

# A New Systems Design Method Based on Concept Transfer

## - Design Concepts Abstracted from Processor Architecture -

Mitsunori MIKI\*, Tomoyuki HIROYASU\* and Fumihiro NAKASHIMA\*\*

(Received July 16,1999)

In the case of designing new systems, there is a method called Technology Transfer, which makes good use of established technologies to solve other problems. This method is very effective in the case of applying these technologies directly. However, it is difficult that those technologies are applied to systems in different fields, because these technologies are developed for specific problem solvings. In this paper, a new design method called Concept Transfer is proposed. It applies existing design concepts instead of technologies to designing various systems. At the first stage, design concepts are abstracted from existing systems that seem to have effective concepts for the concept transfer. As a subject for abstracting design concepts, microprocessors of computers are choosed because many effective design concepts are considered to be found from them. After doing detailed investigation of processor architectures, many design concepts that seem to be effective for designing other artifacts can be abstracted. By applying these concepts to another problems, it is expected that they are solved in a new and different way compared to the technology transfer.

**Key words** : Concept Transfer, Technology Transfer, Design, Concept, CPU

キーワード : コンセプトトランスファー, テクノロジトランスファー, 設計, コンセプト, CPU

## コンセプトトランスファーに基づくシステムの設計

### - CPU アーキテクチャから抽出された設計コンセプト -

三木光範・廣安知之・中島史裕

#### 1. 緒言

エンジニアの使命は、科学的知識を技術問題に適用し、与えられた材料、技術的または経済的制約のもとで最適な設計を行い、人類にとって有益なものを作り出すということである。この役割に向けて設計者は高度に重要な貢献をなすわけであり、設計者のアイデア、知識や能力が製品の特性、顧客に与える製品の魅力や全体的な有益性に深く影響を及ぼす。

設計するということは、可能な最善の方法によって与えられた要求を満足させようとする知的な試みである。それは人間生活のほとんどあらゆる局面にかかわり、科学上の発見や法則に依存し、これらの法則を有用な製品の製造に適用するための状況を創る工学活動である<sup>1)</sup>。

基本的な設計プロセスとしては、まず目的を認識し、次に問題設定を行い、その問題を解決するためにアイ

\* Department of Knowledge Engineering and Computer Sciences, Doshisha University, Kyoto  
Telephone: +81-774-65-6434, Fax: +82-774-65-6796, E-mail: mmiki@mail.doshisha.ac.jp

\*\* Department of Knowledge Engineering and Computer Sciences, Doshisha University, Kyoto  
Telephone: +81-774-65-6716, E-mail: fumi@mikilab.doshisha.ac.jp

ディアを創造した後に実際の設計作業を行う<sup>2)</sup>。そして、これらの煩雑な設計プロセスを軽減するために、いろいろな方法論が提案されてきた。

たとえば、テクノロジートランスファーという手法がある。この手法ではすでに確立されている技術を別の問題を解決するために利用する。しかしながら、優れた技術とは定められた問題に特化されて開発されているため、テクノロジーレベルでそのまま他の分野の問題に利用しようとする場合、有効に適用できる範囲は狭いものとなる。

したがって本研究では、テクノロジーに内在するコンセプトの広範囲での応用を目的とするコンセプトトランスファーという新しい設計手法を提案し、その前半部分にあたる設計コンセプトの抽出、解析を中心に行う。

## 2. テクノロジートランスファー

### 2.1 テクノロジートランスファー

なにか新しいものを創造しようとするとき、すべてを自らの発想で行うことは、多大な時間と労力を必要とする上にその効果がどれほどのものかわからない。このような煩雑な作業を軽減するための設計手法の一つにテクノロジートランスファーがある。

テクノロジートランスファーとは、既に他分野においては確立されているテクノロジーを、問題解決や性能向上に利用しようとする考え方である。この手法の利点を以下に示す。

- そのテクノロジーが必要とする条件が整っていれば簡単に導入できる。
- 設計段階でコストや性能が簡単に予想できる。
- もとのシステムを参考にすることで、システム設計のプロセスが利用できる。
- 自社製品などのテクノロジーを用いる場合は、製造機械も流用でき、製造コストを下げることができる。

この手法を用いて、あるシステムの問題解決または性能向上を考える場合、Fig.1 に示すような手順となる。

1. まず、従来のシステム設計と同じく、問題となっている部分の調査、解析を行い、問題を把握する。
2. 次に、同様の問題点を持っているようなシステムで既に解決されたシステム、システム自体および

