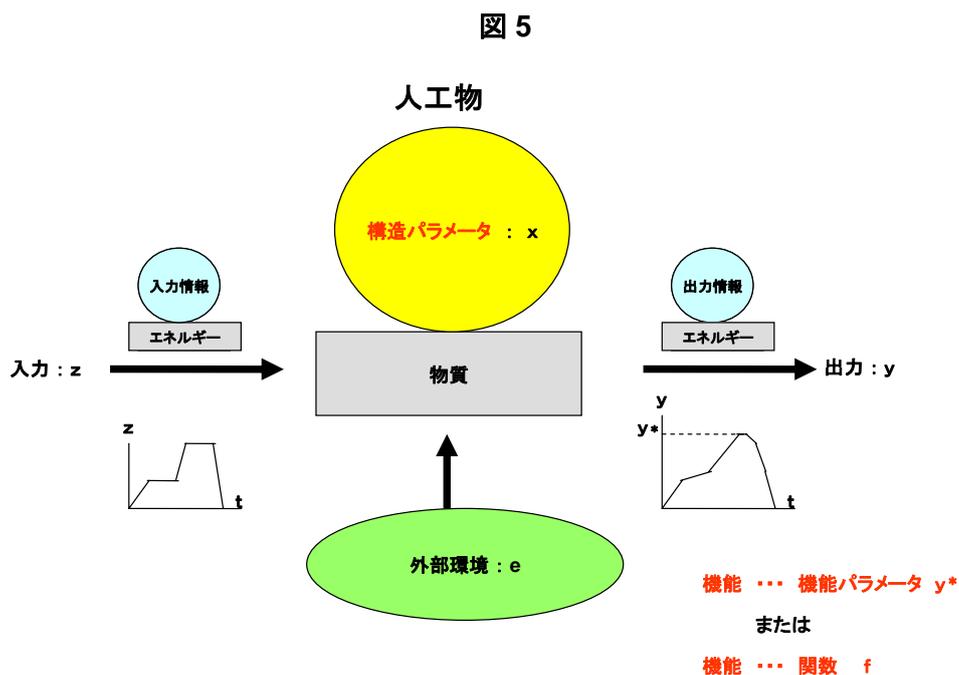
## 3 複雑化する人工物設計への複合的アプローチ：制御論の視点から

### 3.1 人工物制御の基本形：人工物・出力・入力・外部環境

それでは、複雑化する「製品＝人工物」において、ねらった製品機能の確実な発現を保証するために、企業はどのような方策をとるのだろうか。「複雑な人工物の機能保証」という課題については、事前の因果情報を前提にした固有技術論（機械工学、電子工学など）、固有技術論を一般化・抽象化した公理系設計論、実物試作を用いた試行錯誤による開発、デジタル情報技術（IT）を用いた試行錯誤による開発、実験計画を活用する品質工学、自動制御を用いる制御工学など、様々な手法が補完的に使われているのが実態であるが、その間の相互関連性は必ずしも明確に意

識されていない。そこで本稿では、人工物を視覚的なモデルとして表現することで、これらの諸手法の間の関連性について考察しよう。

その出発点は、人工物・外部環境・入力・出力の関係を記述する、制御理論の枠組みを援用することである。その基本形を**図 5**に示す。



**図 5**において、人工物( $x$ )は前述の「開かれたものづくり論」の枠組にしたがい、設計情報が媒体に転写されたものと考えられる。より正確に言うなら、構造設計情報(例えば構造パラメータ群)が物的な媒体に転写されたものが、物財としての製品である。

これに対して、アウトプット( $y$ )は、基本的にはエネルギーという媒体に機能設計情報(例えば機能パラメータの経時変化)が転写されたものと考えられることが出来る。人工物が人間にもたらす効用、すなわち機能は、究極のところは、製品が使用者に対して行う仕事(サービス)であり、それは基本的に、エネルギー媒体に乗った機能設計情報である。要するに、顧客が要求する機能設計情報( $y$ )はエネルギーに転写され、構造設計情報( $x$ )はモノに転写される。

また、インプット( $z$ )とは、アウトプットの制御を意図して外から与えられるエネルギー、およびその操作である。機能の発揮を目的として操作されたインプットは、いわば機能設計情報を内包したエネルギーである。制御理論ではこれを操作量( $z$ )という。

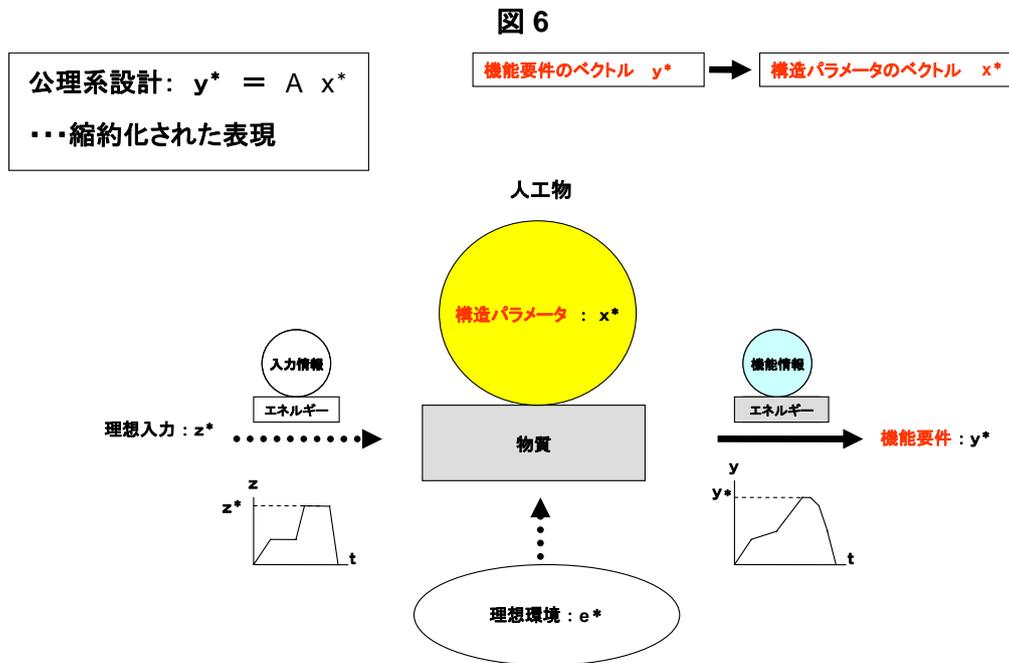
最後に、所与の人工物において、インプットとアウトプットの関係に影響を与える外部の諸力を外部環境( $e$ )という。人工物の構造( $x$ )と出力の経時変化( $yt$ )および入力の経時変化( $zt$ )の関係

は、外部環境( $e$ )の影響を受けることが多い。

このように、人工物=自動車（構造設計情報  $x$ ）は、ある環境条件( $e$ )のもとで、ある操作入力( $z$ )に反応して、使用者が要求する機能（機能設計情報  $y$ ）を発揮（発信）することを期待されている。つまり、人工物における機能保証は、最も基本的な形としては、 $y = f(x, e, z)$ と示せる。

### 3.2 公理系設計論：縮約化された表現

さて、こうした枠組みで考えるならば、公理系設計は、人工物に対する、ある意味で縮約化された表現と考えられる。すなわち、公理系設計は、代表的な環境条件と最適の入力を仮定し、人工物を、その構造要素パラメータ( $x$ )と、代表的な外部環境におけるその機能要素の最適値( $y$ )との間の関数関係で記述する。つまり、 $y = f(x, e, z)$ は、 $e$ と $z$ を所与とした形で、 $y = f(x)$ と縮約的に表現される（図6）。



ここで設計プロセスは、要求仕様  $y^*$  を所与として、あるべき構造パラメータ  $x^*$  を探索する過程であると、近似的に表現されるわけである。

例えば、自動車の燃費性能を考えてみよう。自動車の車体、エンジン、トランスミッションなどの構造設計( $x$ )が決まれば、所与の走行環境 ( $e$ : 路面、勾配、曲がり、標高、ガソリンの質など)、所与の操作入力 ( $z$ : スロットル、ギア選択、ブレーキ) のもとにおける燃費性能 ( $y$ : ガソリン1リッターあたり走行距離など) は推定できるが、その数値はむろん、走行環境や操作入力の違いによって千差万別である。

---

これに対して、例えば日本の自動車業界では、10モードと呼ばれる所定の走行環境・操作入力（平地・平坦・直線舗装路での加速・減速パターン）を前提に、各モデルの燃費性能を測定することが慣例である。つまり、「このクルマ( $x$ )は燃費がリッターあたり20キロメートル( $y$ )だ」といった形で表現する。これは、 $y = f(x)$ という、公理系設計の人工物表現と整合的である。したがって、たとえば「燃費20キロ( $y^*$ )の新型車を開発せよ」という開発命令が経営陣から出れば、設計者は、他の性能要件を所与として、また10モードの走行条件( $e, z$ )を前提に、この性能を満たす構造設計パラメータ( $x^*$ )を探索するのである。

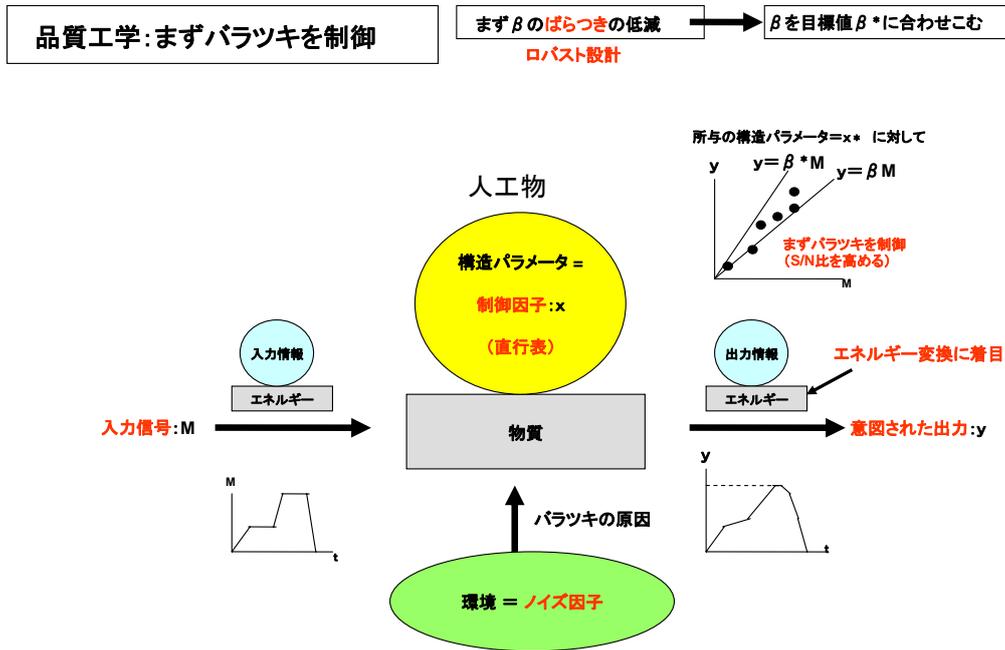
### 3.3 品質工学：機能パラメータのばらつき制御

品質工学（田口、1999）は、田口メソッドとしても知られる、一つの有力な設計パラメータ探索法であるが、その基本的な論理構造は、人工物を制御系としてみる上述の枠組みと親和的であるように見える。

品質工学は独特の用語法を用いるので、他のアプローチとの比較は難しいが、その中核をなすパラメータ設計・ロバスト設計は、基本的には制御理論の枠組と親和的であり、品質工学がエンジニアに人気がある一因とも推測される。

すなわち、品質工学の基本的な人工物表現法によれば、入力信号( $M$ : 本稿では  $z$ ) に対して、さまざまなノイズ因子（本稿の外部環境  $e$  出力に近い）が介在した結果としての出力( $y$ )のばらつき（S/N比）を最小に抑え、かつ、出力の平均値を目標とする出力( $y^*$ )に近づける努力をするために、適切な制御因子（本稿における構造設計パラメータ  $x$  に近い）を探索するのが、パラメータ設計である（**図 7**）。ノイズ因子に対して出力のばらつきが少ないことをロバスト性（頑強性）ということから、これは「ロバスト設計」とも呼ばれる。いずれにしても、目標機能の達成よりも機能のばらつきの制御を先行させ、ばらつきを抑え込む構造設計パラメータの探索を、実験計画法を応用して迅速に行うことが、品質工学の主たる効能の一つである。

図 7



品質工学の基本形では、入力と出力の関係は比例的と仮定され ( $y = \beta M$  ;  $\beta$ は傾き)、この関係あるいは関数 ( $y = \beta M$ ) のことを「基本機能」と呼ぶ (これに対して、公理系設計では、出力  $y$  のことを「要求機能」と呼ぶ)。田口メソッドを創出した田口玄一氏は、人工物の基本機能を見つげるときにはエネルギーの流れに注目せよ、と示唆しているが、これは、人工物の入力、出力がエネルギーを媒体にした機能設計情報である、という前述の考え方も整合的である。

このように、品質工学におけるパラメータ設計の基本形は、「出力= $f$ (制御因子、ノイズ因子、入力信号)」であり、上述の制御アプローチにおける  $y = f(x, e, z)$  と同形である。複雑な人工物に対して、フィードバック機構などを用いてリアルタイムで出力 (機能) のばらつきを抑え込もうとするのが自動制御の考え方だとすれば、品質工学とは、構造設計パラメータ最適化によって、ばらつき制御を、構造の中に、いわば事前に埋め込んでしまおうとする試みだと解釈することができよう。

### 3.4 伝統的な機械設計：制御対象のメカ設計

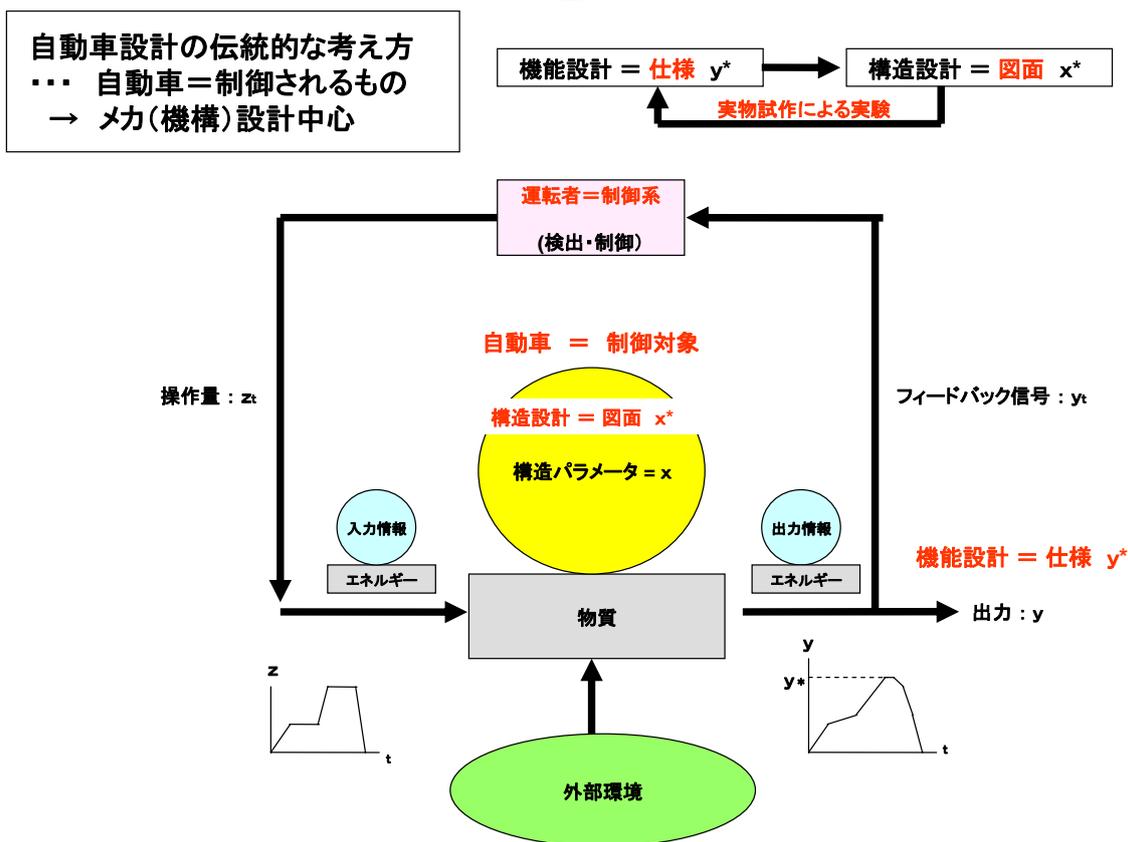
次に、機械工学における伝統的な機構設計の考え方をみておこう。もともと機械とは、機構 (メカニズム) に埋め込まれた構造設計パラメータ ( $x$ ) によって入力 ( $z$ ) と出力 ( $y$ ) の関係を統御するシステムであり、とりわけ入力・出力が人力以外の、たとえば水力、蒸気力、火力、電力などであるものを指す。しかし、自動制御は機械の要件ではない。伝統的な機械は、人間による手動制御が基本である。産業史においても、イギリス産業革命の「機械化」 (mechanization) とは、人力から蒸気力などへのエネルギーの切り替えを意味していたのに対し、20 世紀における「自動

化」(automation)は、制御情報を担う媒体を人の頭脳から他の媒体に移すメディアの切り替えであり、両者は本質的に異なる。

例えば自動車は、20世紀前半には、もっぱら人間が操作する機械であった。言い換えれば、自動車はもともとメカニカルな「制御対象」つまり「制御される側」であり、その制御系(制御する側)は運転者自身、つまり人間に他ならなかった。自動車は、自動制御機械ではなく、手動制御機械だったのである。

したがって、伝統的な自動車設計は、運転者の操作( $z$ )に対して、ある環境( $e$ ; 例えば路面状況や燃料属性)のもとで、あるべき反応( $y$ )をするような、メカニカルな構造設計パラメータ( $x$ )を探索することであった。またその開発では、テストドライバーが運転者( $z$ )、テストコースが環境( $e$ )、試作車が構造設計( $x$ )をシミュレーションする実物における機能検証が中心であった(図8)。

図 8



以上をまとめるならば、20世紀前半の伝統的な自動車の設計は、ほとんどがメカニカルな人工物の設計プロセスであり、それは本質的に「制御対象(制御されるもの)の設計」であった。制御するのはもっぱら運転者であり、制御する側の人工物設計、すなわちエレキやソフトによる制御系の設計は、この段階では視野に入っていなかったのである。