3. 集められたテクノロジーを解析し、問題となっているシステムへの適用性や導入後の性能やコスト等を判断する。
4. 有効性が認められた場合には、問題となっているシステムを持つ物理的な条件等の制約条件にあわせてテクノロジーを導入する。

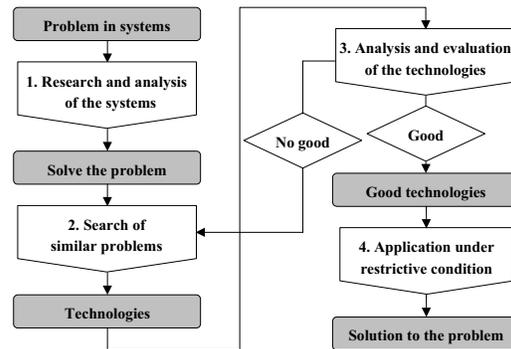


Fig.1 Technology transfer for given problems.

また、テクノロジートランスファーを行うにあたって問題である点は、有効なテクノロジーをどのように入手するかである。そこで、実際の問題に適したテクノロジーをあらかじめ分類、整理しておくことで、テクノロジートランスファーを効率よく行う方法もある。この場合、Fig.2 に示すような手順となる。

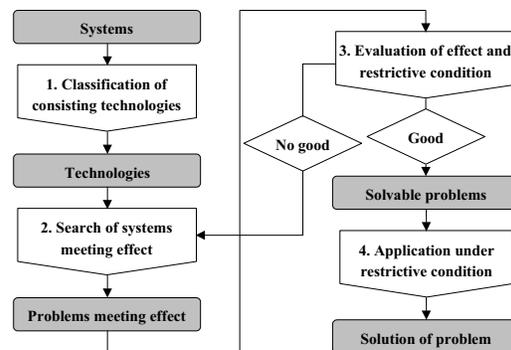


Fig.2 Technology transfer for given technologies.

テクノロジートランスファーの具体例としては、扇風機と換気扇がこれにあたる。両者には羽と羽を回転させる動力とが共通して備わっており、回転運動によって空気を移動させるという効果を持つシス

テムとなっている。システム的な大きな違いは羽の向きが逆であり空気を移動させる方向が異なることで、それは機能の違いとなっている。仮に扇風機に、機能的に共通な部分である回転機構の回転速度を飛躍的に上昇させるようなテクノロジーができたとする。それは簡単に換気扇に応用でき、かつその効果の予想もできることがわかる。同様に回転機構について、省エネや消音ができるようなテクノロジーは両者の間で簡単に応用が可能であると考えられる。

また、テクノロジー転スファアは、このように酷似した用途を持つ様な場合だけが有効なわけではない。たとえば殺虫剤の噴霧機構で培われたテクノロジーを、ゴミ収集所などでの大規模な消臭装置に応用した例などがある。このような場合、もともとの用途から考えれば異なる分野に属するため、一見応用が困難だと考えられる。しかし、機構的には似ている部分があるため応用が可能である。レーザー光による熱を用いて画像形成を行うテクノロジー（レーザー・サーマル・イメージング・テクノロジー）を応用し、印刷機器やフィルムの新しいシステムを実現する場合<sup>3)</sup>や、人工衛星の姿勢制御に使われてるジャイロのテクノロジーを応用して物体の揺れを大幅に減少させるCMG減揺機を開発し、プレジャーボート、ゴンドラ、ロープウェイまたはクレーンなどに適用する場合<sup>4)</sup>などがこの例にあたる。

## 2.2 テクノロジートランスファアの問題点

2.1節で述べたように、テクノロジー転スファアは設計の際のプロセスをいくつか省略できる優れた手法である。しかし、この手法を用いるためには、現在設計しようとするシステムに適したテクノロジーを入手する過程に、次のような問題点がある。

1. 利用可能なテクノロジーが存在しない場合がある。
2. どの分野に有効なテクノロジーであるかわからないため、広い範囲での探索が必要である。
3. 優れた効果があるテクノロジーを発見しても、物理的な制約などから導入できない場合がある。
4. 既にあるテクノロジーを応用するだけなので、従来の設計法に比べて創造性の面で弱さがあるといえる。