---

そして、自動車の開発が元来「メカニカルな制御対象の設計」であったがゆえに、基本的には構造設計重視につながった。すなわち、「人間がうまく操作すれば狙った機能が実現するような構造設計を行え」というのが伝統的な自動車設計の考え方であるから、あるべき機能を決める機能設計は比較的シンプルでよかったが、構造設計はそれなりに精密なものが要求されたのである。

こうした、伝統的な機械設計における構造設計重視は、いわゆる「図面主義」と呼ばれる傾向にもつながったと考えられる。図面主義とは、人工物の設計において最も重要な情報は、人工物の形状・寸法を2次元で図形表現した構造設計情報、すなわち図面だ、という考え方である。あるいは、所与の人工物の製作に必要な情報は図面によって完全に記述できる、という考え方である。この発想のもとでは、図面の構造情報さえ完全であれば、その機能はおのずと推定できるとして、図面が重視される。要するに、「機能設計（仕様）は簡潔に、構造設計（図面）は精密に」ということである。

### 3.5 現代の機械設計：CADに残る構造設計重視の伝統

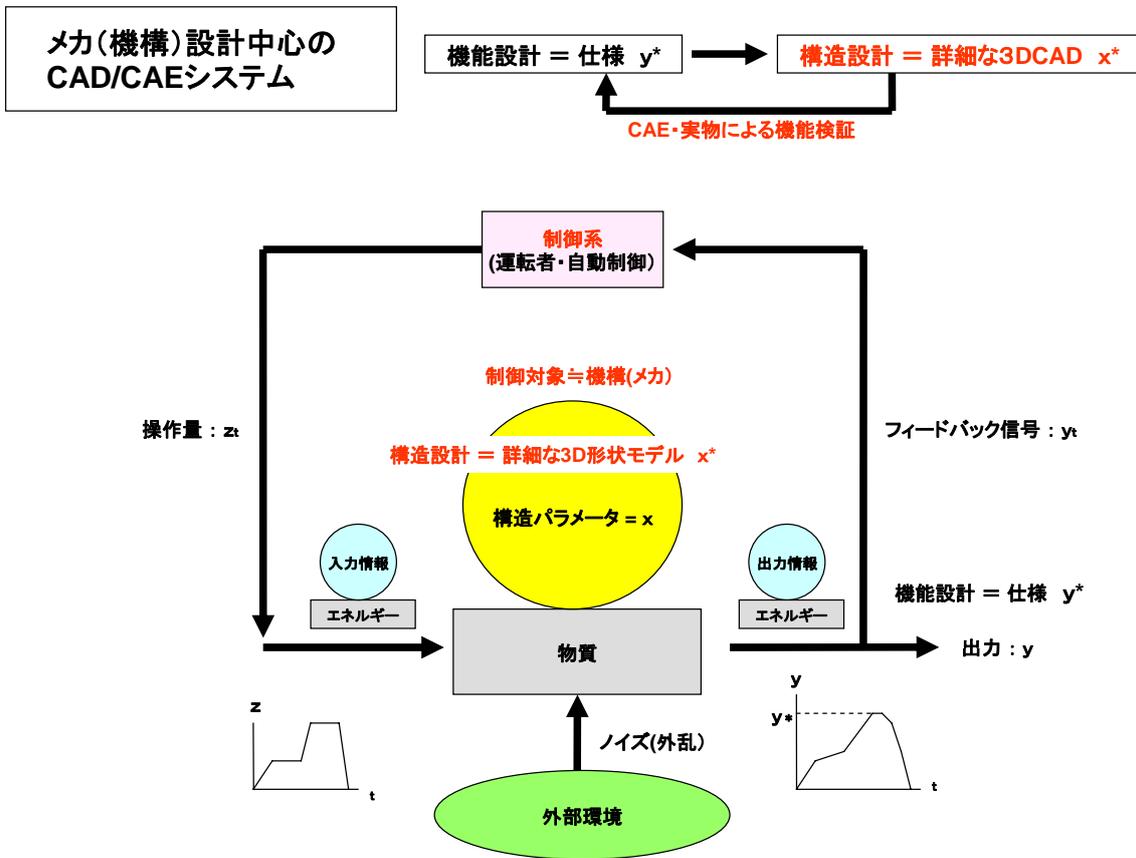
現代の人工物設計では、デジタル情報技術（IT）による開発支援が普及している。例えば、製品の構造設計情報を電子媒体上で表現するCAD（コンピュータ支援設計）、構造設計情報をもとに機能検証（シミュレーション）を電子媒体上で行うCAE（コンピュータ支援エンジニアリング）、構造設計情報をもとに金型などの工程設計情報を電子媒体上で表現するCAM（コンピュータ支援マニファクチャリング）などである。CADは、もともとは2次元の図面情報の電子媒体化から始まったが、近年は3次元の構造設計情報を電子化した3次元CADが普及している。

このように、機械設計ではデジタル情報技術の発展が著しいが、それは、機械設計の伝統をある意味で継承しており、したがって、現代の3次元CAD主体の設計にも、構造設計重視の発想が色濃く反映されていると考えられる（図9）。

例えば、現代の自動車設計、とりわけ制御対象である機構（メカ）部分の設計では、自動車の各部品を表裏面と質量を伴う3次元形状で表現できるソリッドモデルによって、精密に表現しようとする流れが一部で顕著である。「製品構造の完全な3次元表現さえできれば、あとはシミュレーション（CAE）や精密な実物試作によって、機能の検証と改善は後からいくらでもできる」という考え方がその背後にありそうである。

かくして、「機能設計は簡潔に、構造設計は精密に」という、機械（メカ）設計の伝統は、ソリッドモデルによる先端的なデジタル・エンジニアリングに継承されている。そしてこれは、後述のように、機能設計の精密な記述を重視する制御系設計の考えかたと、ある意味で対照的なものである。

図 9



### 3.6 機械の電子制御化とエレキ・ソフト設計

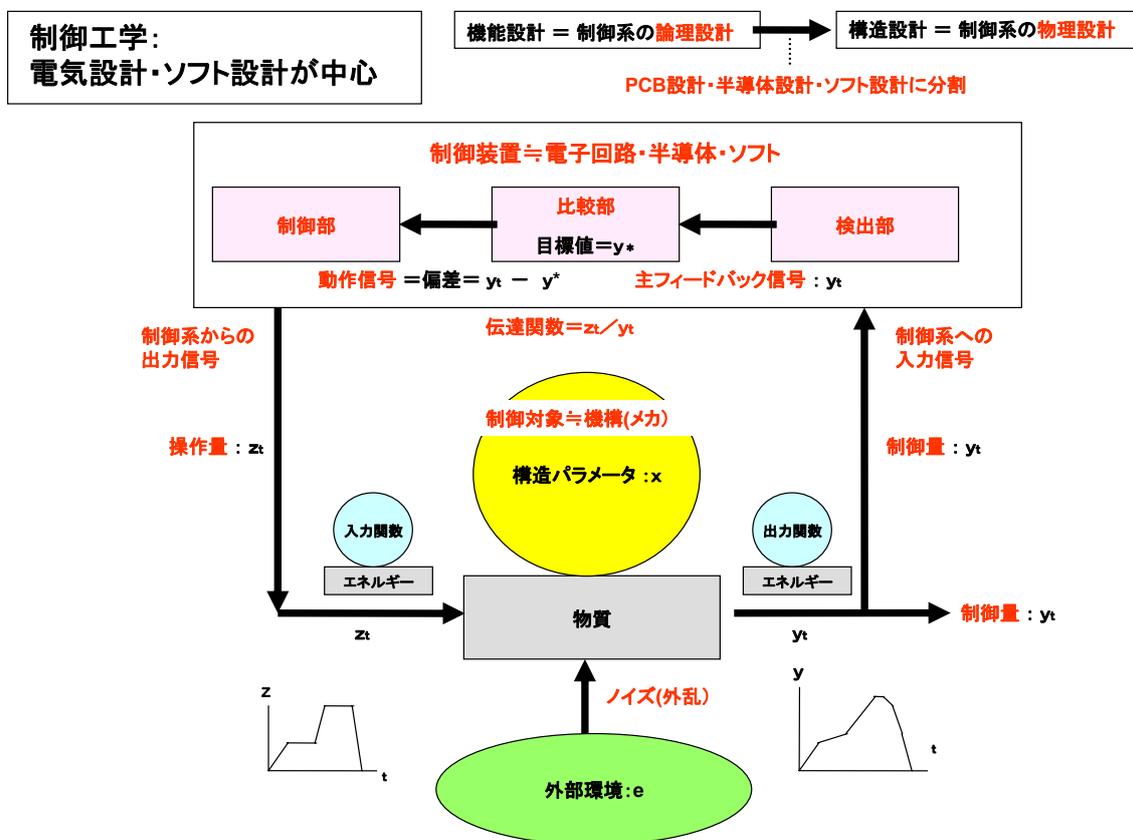
しかし、20 世紀後半、とくに最後の四半世紀には、メカ的な人工物の電子制御化が急速に進んだ。エレクトロニクスとメカニクスの融合という意味で、この傾向はしばしば「メカトロニクス」と呼ばれる。その背景には、コンピュータなどの電子技術の発展もあったが、同時に、機械システムに対する機能的要求や社会的制約の高まりを指摘できる。