本研究では、テクノロジー転スファアのこれらの問題点を克服するために、テクノロジーに内在するコンセプトを応用するコンセプト転スファアという手法を提案する。

## 3. コンセプト転スファア

### 3.1 コンセプト転スファアの定義

前章で述べたように、テクノロジー転スファアにはテクノロジーを有効に応用できる範囲が狭いという大きな問題点がある。しかし、システムに内在するコンセプトに着目すると、システムにあまり依存しない形で有効な点や問題点を把握できると考えられる。ここでは、システムが持つ有効な要素をコンセプトとして抽出して別の分野のシステム設計に利用する手法を提案し、これをコンセプト転スファアと呼ぶことにした。

### 3.2 コンセプト転スファアの有効性

コンセプト転スファアは、テクノロジー転スファアと同様にすでに存在する有効な要素を応用することにより、従来の設計手法よりも少ない手順で新しいシステムを設計しようとする手法である。従来の設計手法では、問題点を把握した後に、それを解決する方法を考えなければならない。しかしコンセプト転スファアでは、問題点を解決できるようなコンセプトを入手することができれば、それを具体的に実現するだけで新しいシステムを設計することになるため、従来の設計手法より早い問題解決が可能であるといえる。

コンセプト転スファアは、既存の有効性の高いシステムに内在するコンセプトを抽出して扱うことにより、システムが持っていた有効性を失わずにシステムへの依存性だけを取り除く手法である。これは、コンセプトは物理的な制約を受けないで存在できるからである。これにより、テクノロジー転スファアの問題点であった適用範囲の狭さを克服している。すなわちコンセプト転スファアでは、物理的なシステムに用いられているコンセプトを、社会的なシステムの問題解決に応用するといった、異なる分野での有効性の応用が可能であると考えられる。

テクノロジー転スファアの効果的な方法として、テクノロジーを先に多く収集しておく方法を述べた。しかしこの手法においては、整理、分類したテクノロジーは元のシステムに依存しているため、適用の際にシステムの物理的な構成を考慮する必要がある。すなわちテクノロジー転スファアでは、テクノロジーを整理する段階と応用を考える段階を切り離して行うことが困難であるといえる。一方コンセプト転スファアでは、コンセプトはシステムにあまり依存しないため、抽出段階と適用段階に分けて

行うことができる。これにより、ある問題を解決するために抽出し利用したコンセプトやその際に利用しなかったコンセプトも、収集しておくことができ、広い分野での再利用に用いることが可能となっている。

テクノロジートランスファーでも、類似した分野での利用を考えるならば、このような二次的な利用は有効である。しかし、テクノロジートランスファーでは、最新のテクノロジーもすぐに新たな技術革新によって効果が減少することが問題となる。コンセプトにはこのような問題がないため、コンセプトトランスファーではコンセプトを資産として保存しておくことが利点となる。

また、テクノロジートランスファーは、従来の設計法に比べて創造性の欠如という問題点があると述べた。コンセプトトランスファーでは、そのコンセプトの抽出元のシステムにとらわれないため、創造性のある自由な応用が可能であり、テクノロジートランスファーの創造性の欠如という問題点を克服しているといえる。

### 3.3 設計コンセプトの抽出方法とその利用方法

#### 3.3.1 設計コンセプトの抽出方法

コンセプトトランスファーを行う方法には、テクノロジートランスファーと同じように、ある問題を解決するために有効なコンセプトを探す方法と、あらかじめ多くのコンセプトを収集しておきその利用を考える方法が考えられる。しかし前者の、問題を解決するコンセプトを得るためにコンセプトの抽出を行う方法は、システムとコンセプトに依存関係がないため、うまく問題に適したコンセプトを発見できるとは限らないことが問題となる。そこで後者の、あらかじめ多くのコンセプトを準備する方法について考える。

3.2節で、コンセプトトランスファーでも、テクノロジートランスファーでテクノロジーを準備しておくことと同様に、多くのコンセプトを準備しておくことは有効であると述べた。さらにコンセプトトランスファーでは、コンセプトの抽出段階と適用段階を切り離してもコンセプトの有効性が失われないという特徴がある。そこでコンセプトトランスファーの実行手順を、抽出段階と適用段階に分けて解説することにする。

システムからコンセプトを抽出する際のフローチャートを、Fig.3に示す。

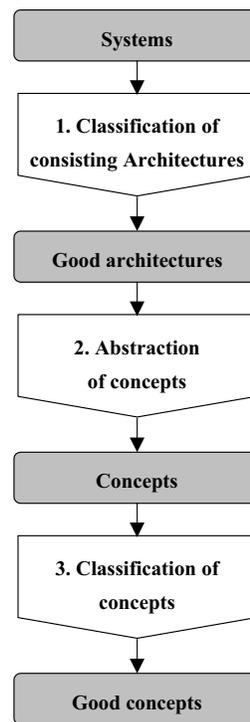


Fig.3 Abstraction of design concepts.

コンセプトの抽出段階の手順は次のようになる。

1. 用いられているアーキテクチャを、システムの目的、特徴、および主な効果などに着目して整理する。
2. 有効なアーキテクチャの特徴的な目的や構造に着目してコンセプトを抽出する。
3. コンセプトを利用しやすい形で分類、整理する。

コンセプト抽出段階では次の二つの点に注意しなければならない。

第一に、コンセプトを抽出する前に、システムまたはそれを構成するテクノロジーについて、そのテクノロジーがもつアーキテクチャがどのような目的を持つかを把握しておく必要がある。これは、有効性の高いシステムは多くの有効なアーキテクチャの集合である場合が多いため、システムから直接コンセプトを抽出すると、いくつかのコンセプトの効果間で矛盾を生じ、効果の混同が起こる危険性があるためである。

たとえば、高い能力を目指したアーキテクチャと低エネルギーを目指したアーキテクチャの両方を持つシステムについて考える。高性能を目指したアーキテクチャの部分は、そのエネルギー面を犠牲にして目的を達成するコンセプトだとする。低エネルギーを目指したアーキテクチャの部分は、性能を犠牲して目的を達成するコンセ

プトだとする。それぞれのアーキテクチャについて見ればコンセプトは有効に作用していることがわかる。しかし、個々のアーキテクチャを考慮せずに一つのシステムとしてコンセプトを列挙した場合、このコンセプト間に矛盾を感じるようになる。すなわち、システムにとっての重要なコンセプトを誤認する危険性があるといえる。

このように、構成するアーキテクチャについて、それぞれの目的の区別がない状態でシステムからコンセプトを抽出すると、コンセプトの効果を誤る可能性がある。このため、個々のコンセプトの効果を正確に把握するためには、システムを構成する個々のアーキテクチャを目的などで整理し、アーキテクチャごとにコンセプトを抽出していく方法が有効であると考えられる。

第二に、収集したコンセプトをうまく整理しておく必要がある。多くのコンセプトを収集していく際、これらを規則性のない状態で保持しているといくつかの問題が生じる。たとえば利用段階ではコンセプトを選択する際に、必要ではないコンセプトまでも考察することになる。また、新たなコンセプトを追加する際には、同じコンセプトを保持しているかどうか調べるのが困難となる。このため、収集したコンセプトを、何らかの法則に従って分類、整理を行う必要がある。

たとえば、コンセプトが持つ主な効果での分類、整理が考えられる。この方法では、コンセプトを選択する際に、求める効果があるコンセプトだけを試せばよく効率的である。このように、効果による分類、整理は、先にあげた問題点を軽減しているといえる。しかしさらに多くのコンセプトを追加整理すると、コンセプトの効果だけで選択したのではうまく絞り込めず、結局多くのコンセプトについて考察しなければならない。また、どのような効果が有効か分からないシステムへのコンセプトトランスファーを考えると、この分類は何も効果を持たなくなる。

このため、コンセプトトランスファーを広く適用するのであれば、他に複数の分類を考える必要がある。コンセプトが効果的となる条件や、コンセプトの構造的な特徴が、他の分類として考えられる。このように、コンセプトを抽出する際には、何らかの利用時の利便性を考慮した分類、整理が必要となる。

### 3.3.2 設計コンセプトの抽出方法

新しいシステムを設計する際の、コンセプトを抽出し、整理したコンセプトを利用する段階の手順は次のようになる。

1. まず、従来のシステム設計と同じく、問題となっ

ている部分の調査、解析を行い、問題となる部分を把握する。

2. 整理したコンセプト群の中から、解決すべき問題部分が求める効果を持つものを列挙する。
3. 各コンセプトについて、システムにあわせて具体化した場合のコンセプトの効果を考察し、システムの問題となっている部分に有効であるコンセプトを選択する。
4. そのコンセプトを採用することにより、さらなるコンセプトが採用できる場合があるため、その可能性を考える。
5. 選択された有効なコンセプト群を具体化してシステムに導入する。