例えば自動車の場合、安全、環境、燃費、利便性などに対する市場および社会の要求が高まるとともに、1970 年代ごろから、自動車の電子制御化が急速に進んだ。当初は、一部の機能部品（例えば燃料噴射装置）に、部品単体ごとに電子制御装置が付く程度であったが、21 世紀に入るあたりから、これらの電子制御装置が車載ネットワークでつながった車両統合制御へ向かい、自動車の電子制御系は急速に複雑化した。

つまり、自動車は、手動制御のメカ製品から、急速に、手動・自動混合型の人工物へ変化してきた。そしてこれに伴い、電子回路（エレキ）およびソフトウェアからなる電子制御系が急速に複雑化していった。人工物におけるエレキ設計・ソフト設計の比重増加とは、このように、本質

的には自動制御系の拡大を意味している（図 10）。

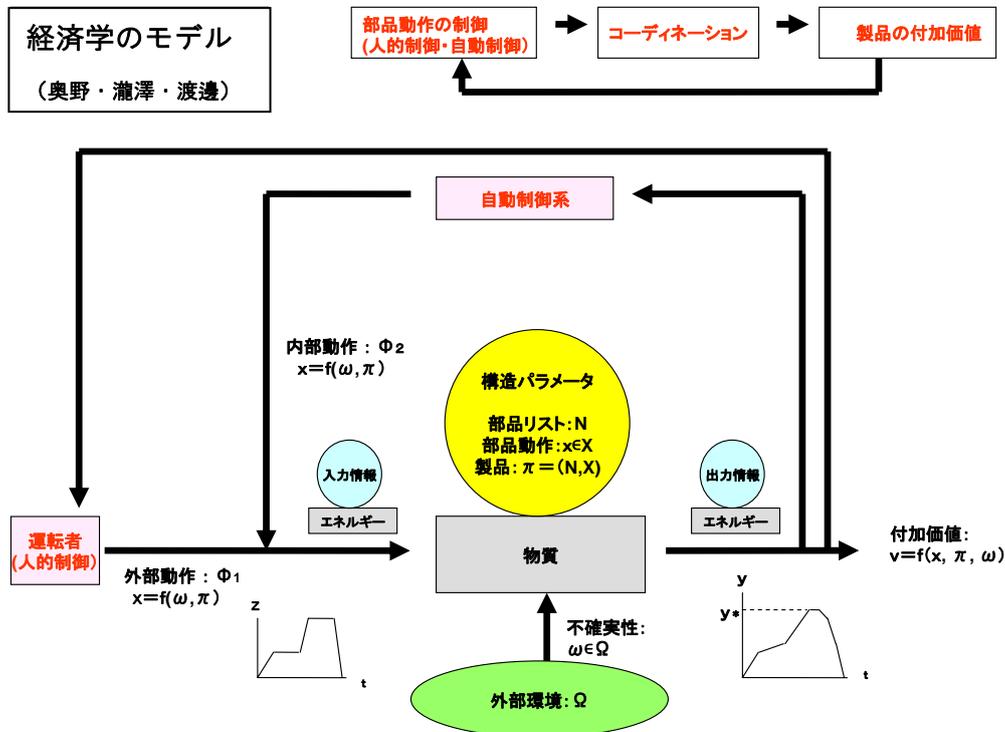
図 10



一般に「制御」(control)とは、ある目的に向けて制御対象(機械など)に操作を加えることである(澤井監修・緒方、1970)。したがって、制御すべき機能を、事前に明確に記述する必要がある。つまり、機能設計を周到に行う必要がある。制御工学では、人工物の振る舞い、働き、出力、すなわち機能は、例えばブロック図・フロー図・状態遷移図などで表現することになる。

このように、制御部分の設計を担当する制御工学の発想では、人工物の入力(操作量;  $z$ )と出力(制御量;  $y$ )のあるべき姿の明確な記述が、何よりもまず重視される。重要なのはあくまでも  $y = f(z)$  であり、構造パラメータ( $x$ )は、この入力・出力関係の実現に貢献する限りにおいて重視される。つまり、制御工学を支える電気設計・半導体設計・組込みソフトウェア設計などでは、機能設計が重視される傾向がある。なお、奥野・瀧澤・渡邊らによる人工物の経済モデルは、こうした制御系としての特性を抽出しており、ここでの分析と親和的である(図 11)。

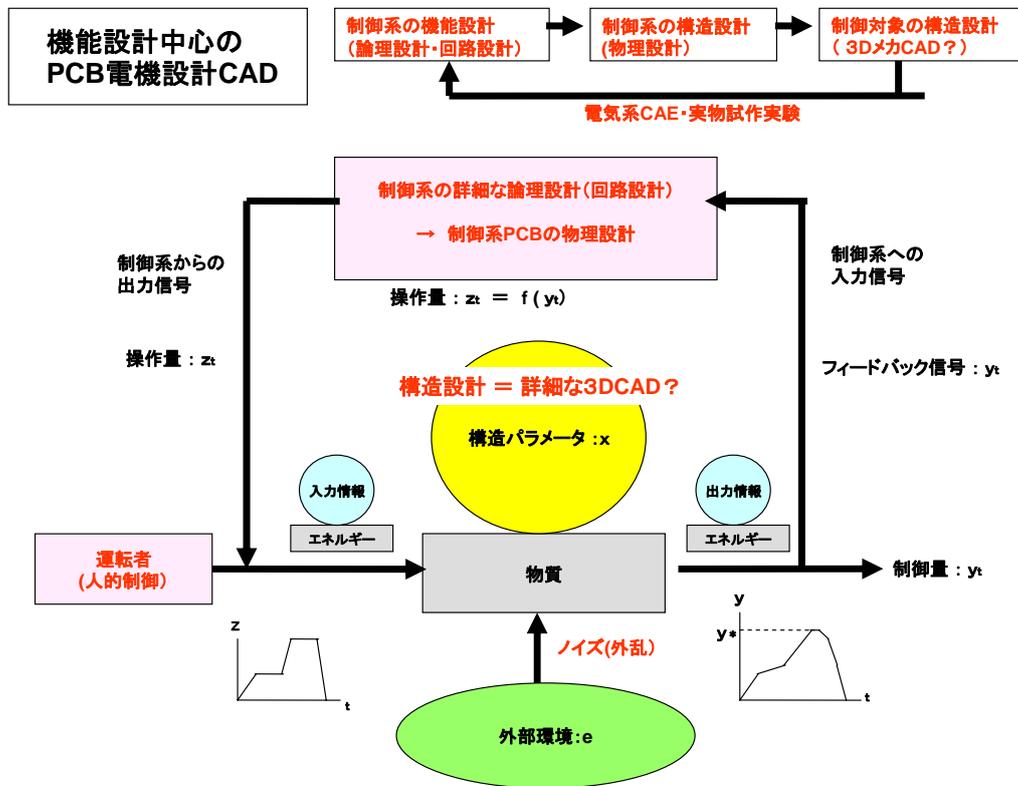
図 11



### 3.7 電気設計におけるデジタル情報技術の利用

エレキ系の設計プロセスにおいても、メカ系同様、デジタル情報技術 (IT) による開発支援ツールが発達している。しかし、この分野における開発支援 IT の発達の態様は、制御系を中心におくエレキ系と被制御系が中心のメカ系では、かなり異なる (上野、2005；新木、2005；上野・藤本・朴、2007)。すなわち、エレキ系の設計支援 IT は、制御系の設計プロセスを前提にした、機能設計重視・論理設計重視という、制御系の設計の特性を色濃く反映しているように見える (図 12)。要するに、エレキ・メカ設計の比重拡大は、制御系の設計の拡大に他ならず、それは機能設計重視のカルチャーを持つ。例えば、制御工学では、構造設計は、ある環境・入力のもとで所定の機能をもたらす関数 (たとえば伝達関数) で示される。製品構造は関数によって抽象化されるわけであり、形状・寸法・材質といった構造情報を図面や 3 次元 CAD で精密に再現することは、メカ設計の場合ほどには重要ではないのである。