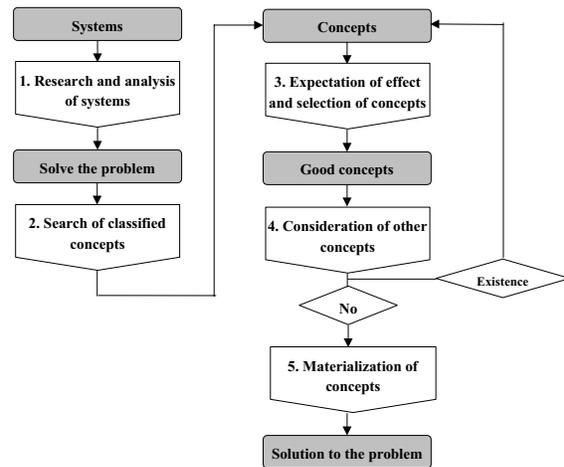


Fig.4 Application of design concepts.

また、Fig.4にコンセプトの利用段階のフローチャートを示す。Fig.4の手順4の段階でいったんコンセプトを選択する段階に戻るのには、そのコンセプトを導入する事で、元のシステムの条件が変わるためである。すなわち、より効果的なコンセプトが活用できる可能性や、コンセプトを導入したために起こるデメリットを軽減させるコンセプトが存在する可能性などがあるためである。再び手順4の段階になったときに前回と変化がなければ、応用できるコンセプト群の中にこれ以上有効なコンセプトの組み合わせはないと考え手順5の段階に進む。適用の際に重要となるのは、手順4の段階を何度か繰り返すことにより、最初の問題点だけでなくその周囲の問題点も考慮する必要があることである。このためには多くの有効なコンセプトを準備しておく必要がある。

また、コンセプトトランスファーは広い分野でシステムの問題解決や性能向上が可能であるが、そのため

にはより多くの有効なコンセプトを保有しておく必要がある。

#### 4. CPU アーキテクチャにおける設計コンセプトの抽出

##### 4.1 設計コンセプトの抽出対象

コンセプトを抽出するには、まずその抽出対象を決定しなければならない。抽出対象には開発期間が長く、何回も試行錯誤され改良されてきたものが適していると思われる。これは、そのような対象には多くのコンセプトが含まれると考えられるためである。コンセプトが少ない対象を選択した場合、有効なコンセプトが得られず、また新たな対象を探し出さなければならない。つまり、この抽出対象の選択はコンセプトトランスファーを行うにあたって、重要なファクターである。

本研究では、抽出対象としてコンピュータの CPU に用いられているアーキテクチャを選択した。CPU の歴史を見ても分かるように、CPU の高性能化を達成するためには数々の問題点が生じてきたが、その都度、それらの問題点の解決するようなアーキテクチャが開発されてきた。そして、そのアーキテクチャには有効なコンセプトが存在すると考えられる。

##### 4.2 CPU アーキテクチャの調査

CPU は現在に至るまで、高性能化のために発展し続けてきた。その高性能化を可能にしてきたのは、数々の優れたアーキテクチャによるものである。そして、それらの優れたアーキテクチャの土台となっているものは、優秀なエンジニアが研究を重ねて考え出した優れたコンセプトであると考えられる。よって、それらのコンセプトが我々の求める有効な設計コンセプトであると考えられる。そのためにまず、それらのアーキテクチャがどのようなコンセプトで開発されたかを理解していく。ここでは、その準備段階として CPU に用いられているアーキテクチャを文献<sup>4-14)</sup> や Web 情報<sup>15,16)</sup> などを用いて調査し、その結果、計 20 件をフォーマットにしたがって整理した。

##### 4.3 抽出結果

前節の調査結果をもとに、CPU のアーキテクチャの特徴的な要素を抽出した。特徴的な要素とはパイプライン技術 (Fig.5) を例にとると、以下に示すような要素である。

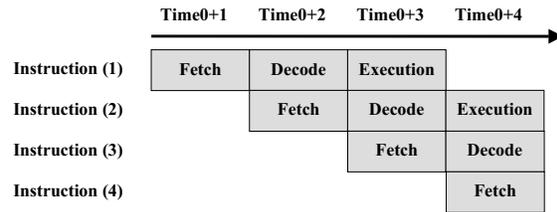


Fig.5 Pipeline architecture.

1. 一般的に命令の制御はいくつかのステージに分割して、実行される。
2. 各ステージの処理は独立した機能ユニットが担当している。
3. 各ユニットは担当する動作のみを命令毎に実行し、次のユニットに後の処理をまかせている。
4. 各クロック毎に命令を処理できることにより、命令のスループットが増加する。
5. パイプラインの段数を増やすことにより、各段の論理が単純化される。

次に抽出した特徴的要素が一つの目的として見たときに、互いに独立であるかを考察し、それらの要素を目的別に分ける。前述の例で考えると、1, 2 および 3 は独立ではなく、4, 5 はそれぞれ独立しているといえる。これは、3 は 2 が存在して初めて成り立ち、2 は 1 が存在して初めて成り立つものと考えられるからである。4 は 1, 2 および 3 の結果であるが、目的として見ると異なる。1, 2 および 3 の目的は、命令を実行する場合にユニットを機能的に分割させる、ということである。それに対して 4 の目的は、単位時間内に実行される命令数を増加させる、ということである。よって、1, 2 および 3 と 4 は異なるといえる。また、5 の目的は分割するユニットの数を増加させて各ユニットを単純化させる、ということであり、これは他の特徴的要素とは異なると考えられる。

最後に目的別に分けられた特徴的要素の一つのコンセプトとして考え、それを適切に表現する。前述の例の場合、我々は 1, 2 および 3 を仕事の機能的細分化、5 を「処理装置の単純化」と表現した。また 4 は、前半の「各クロックごとに命令を処理できることにより」の部分は「直列型並列処理」と表現し、後半の「命令のスループットが増加する」の部分は「単位時間当たりの仕事の処理量の増加」と表現した。この場合、一つの目的から二つのコンセプトが抽出されたことになるが、これは次章で述べるコンセプトの階層構造によるものと考えられる。

以上のような手順でコンセプトの抽出を行い、計 52 件のコンセプトを抽出することができた。

## 5. 抽出したコンセプトの解析

### 5.1 コンセプトの抽象化

前章で抽出されたコンセプトは、抽出元のアーキテクチャから独立したものでなく、他分野への応用を考えるためには、これらのコンセプトを一つの抽象的なコンセプトとして抽出しなければならない。これにより、コンセプトに汎用性が現れると考えられる。本研究では、30 件のコンセプトについて抽象化することができた。Table.1 にその一覧を示す。

### 5.2 コンセプトの階層構造

抽出したコンセプトを収集するだけでは、利用を考える際や新たにコンセプトを収集する際に効率が悪い。

このため収集したコンセプトをうまく整理、分類して保管する必要がある。そこで抽出したコンセプトについて、その特徴といえる構造、目的および効果について考察して考察を行った。この結果、CPU から抽出された設計コンセプトには、効果と、特に有効となる条件に着目して分類すると、何らかの階層構造をなすと考えられた。この階層構造は、適用段階で問題に適したコンセプトを選択する際に、効果的な指針となると考え、これに基づいた整理を行った。なお、以下で用いる「下のコンセプト」とは、その分類で最も重要となっている効果について、それより上に分類されたコンセプトの効果を増す具体的なコンセプトという意味である。また、あるコンセプトについての「上のコンセプト」とは、その分類で最も重要となっている効果について、その効果が十分発揮できるための条件と言う意味である。

Table.1 List of concepts abstracted.

No.	Concepts	No.	Concepts
1	Speculative execution	16	Increase in workload per unit time by parallel handling of inherently sequential jobs
2	Temporary increase of the maximum capacity	17	Keeping newest data
3	Energy-saving by changing abilities according to events	18	Sparing trouble by making use of units with high abilities as much as possible
4	Avoiding of overlap by renaming temporarily	19	Transparent execution
5	Total improvement in performance by adjusting high level object to low level	20	Assurance of restoration in case of failing in speculative execution
6	Effective management by setting fixed value	21	Dynamic dividing of operation fields
7	Functional dividing of jobs	22	Joining frequent jobs
8	Effective design of hardware by handling jobs in software	23	Omitting infrequent jobs
9	Executing feasible jobs first	24	Increase in workload per unit time by parallel handling of inherently parallel jobs
10	Simplification of systems	25	Providing many procedures
11	Making good use of units with low abilities	26	Doing other jobs during waiting
12	Sparing trouble by combined handling	27	Increase in efficiency by making use of identification
13	Preparing of bypaths	28	Distribution of abilities according to conditions
14	Speed buffer	29	Making good use of excessive abilities
15	Direct application of data	30	Quantity buffer