図 12

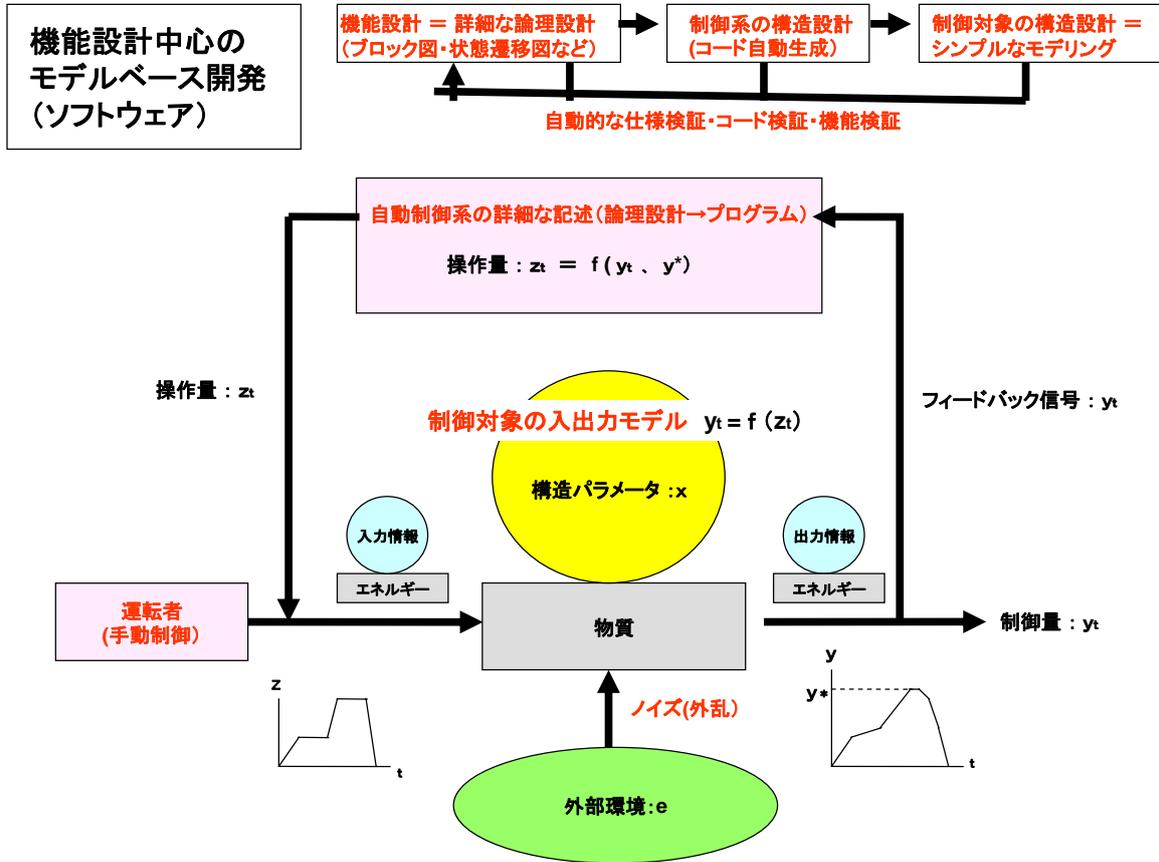


### 3.8 ソフトウェア設計におけるモデル・ベース開発

ハードウェアの振る舞いを制御する「組み込みソフトウェア」(embedded software)も、本質的に「制御系」の設計であり、したがって機能設計重視の傾向が顕著である。すなわち、まず要件の定義を明確にし、制御対象である人工物の振る舞い、すなわち機能の詳細な記述が要求される。ソフトウェア工学において、顧客の要求機能を明晰に記述する要求工学が重視されるのは、その一つの現れである。一方、いったん機能が精密に定義されれば、ソフトウェアの構造設計に当たるソースコードは、自動生成もある程度可能である(酒井、2006)。

つまり、ソフトウェアのモデル・ベース開発は、①詳細な機能設計(論理設計)、②自動化の進んだ制御系の構造設計(コード生成)、③シンプルなモデリングによる制御対象の構造設計を内容とする、機能設計重視の体系だといえる(図13)。

図 13



### 3.9 小括：人工物複雑化に対する複合的なアプローチ

ここまでの議論をまとめておこう。人工物に対する機能要件や制約条件が厳しくなる中で、自動車・デジタル機器・精密機械などの機械システムの多くが電子制御を伴いつつ複雑化し、企業の製品開発に大きな負荷をもたらしている。

こうした趨勢に対して、企業は複数の補完的なアプローチで対応しているように見える。すなわち、まず、①製品アーキテクチャのモジュラー化が複雑化に対抗する有力な手段とされるが、何らかの理由で徹底したモジュラー化が難しい製品の場合、②従来型の、実物試作により機能検証を行う試行錯誤的な製品開発の能力を高めることに加えて、③開発支援 IT (例えば 3 次元 CAD など) を活用した試行錯誤的なデジタル開発、④機能のばらつきが少ない構造設計を効率的に探索する品質工学、そして⑤リアルタイムで目標機能の実現を保証する電子制御系、などを補完的に組み合わせることによって、人工物の複雑化に対処しようとしている。さらに、⑥電子制御系そのものが複雑化する場合、それを構成する電気設計 (エレキ) 系の CAD、あるいは、⑦組み込みソフトウェアの設計を支援するモデル・ベース開発なども援用される。

---

従来、上記のような諸アプローチ間の補完関係は、必ずしも明確に意識されておらず、それぞれが独自の有効性を主張する傾向があった。しかしながら、人工物の複雑化が急速に進んでいる領域では、複数の対策を補完的に応用する、複合的なアプローチが今後は必須と思われる。

さらに、設計対象物である人工物は往々にして、機構部分、電子回路制御部分、ソフトウェア部分が複合的に絡み合った複雑な「メカ・エレキ・ソフト複合製品」となっている。その製品開発プロセスでも、メカ設計者・エレキ設計者・ソフト設計者の間の緊密な連携が要求されてきている。

しかし、いまだこの問題に、体系的にアプローチした経営学的な研究はほとんどない<sup>6</sup>。メカ・エレキ・ソフトのアーキテクチャおよび設計プロセスの共通点・相違点について、設計論の基本に戻った考察が、今後の研究課題の一つである。

## 4. メカ・エレキ・ソフト設計の特性

以上のように、複雑化する製品＝人工物が安定した機能を発揮するためには、多くの場合、それらが自らの制御系を発達させる必要がある。その結果、主に制御対象の側ではメカ設計（機構設計）の複雑化、そして制御系の側ではエレキ（電気）設計およびソフト設計の複雑化が起こる。つまり、単に機能数、部品数やその間の相互作用の数が増えるだけでなく、エレキ設計・メカ設計・ソフト設計という、歴史も設計風土も異なるサブシステムが連動するという意味で、人工物は複雑化するのである。

そこであらためて、メカ設計、エレキ設計、ソフト設計の特性の違いについて簡単に整理しておこう。

### 4.1 メカ設計の特性

人工物のメカ（機構）設計について改めて整理するなら、以下のような傾向を持っているといえる。

**制御される側の設計**：メカ設計は、「制御される側」すなわち制御対象の設計であることが多い。この点、エレキ設計やソフト設計が基本的に「制御する側」の設計であるのとは対照的である。メカ設計の対象は、例えば駆動、躯体、外装など、人工物が実際に機能を発揮する部分である。そうしたメカ部分の仕事は、連続的に動く、つまりアナログ的なものであることが多い。

**先行者としての発言力**：歴史的にメカ設計は、自動制御が登場する以前からの長い歴史があり、企業の開発組織内でも、エレキ設計やソフト設計より、先行者としての権力や影響力を持つ傾向

---

<sup>6</sup> 例えばITの関連では、メカ系について新木（2005）、エレキ系について上野（2005）、メカ・エレキ・ソフトの連動については富士通・日本発ものづくり研究会（2007）など、実務系の著者による論考が既にある。しかし、設計論・製品開発論の基本に戻った学術的な分析はまだ始まったばかりである。

---

がある。とくに、もともと自動制御系を持たなかった機械、例えば自動車においてそうである。

**メカのアーキテクチャの傾向**：メカ（機構）設計とは、重量、容積、強度などの物理的制約条件の下で所定の機能を達成するための機構部品群の設計である。質量・体積のある部品が機械的に連結し、自身の強度を保ち、かつスペース面の部品干渉を避けながら、部品間で運動エネルギー等のやりとりをする。このため、特に小型化・軽量化の条件が厳しい製品において、部品間の構造的な相互依存性が高まる。そうした物理的制約条件は個別製品ごとに異なる傾向があるので、新製品ごとに各部品を新規設計する傾向が高まる。また、外観の意匠性も重要な機能的要素となることが多い。これらの結果、とくに顧客の機能や意匠に対する要求水準が高い場合、メカ製品はインテグラル型アーキテクチャになりやすい。

**精密な構造設計**：前述のように、メカはその重量、寸法、形状などの微妙な違いが機能に影響する。このため、製品機能の事前検証（実験評価）を正確に行うためには、詳細な構造設計が必要になる。そうした詳細な構造設計をもとに、実物試作やコンピュータ・シミュレーションを行い、それを用いて要求機能を事前検討するのである。したがって、メカ設計では、開発のかなり早い段階で、詳細な構造設計が要求される。メカの場合、正確な機能検証は、精密な構造設計を要求する。

**簡素な機能設計**：メカの機能は、通常はエネルギーの流れを伴う仕事である。したがって、いったん機能（＝仕事）が顕在化すれば、それを数値で測定したり、言葉で表現したりすることは比較的容易である。逆に言えば、要求機能は、数値や自然言語で事前に設計することができる。

一方、メカの場合、機能設計情報が完備していなくても、詳細な構造設計は可能である。ある人工物にとって必要不可欠な機能要件のみを示せば、とりあえずその要件を満たす構造設計は可能である。しかし、その構造設計で試作と実験を行えば、さまざまな環境下で予想されなかった挙動を示す。隠れていた機能要件がそこで顕在化し、いわば機能設計の事実上の拡充が事後的に行われる。

実際、メカ設計では、事前に示す機能要件（要求仕様）は絞り込む方がよい、という考え方が強い。当初の機能設計は簡素でよい、という考え方である。

**機能設計と構造設計の混在**：このように、メカ的人工物の場合、いったん構造が定まれば、その機能（ふるまい）は外から観察や測定をしやすいが、逆に、所与の構造がもたらす機能をすべて事前に予測することは難しい。そこで、必要不可欠と考えられる機能要件をまず確定し、直ちに構造設計・試作・実験を行い、そこで顕在化する機能をあとから補足する。詳細な機能設計の完成は、事実上、構造設計の後になる。