たとえば C という効果について「A というコンセプト」の下に「B というコンセプト」という構造がある場合を考える。コンセプト A から見ると、C という効果を増すための具体的なコンセプトとしてコンセプト B があるということである。コンセプト B から見ると、C という効果を発揮するためにはコンセプト A という土台が必要だということである。このようにコンセプトは階層構造をなすことにより、その効果を発揮していると考えられる。

また、コンセプトを整理する手順は以下に示す通りである。

1. コンセプト群を、各コンセプトの特徴的な効果に基づき、いくつかのグループに分ける。そのグループ内で最も重要な効果またはコンセプトは、システムの問題解決や性能向上を考える際に、重要な目的となると考えられる。そこでこれを目的コンセプト (Fig.6) と呼ぶことにする。
2. ある目的コンセプトについて、単体で十分な効果があるコンセプトを目的コンセプトのすぐ下のコンセプトとする。これらのコンセプトに、方法的な共通点や構造的な共通点がある場合は、これに基づいてまとめることを考える。
3. 各コンセプトについて、そのコンセプトおよびそれより上位のコンセプトの効果を増すことが可能なコンセプトをコンセプト群の中から探す。
4. 手順3のコンセプトがある場合は、それを一段下のコンセプトとする。この際、効果や用法等で共通点があるものは、まとめられるか考察する。また、これらのコンセプトの下にあたるコンセプトがある可能性があるため、手順3にもどる。
5. 手順3のコンセプトがない場合は、ここまで構築された階層構造について、類似した構造が他の階層や目的コンセプトの下にあるかどうか調査する。
6. 類似した構造がある場合は、その階層構造によってなんらかの効果が生まれているものとして、これを考察する。有効性が認められた場合には、より上位のコンセプトとしてまとめなおす。
7. 類似した構造が特にない場合は、その目的コンセプトについて階層構造が構築されたものとして、手順2に戻り他の目的コンセプトについて考察する。

また、これにより構築された、CPU から抽出された設計コンセプトの階層構造の一部を Fig.7 に示す。

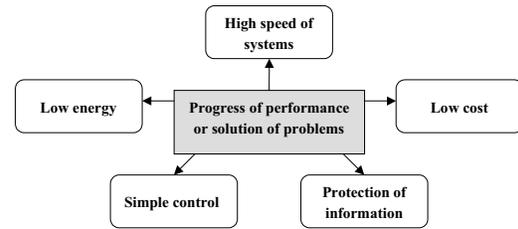


Fig.6 Objective concepts.

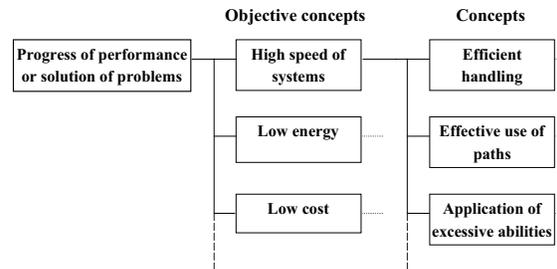


Fig.7 Class structure of concepts.

ここで、この分類方法に従って CPU から抽出したコンセプトを整理した場合の主な利点について述べる。

第一の利点は、単純な形でコンセプトを分類することである。コンセプトには、物理的なシステムに依存しない形で存在できるという特徴があるため、コンセプト間の関係は定義が困難であった。たとえば、A というコンセプトが B というコンセプトに効果的である場合があれば、その逆の関係を持つ場合もあるためである。このため、特徴的な効果や構造だけで整理する場合、さまざまなコンセプトからループを生じるために複雑な構造を持った分類となる。今回用いた分類では、最初に適当な目的を定めて、それに絞った効果について分類を進めるため、効果について分類してもループ状態があまり発生しないという特徴がある。この分類では、「目的となるコンセプト」「特徴的な構造または方法」「具体的な方法」という構造が基本となっている。すなわち、階層構造の中段のコンセプトについても、そのコンセプトを目的コンセプトとしてとらえるとこのような構造として整理される。

第二の利点は、新たに抽出したコンセプトを追加分類する際に、単純な作業で行えることである。この分類では、新たにコンセプトを追加する際にも、各効果についてその目的にそって分類していくため、複雑なループを考える必要がない。また、類似のコンセプト

が収集されているか、分類する過程で同時に調査される。すなわち、新たにコンセプトを追加する際に従来生じていた複雑な作業を、この分類方法では単純で機械的な作業で行うことができる。

## 6. 結 論

従来の設計手法は、多くの時間と労力を必要としていた。また、既存のシステムが持つ有効な技術を利用して、効率良く設計を行う手法であるテクノロジー・トランスファーは、テクノロジーが元のシステムに依存しているため、狭い範囲でのみ有効であった。その解決策として、システムに内在するコンセプトを利用する、コンセプト・トランスファーという手法を提案した。その重要な特徴を以下に示す。

1. コンセプトレベルでの応用を考えるため、原案となるシステムの物理的な制約にとらわれず、テクノロジー・トランスファーよりも広い分野でのアーキテクチャの利用が可能になっている。
2. コンセプトはシステムにあまり依存していないため、抽出段階と応用段階をはっきり分けることができる。
3. コンセプト・トランスファーによる効果を上げるためには、問題解決の支援となる有効なコンセプトを、より多く収集することが重要となる。
4. コンセプト・トランスファーでは、漠然とコンセプトを収集するだけではコンセプトを効率良く問題解決に利用できるとは限らない。

また、上記の特徴である4の問題点を補うためには、収集したコンセプトを何らかの特徴で分類、整理する必要がある。これを考えるため、抽出したCPUの設計コンセプトを考察した結果、CPUの設計コンセプトには次のような特徴があることが分かった。

- コンセプトの中には特定のコンセプトとともに導入しなければ効果が発揮できないものがある。
- コンセプトの中には特定のコンセプトとともに導入した場合効果が増すものがある。
- 適当な構造をもって整理したほうが良い。
- コンセプトが持つ目的や効果によって整理した場合、利用時にコンセプト選択の指針となる構造ができる。

また、これらのコンセプトの他の分野への適用は今後の課題である。しかし、まず目的を見つけてから、

その目的にあった階層構造に基づいて今回抽出したコンセプトを適用すれば、効果的なコンセプト・トランスファーが実現できると考えられる。

## 参 考 文 献

- 1) G.Pahl and W.Beitz「工学設計 = 体系的アプローチ」(培風館, 東京, 1995)。
- 2) 北郷 薫「設計工学シリーズ 1 設計工学基礎」(丸善, 東京, 1972)。
- 3) <http://www.cgarts.or.jp/dictionary/k42/k4213.htm>.
- 4) <http://kiki.kiic.co.jp/homepage/2196.html>.
- 5) David A. Patterson and John L. Hennessy「コンピュータの構成と設計 ハードウェアとソフトウェアのインタフェース 上・下」(日経 BP 社, 東京, 1996)。
- 6) John L. Hennessy and David A. Patterson「コンピュータ・アーキテクチャ 設計・実現・評価の定量的アプローチ」(日経 BP 社, 東京, 1992)。
- 7) Mike Johnson「スーパースカラ・プロセッサ マイクロプロセッサ設計における定量的アプローチ」, (日経 BP 社, 東京, 1994)。
- 8) 嶋 正利「次世代マイクロプロセッサ」(日本経済新聞社, 東京, 1995)。
- 9) 神保進一「最新マイクロプロセッサテクノロジー」, (日経 BP 社, 東京, 1996)。
- 10) Daniel Tabak「RISC システム 新しいプロセッサのアーキテクチャ」(海文堂出版, 東京, 1991)。
- 11) 大貫 徹「RISC プロセッサ入門 R3000・SPARC・860 など主要 5 機種を比較研究」(CQ 出版, 東京, 1991)。
- 12) 小鷲英一「MMX テクノロジー最適化テクニック」, (ASCII, 東京, 1997)。
- 13) 熊谷正康, "Pentium アーキテクチャの徹底研究", Interface, CQ 出版社, pp.92-144(1998-3)。
- 14) 保科正行, "分かる最新プロセッサ技術", SUPER ASCII, ASCII, pp.184-190(1997-6)。
- 15) <http://www.intel.co.jp/>.
- 16) <http://www.sun.co.jp/tech/sunexpert/>.