こうした特性ゆえに、自動車などのメカ設計では、開発の早い段階から、機能設計に加えて、概略の構造設計（レイアウト）や概略の意匠設計（デザイン・スケッチ）が開発プロセスの先頭

---

で示され、はじめから機能設計・構造設計混在となりやすい。公理系設計の理論では「機能設計→構造設計」という順序で設計が進むことになっているが、メカ設計の場合、概略構造設計の先行、詳細機能設計の延期という形で、機能設計と構造設計の期間的な重複・混在化が起りやすい。

**構造設計重視**：以上をまとめるならば、全体に「構造設計重視」ということが、メカ設計の特徴であろう。しばしば「メカ設計は図面重視」といわれる所以である。要するに、「機能設計は簡潔に、構造設計は精密に」という傾向が表われやすいのである。

## 4.2 エレキ設計の特性

**制御する側の設計**：エレキ（電気）設計とは要するに、制御する側、すなわち制御系の設計である。典型的には、制御対象から情報を受け取り、制御対象に操作信号を送信するための制御回路システムの設計である。

**後発者としての弱さ**：企業の開発組織の中で、エレキ設計はメカ設計より新しいことが多い。テレビなど音響・映像家電のメーカーの場合は、もともとエレキ設計が開発の中心であったため、エレキ設計者が発言力を持っている場合もあるが、自動車など、もともと電子制御のなかった古い機械製品の場合は、メカ設計者に対して、エレキ設計者は新参者ゆえに力が弱いことが多い。

**精密な機能設計**：エレキ設計における機能とは「制御すること」であり、その意味で、顧客に働きかける実際の「仕事＝機能」を要求されるメカ製品とは異なる。

一般に制御系の機能設計は、トランジスターや抵抗やコンデンサを線でつないだ抽象的な「論理回路図」で表現することができる。AND、OR、NOTなどの汎用的な論理素子をブール代数的に結合し、独自の機能＝論理を構築する（澤井監修・緒方、1970）。ここでは、「制御する」という制御系の機能だけが問題であり、素子や結線の位置、体積、重さなどは問題にされない。

要するに、エレキ設計における機能設計は、制御信号の設計であり、それは論理言語で記述される。制御系の設計とは、「制御する意志」の設計のことである。そこでは、まさに「機能先にありき」であり、機能＝制御の完全な記述が必要になる。

**簡素な構造設計**：エレキの場合、構造設計は、物理設計、すなわち回路のレイアウト図である。そこでは、結線は具体的な太さと形状を持ち、素子も体積・重さ・形状を持ち、それらの物理的な位置関係は、部品の位置取りの干渉や電磁干渉に影響を与える。

電気回路は、汎用的な論理素子や論理ブロック（言語で言えばアルファベットや単語）を組み合わせて、一気に要求機能を実現するため、比較的フラットで簡素な構造ヒエラルキー（部品表）となりやすい（上野、2005）。

**エレキのアーキテクチャ傾向**：エレキの制御系の場合、汎用的な論理素子で機能を構成するため、少なくとも階層の下位レベルはオープン・モジュラー型のアーキテクチャになりやすい。論理系

---

子・論理ブロックに対応する汎用的な個別電子部品あるいは半導体の物理設計を対応させるわけであるが、いったん論理設計（機能設計）が完成すれば、物理設計への展開は繰り返し作業的な部分が多く、自動化を進めやすい。

**機能設計→構造設計という先後関係**：したがって、エレキ設計の場合、回路設計により顧客が要求する機能をどう論理的に表現するかに力点が置かれる。つまり、機能設計（回路設計）が構造設計（レイアウト設計）より重視される傾向がみられる。また、メカ設計に比べれば、「機能設計→構造設計」という順序関係が明確である点も、エレキ設計の特徴といえそうである。したがって、開発初期において、機能設計情報を完備させねばならない、というプレッシャーは、メカ設計以上にある。

**機能設計重視**：以上をまとめるならば、エレキ設計では、機能設計が主体である。繰り返すが、制御系の機能は「制御すること」であり、そこでは、紛れのない、機能の事前記述が必須である。一方、構造設計は、その論理設計に適合する汎用部品を市場で購入する、すなわち「部品を拾う」ことで完成する。メカに比べれば、エレキの構造設計は比較的簡素である。

### 4.3 ソフト設計の特性

**制御する側の設計**：複雑なメカ・エレキ・ソフト製品におけるソフト設計とは、通常は、メカを制御対象物とする制御ソフトであり、いわゆる「組み込みソフト」と呼ばれる人工物である。制御する側であるから、当然、制御目標は明確でなければならない。

一般に人工物の制御系は、基本的にエレキ設計とソフト設計で分担することになるが、エレキ系とソフト系をどこで切り分けるかについては、設計者はかなりの自由度を持つ（上野・藤本・朴、2007）<sup>7</sup>。

**最も後発である故の発言力不足**：ソフトのエンジニアは、機械製造企業の開発組織において、最も新しいグループであることが多い。したがって、メカ設計者、エレキ設計者に対して、十分な発言力を持たない傾向がある。このため、例えば開発後半で機能達成が難しくなったとき、最後に動員されるのはソフトウェア設計者であることがよくある。

**機能設計は論理表現**：ソフトウェアが担う制御情報は、半導体上で表現される。したがって、機能要件は、論理設計情報として事前に完備していなければならない。その表現は、より自然言語に近い、コンパイラ言語で表現されている。つまり、ソフトウェアの機能は、記号（言語）表現→論理設計表現の順に翻訳されることになる。ソフトウェアの論理表現には、従来はフローチ

---

<sup>7</sup> ソフトウェアは、制御系を担うという意味ではエレキ設計と同様の役割を担う。しかし、いわゆる電気設計の場合、PCB（プリント基板）上で汎用的な部品を選択して挿入する、という「逐次転写型」（組立型）のものづくりであるのに対し、半導体の場合は、文字通り、トランジスタと回路を一括して掘り込んでいく一括転写型（加工型）のものづくりであり、この点で大きく異なる。

---

チャートなどが使われており、より新しい UML の場合は、ユースケース図、ドメイン構造図（階層）、クラス図（階層）などの形式的モデルで表現される。

いずれにせよ、ソフトウェアの機能、すなわち制御操作は、構造設計（コード生成）に入る前に、論理的に完結していなければいけない。このように、機能設計の完備性を要求する点では、ソフト設計はエレキ設計と同様である。

**構造設計はソースコード**：ソフトウェアの構造設計は何か、という解釈は難しいが、ソフトウェアが言語であるということから、書かれた言葉に当たる「ソースコード」がソフトウェアの構造設計に当たると考えてよい。ソースコードの生成は、機能設計が完備していれば、かなりの程度自動的に行われる。

**ソフトのアーキテクチャ**：制御対象になる機器のリアルタイム要求が厳しい場合、ソフトには割り込み処置が複雑に入り、ソフトのアーキテクチャはインテグラル型になりやすい。また、CPU（半導体のハード）に直接、アセンブラで記述したアプリケーション・ソフトが乗るような非階層的な構造になる傾向があり、その点から見てもインテグラル的である。

そうした機能要求が厳しくない場合は、CPU→OS→アプリケーション、あるいは CPU→OS→ミドルウェア→アプリケーションといった階層構造になり、全体にソフトウェアはモジュラー化しやすい。

つまり、ソフトのアーキテクチャは、機能要求に応じて、モジュラーにもインテグラルにもなる。しかし、リアルタイム要求の厳しい製品の場合は、インテグラル寄りになりやすい。

**機能設計重視**：以上をまとめると、ソフトウェア設計の場合も、エレキ設計と同様、機能設計（論理設計）が完備しないと構造設計（コード生成）に進めず、逆に機能設計が完備すれば、構造設計はかなりの部分、自動コード生成が可能である。つまり、顧客の要求を論理に翻訳する機能設計が重要である。

以上のように、メカ設計、エレキ設計、ソフト設計は、歴史的背景も、設計の文化も、かなり異なる傾向がある。何よりも、メカは構造重視（簡素な機能設計・精密な構造設計）、エレキ・ソフト設計は機能重視（精密な機能設計・簡素な構造設計）という発想が顕著であり、しかも、歴史的に先行するメカ設計がエレキ設計に対しより強い発言力を持つ、あるいはメカ設計・エレキ設計がソフト設計に対して発言力を持つことが多い。こうした設計風土の分化（Lawrence and Lorsch, 1967）が、複雑な人工物の開発組織・開発プロセスの統合化をいっそう難しくしていると考えられる。

---

## 5 メカ・エレキ・ソフト統合化への筋道

### 5.1 記号としての設計

以上のように、人工物の複雑化に伴い、制御系の複雑化、それにもなうメカ設計、エレキ設計、ソフト設計の連動性が重要度を増している。歴史も発想も相当に異なるこの3つの流れをいかに統合し、全体として人工物の機能を保証するかが、企業にとっての課題となっている。

この問題を考える上でのひとつの出発点は、「設計情報とは人工物の機能と構造を表象する記号の体系である」という、ものづくり論の基本認識に立ちかえることだと筆者は考える。つまり、人工物の設計情報を「複合的な記号体系」と考え、記号間の翻訳を円滑に行うことをめざすのである。

一般に設計とは、人工物を表現する記号である。記号には、記号の恣意性・流通性などの特性の違いに応じて、①自然言語（コトバ）、②論理記号（人工物の物理的特徴を抽象的・論理整合的に表現）、③物理記号（人工物の物理的特徴を表現）などがある。この間の翻訳の巧拙が、人工物の出来を左右すると見てよいだろう。

### 5.2 自動車の設計体系：メカ設計の支配的地位

たとえば、複雑化する人工物の典型例である自動車を考えてみよう。

かつて、ボディ（車体）やシャシー（車台）だけでなく、エンジンやステアリングといった機能部品も全て「メカ」でつくられていた時代の自動車は、おそらく自然言語は直接的に物理記号（図面など）に翻訳されていたと思われる。例えば、「流れるような曲面ボディーデザイン、最高スピードと加速性能を持つスパルタンなコンパクトカー」という製品コンセプトをコトバとして共有することで、その解釈に厳密な取り決めがなくても、最小限の機能設計情報で、構造設計へ直ちに移行し、部品の詳細構造設計をとりあえず行い、部門間で設計の擦り合わせを行い、試作車両を完成させ、それによって機能検証を行うことで、機能設計・構造設計双方の完成度を試行錯誤的、スパイラル的に高めることが可能であった。

しかし今日の自動車を、この様な方法で開発するのは難しい。それは自動車の電子制御が、少なくとも大半の機能部品、あるいはそれらをシステム化した自動車の機能系において、設計の大前提となってきたからである。

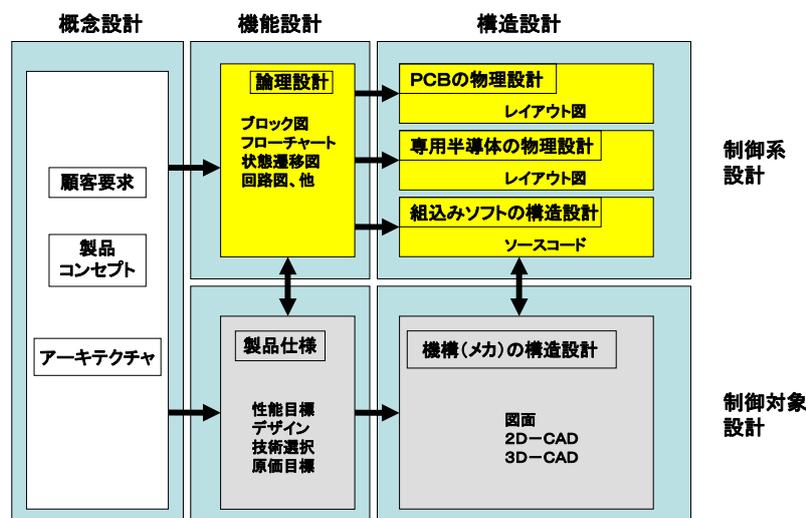
以上、「制御」の観点から自動車の設計は次のように要約できる。

- ・ 自動車はもともと、メカニカルな「制御対象」、つまり「制御されるもの」であった。制御系は運転者自身、つまり人間であった。自動車は、自動制御機械ではなく、手動制御機械であった。
- ・ したがって、伝統的な自動車設計は、運転者の操作( $z$ )に対して、ある環境 ( $e$ ; 例えば

路面状況)のもとで、あるべき反応(y)をするような、メカニカルな構造設計パラメータ(x)を探索することであった。また、テストドライバーが運転者(z)、テストコースが環境(e)、試作車が構造設計(x)をシミュレーションする実物での機能検証が中心であった。

- このため、伝統的には、自動車の設計はメカニカルな人工物の設計プロセスであり、それは本質的に、「制御されるもの」(制御対象)の設計であった。「制御する側」すなわち制御系の設計、すなわちエレキやソフトの設計は、この段階では重要ではなかった。
- かくして、自動車の設計プロセスでは、歴史的に先行するメカ設計の論理が支配的であった。そして、メカニカルな設計は、概して構造設計重視であった。機能設計は簡潔、構造設計は精密であることを良しとするのが「メカ設計の文化」である。
- しかし、近年の自動車では、電子制御系が発達し、プリント・サーキット・ボード(PCB)、半導体、組込みソフトなどを設計するエレキ設計・ソフト設計の比重が増した。ここでは、機能設計情報が早期に完備されることが重視される(図14)。
- 歴史的に後発であるエレキ設計・ソフト設計の論理は、自動車開発では相対的に軽視され、これら制御系の技術者の不満が蓄積される傾向がある。例えば、電子制御系を担うエレキ・ソフト設計は機能設計(論理設計)をできるだけ早い段階に完全なものにしたいが、メカ設計は簡潔で絞り込まれた機能設計(要求仕様)を指向し、その完備化を急がない。このため、エレキ設計やソフト設計に対する完備された仕様の提示が遅れ、彼らの仕事が開発終盤に追い込まれる。とりわけ、融通が利くと見られがちな組込みソフトウェア開発において、こうした駆け込み設計の圧力が顕著である。

図14 メカ・エレキ・ソフトの設計体系



---

**図 14** で示したように、制御系を含む複雑な人工物（たとえば自動車）の設計では、まず自然言語（製品コンセプト）で車 1 台をまるごと表現し、しかるのちに、①メカの場合は仕様→構造設計、②エレキ設計や半導体設計の場合は論理設計→物理設計（レイアウト）、③ソフトの場合は論理設計→ソースコード、という形で翻訳する。そして、これらの間の「メカ・エレキ・ソフト統合」が、複雑な人工物の設計を成就させる上で、重要性を増しているのである。

### 5.3 記号系としての自動車設計と「エレキ・メカ・ソフト統合」

このように、メカ・エレキ・ソフトが複合的に連動する人工物となった自動車を、我々はどうのように設計し、全体の機能を保証しているのだろうか。

「メカ・エレキ・ソフト統合」の基本は、概念設計段階で、車 1 台をできるだけ周到に、自然言語で表現することであろう。つまり、一般の人間が理解できる記号、すなわち自然言語で、まず、その自動車の「あるべき姿」をまるごと総合的に記述することである。人間は、最終的には言語で製品＝人工物を評価する。多義的な概念も、メタファー（暗喩）で表現する。したがって、その設計では、まず最初に、自然言語による人工物の表現、つまり製品コンセプトが必要となるのである。

自動車、とりわけ乗用車は消費財であり、一般消費者が自然言語（コトバ）で理解できる人工物でなければいけない、ということが、自動車設計の原点である。コトバで要約された製品設計情報のことを、製品コンセプトといい、自動車設計の起点はここである（Clark and Fujimoto, 1991）。したがって、自動車の製品開発とは、コトバで表した製品コンセプトを、現物であるクルマに落とし込むための、記号体系間の翻訳・転写作業（藤本、1997；藤本、2003）だとみなすことが出来る（**図 15**）。

図 15 メカ・エレキ・ソフトの記号体系

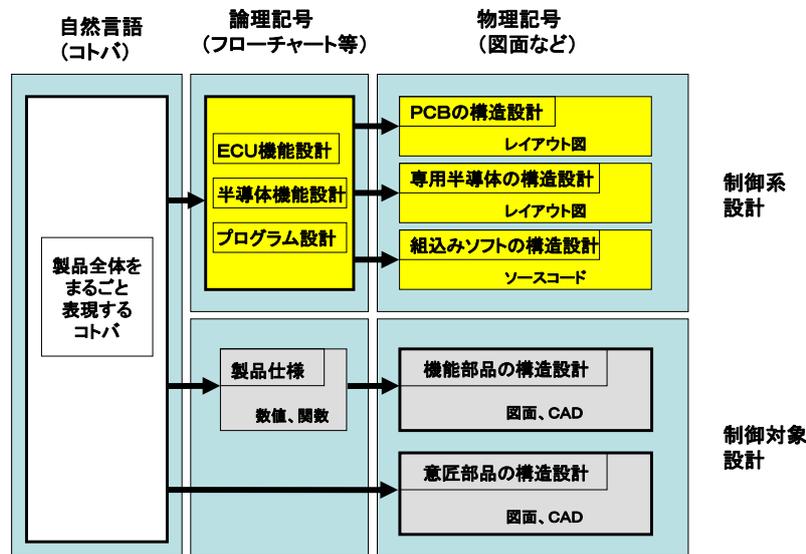


図 15 に示したように、設計プロセスでは、自然言語に加えて、論理言語（抽象的な記号の間に厳密な論理演算が成立するもの）、物理言語（対象物の物理特性を表象した記号。図面やレイアウトなど）などがあり、開発者は、設計の段階と用途に合わせてこれらの記号系から適切なものを選ぶ。そして、同じ人工物を表象するこうした記号系・言語系の間で、しるべき翻訳を行うのが、製品開発の仕事の重要な部分である。

要するに、人工物の機能的な一貫性（プロダクト・インテグリティ；Clark and Fujimoto, 1990）を実現する「メカ・エレキ・ソフト統合」のためには、まずもって自然言語で製品コンセプトを表現し、しる後に論理記号や物理記号に正確に翻訳する必要がある。結局、「自然言語による人工物の丸ごと記述」が、統合の要の役割を果たすわけである。その意味では、製品コンセプトの守護者であり、プロジェクトの調整者でもある「重量級プロダクトマネージャー」（Clark and Fujimoto, 1991）の役割は、複雑な人工物の開発において、依然として重要だと思われる。つまり、メカ設計・エレキ設計・ソフト設計の風土を統合する扇の要は、結局、「コトバとしての製品コンセプト」だと考えられるのである。

#### 5.4 まとめ

本稿では、複雑化する人工物の設計に対して、どのようなアプローチで対処するか、とりわけメカ・エレキ・ソフト統合をどのように考えるかについて、予備的な考察を行った。例えば自動車の場合、元来は被制御系であった自動車に、近年電子制御系が急速な勢いで入り込んでいることが、メカ・エレキ・ソフト設計の間の相互作用と緊張関係を生み出しており、それが、製品開発の複雑化に拍車をかけていることに注目した。

---

とりわけ、制御系であるエレキやソフトは機能設計重視、被制御系であるメカ設計は構造設計重視である、という設計風土の違いが、両者の融合を難しくする一因であることを指摘した。また、製品設計を人工物を表象する記号系と考えた場合、メカとエレキでは、選択される記号系や翻訳のタイミングの違いがあり、これらの設計をコンカレントに行おうとしたとき、情報のやりとりにおいてタイミングのずれが生じうることを示した。

本稿は、複雑化する人工物の設計に関する研究における端緒的なものであり、したがって、探索的、かつ仮説構築的である。研究は出発点にあり、今後の研究課題は多い。とりあえず、現状における作業仮説をいくつか示し、今後の研究の方向性に関する一つのヒントとしたい。

- ・ 顧客の要求機能や社会的な制約条件（環境・安全対応など）が高度化・複合化すると、人工物としての製品は機能的に複雑化する傾向がある。自動車はそうした製品である。
- ・ こうした機能の複雑化・高度化に対して、ある製品は製品の大幅なモジュール化で対応する。例えば PC である。
- ・ しかし、要求機能や制約条件が一定以上に厳しくなると、モジュラー化による処理は難しくなり、製品はインテグラルかつ複雑なものにならざるをえない。こうした人工物を正確に機能させるには、電子制御系の発展は必須である。自動車はそうした製品である。
- ・ また、機能の複雑化・高度化に対して、ある種の家電エレクトロニクス製品は、制御系・被制御系の双方をメカからエレキ・ソフトに置き換え、さらに全体をデジタル化させることによって、開発の複雑性を低減せしめている（例えば回転部品の廃止）。例えばテープレコーダーに対する IC レコーダーである。
- ・ しかし、技術的・物理的な理由により、被制御系を中心に機構部品（メカ）が多く残り、結果としてメカとエレキ・ソフトの共進化が促進される製品もある。自動車はそうした製品である。
- ・ この結果、自動車は、被制御系であるメカ設計の高度化と、制御系であるエレキ・ソフト設計の高度化が、同時進行で起こる傾向がある。しかし、メカ・エレキ・ソフトの設計高度化が、相互に協調的な形で起こるとは限らない。
- ・ とくに、この3つの設計系は、機能的および歴史的な理由から、機能設計重視か構造設計重視か、記号系として何を選択するか、記号間の翻訳をどのように行うか、などに関して、異なる設計風土を持つ傾向があり、それらは分化していく傾向さえある。
- ・ また、自動車については、もともと被制御系の人工物であったという歴史的な経緯もあって、メカ>エレキ>ソフトという力関係が今でも見られる。
- ・ こうした分化傾向に対抗して、複雑化する人工物におけるメカ・エレキ・ソフト設計の統合

---

化を進めるには、設計活動の源流にある「自然言語＝コトバによる人工物の表現」をより精緻なものにする、あるいは開発上流におけるメカ・エレキ・ソフト各設計者を巻き込んだチームワーク設計を促進する新たな IT を模索するなど、開発上流における統合化の努力が極めて重要である。

消費者のニーズが高度化し、環境、エネルギー、安全などに関する制約条件が厳しくなる傾向は、今後も長期にわたって続くだろう。したがって、これに対応する人工物の複雑化も、長期的なトレンドと考えるべきだろう。それに対する対応策としては、製品アーキテクチャのモジュラー化、開発支援 IT (3次元 CAD・CAE) や品質工学などによるフロントローディング、電子制御系(とりわけソフトウェア)の高度化、チーム開発の組織能力の構築など、様々な打ち手を総動員する必要があるだろう。

この問題に対する総合的な解については、まだ探索の緒に付いたばかりである。しかし、少なくとも現段階で予想できることは、この問題への対応を誤る企業は、窮地に落ちる恐れがある、ということである。本稿は、こうした人工物の設計、あるいはものづくりに関する、長期的な課題に対する、端緒的な試論と位置づけられよう。

## 文献

- 青木昌彦・安藤晴彦編著 (2002) 『モジュラー化』 東洋経済新報社
- 青島矢一・武石彰 (2001) 「アーキテクチャという考え方」『ビジネス・アーキテクチャ』 藤本隆宏・武石彰・青島矢一編 有斐閣
- 新木廣海 (2005) 『日本コトづくり経営』 日経 BP 社
- Baldwin, C. Y. and K. B. Clark (2000) *Design Rules: The Power of Modularity*, MIT Press, Cambridge, MA. (安藤晴彦訳『デザイン・ルール』 東洋経済新報社、2004)
- Chomsky, Noam (1965) *Aspects of the Theory of Syntax*, MIT Press, Cambridge, MA (安井稔訳『文法理論の諸相』、研究社出版、1984)
- Clark, Kim B. and Takahiro Fujimoto (1990) "The Power of Product Integrity." *Harvard Business Review*. November - December, pp. 107 - 118.
- Clark, K.B. and Fujimoto, T. (1991) *Product Development Performance*, Harvard Business School Press, Boston (田村明比古訳『製品開発力』ダイヤモンド社、2003)
- Fine, Charles. H. (1998) *Clockspeed: Winning Industry Control in the Age of Temporary Advantage*, Perseus Books, MA. (小幡照雄訳『サプライチェーン・デザイン』日経 BP 社、1999)
- 藤本隆宏 (1997) 『生産システムの進化論』 有斐閣
- 藤本隆宏・安本雅典編著 (2000) 『成功する製品開発』 有斐閣

- 
- 藤本隆宏 (2001a) 「アーキテクチャの産業論」『ビジネス・アーキテクチャ』 藤本隆宏・武石彰・青島矢一編、有斐閣
- 藤本隆宏 (2001b) 『生産マネジメント入門』 日本経済新聞社
- 藤本隆宏 延岡健太郎・青島矢一・竹田陽子・呉在烜 (2002) 「情報化と企業組織：アーキテクチャと組織能力の視点から」『電子社会と市場経済』 奥野正寛・竹村彰通・新宅純二郎編著、新世社
- 藤本隆宏 (2003) 『能力構築競争』 中央公論新社
- 藤本隆宏 (2004) 『日本のもの造り哲学』 日本経済新聞社
- 藤本隆宏・東京大学 21 世紀 COE ものづくり経営研究センター (2007) 『ものづくり経営学』 光文社
- 富士通・日本発ものづくり研究会 (2007) 『モノを作らないものづくり』 日科技連出版社
- Langlois, R. N. and P. L. Robertson (1992) “Networks and Innovation in a Modular System: Lessons from the Microcomputer and Stereo Component Industries,” *Research Policy* 21: 297-313.
- Lawrence, Paul R., and Jay W. Lorsch (1967) *Organization and Environment*, Homewood, Illinois: Richard D, Irwin.
- 西村清彦 (2004) 『日本経済見えざる構造転換』 日本経済新聞社
- 延岡健太郎 (2006) 『MOT [技術経営] 入門』 日本経済新聞社
- 緒方興助 (1970) 『入門自動制御』(澤井善三郎 監修) オーム社
- 呉在烜、藤本隆宏 (2001) 「電子調達ネットワークと部品取引方式—自動車産業の事例—」 ディスカッション・ペーパー, CIRJE-J-44, 東京大学大学院経済学研究科・経済学部
- 大鹿隆、藤本隆宏 (2006) 「製品アーキテクチャ論と国際貿易論の実証分析」『赤門マネジメントレビュー』 5 巻 4 号、233-271.
- 奥野正寛、瀧澤弘和、渡邊泰典 (2006) 「人工物の複雑化と製品アーキテクチャ」 ディスカッション・ペーパー, MMRC-J-81
- 酒井由夫 (2006) 『組み込みソフトエンジニアを極める』 日経 BP 社
- 新宅純二郎 (2003) 「アーキテクチャ論から見た中国との分業」『日本機械輸出組合 JMC ジャーナル』 11 月、2-7.
- 新宅 純二郎・善本哲夫・加藤寛之 (2004) 「中国モジュール型産業における日本企業の戦略」『赤門マネジメントレビュー』 3 巻 3 号、95-114.
- Simon, H. A. (1969) *The Science of the Artificial*, MIT Press, Cambridge, MA.
- Suh, N.P. (1990) *The Principles of Design*, Oxford University Press, New York (畑村洋太郎監訳 『設計の原理』 朝倉書店、1992)

---

田口玄一 (1999) 『品質工学の数理』 日本規格協会

Thomke, S. and Fujimoto, T. (2000) “The Effect of ‘Front-Loading’ Problem Solving on Product development Performance.” *Journal of Product Innovation Management* 17: 128-142.

上野泰生 (2005) 『実践デジタルものづくり』 白日社

上野泰生・藤本隆宏・朴英元 (2007) 「人工物の複雑化とメカ設計・エレキ設計 —自動車産業と電機産業のCAD利用を中心に—」 ディスカッション・ペーパー, MMRC 近刊

Ulrich, K. T. (1995) “The Role of Product Architecture in the manufacturing Firm,” *Research Policy* 24: 419-440.