

平成28年1月27日判決言渡

平成27年（行ケ）第10066号 審決取消請求事件

口頭弁論終結日 平成28年1月13日

判 決

原 告 クアルコム インコーポレイテッド

訴訟代理人弁護士  
弁理士

浅 村 昌 弘
松 川 直 樹
金 井 建
浅 村 皓
浅 村 肇
畑 中 孝 之
白 江 克 則
大 日 方 和 幸
水 本 義 光
亀 山 育 也
岩 見 晶 啓
橋 本 裕 之
福 井 淳

被 告 特 許 庁 長 官  
指 定 代 理 人 西 村 泰 英  
戸 島 弘 詩  
相 崎 裕 恒

田 中 敬 規

主 文

- 1 特許庁が不服2013-9418号事件について平成26年12月2日にした審決を取り消す。
- 2 訴訟費用は被告の負担とする。

事 実 及 び 理 由

第1 原告の求めた裁判

主文同旨

第2 事案の概要

本件は、特許出願拒絶査定不服審判請求に対する不成立審決の取消訴訟である。争点は、①手続違背の有無、②理由不備の有無、③引用発明の認定等の誤りの有無及び④本願発明の進歩性の有無である。

1 特許庁における手続の経緯

ザ ユニバーシティ コート オブ ザ ユニバーシティ オブ エディンバラは、平成18年(2006年)4月28日、名称を「再構成可能命令セル・アレイ」とする発明につき、国際特許出願(特願2008-508300号。優先権主張日：平成17年(2005年)4月28日(以下「本願優先日」という。))及び平成18年(2006年)3月6日 英国。請求項の数56。甲1, 24)をしたが、平成24年3月5日付けで拒絶理由通知を受け(甲2)、同日付けで、原告に対し特許を受ける権利を譲渡した(同年4月3日付け出願人名義変更届出。甲13, 14)。

原告は、同年9月10日付け手続補正書(甲4。以下「本件補正書」という。)により、特許請求の範囲を変更する手続補正をした(請求項の数56。以下、「本件補正」という。)が、平成25年1月18日付けで拒絶査定を受けた(甲5)ので、同年5月22日、これに対する不服の審判を請求した(不服2013-9418号。甲10)。

特許庁は、平成26年12月2日、「本件審判の請求は、成り立たない。」との審決（附加期間90日）をし、その謄本は、同月12日、原告に送達された。

## 2 本願発明の要旨

本件補正後の請求項1に係る発明（以下「本願発明」という。）は、本件補正書に記載された以下のとおりのものである（なお、本願の願書に添付した明細書（甲1）を、図面を含めて「本願明細書」という。以下の分説符号は、裁判所において付した。）。

### 【請求項1】

A1：従属および独立の両方のプログラム命令のデータ経路を有するプログラム命令を実行する

A2：プロセッサであって、

B：相互接続ネットワークと、

C：それぞれが前記相互接続ネットワークに接続される異種の複数の命令セルであって、該命令セルの各々は少なくとも加算、減算、乗算、除算あるいはシフトの処理を行う該複数の命令セルと、

D：復号モジュールであって、

E1：従属プログラム命令のデータ経路と独立プログラム命令のデータ経路の少なくとも1つから

E2：前記命令セルの回路へのマッピングをそれぞれ符号化する構成命令を受け、

F：更に構成命令を復号して前記相互接続ネットワークおよび命令セルの少なくともいくつかのデータ経路を構成し、

G：これにより前記データ経路を前記命令セルの回路にマップした前記プログラム命令を実行する、

D：復号モジュールと、

A：を備えるプロセッサ。

### 3 審決の理由の要点

本願発明は、以下の甲7（以下「甲7文献」という。）に記載された発明（以下「引用発明」という。）及び周知技術に基づいて、当業者が容易に発明をすることができたものである。

#### (1) 引用発明

甲7文献（Francisco Barat, Murali Jayapala, Tom Vander Aa, Rudy Lauwereins, Geert Deconinck, and Henk Corporaal, “Low Power Coarse-Grained Reconfigurable Instruction Set Processor”, 「低消費電力 粗粒度結合型再構成可能なインストラクションセットプロセッサ」 LNCS, Field-Programmable Logic and Applications, 2003年9月, Vol.2778, p.230-239。以下、特に断らない限り、証拠番号には枝番を含む。）には、以下の引用発明が記載されている。

「プロセッサであって、

プログラム可能な相互接続を提供するクロスバーと、

それぞれが前記相互接続クロスバーに接続される処理エレメントであって、該処理エレメントは、ALU（算術論理演算）、シフタ、乗算器を含んだ処理エレメントと、

レベル1の命令キャッシュから命令を読み、前記処理エレメント及び相互接続クロスバーと共に設けられたコンフィグレーションメモリヘスライスのデータ経路コンポーネントのための構成をロードする主演算装置であって、前記主演算装置は、トライマランをベースとしたコンパイラであって、再構成可能なアレイを異なる構成へと統合させるステップを有するプロセッサ用命令セットを生成するコンパイラを含む主演算装置と、  
を備えるプロセッサ。」

#### (2) 本願発明と引用発明との対比

##### ア 対比

(本件の争点と関連しない部分は記載を省略。)

引用発明の「主演算装置」と本願発明の「復号モジュール」との、各々に付された特定事項同士を対比してみると、以下の共通関係ないし相当関係が成立する。・・・引用発明の「コンパイラ」と、本願発明の「復号モジュール」とは、前者の元々の意味がプログラミング言語で書かれた、プログラムのソースコード（原始コード）を、機械語、ないしバイトコードなどの中間言語によるオブジェクトコード（目的コード）に翻訳(変換)する機能単位を指す用語であること、一方、本願発明の「復号モジュール」が行うとする「前記相互接続ネットワークおよび命令セルの少なくともいくつかのデータ経路を構成し、これにより前記データ経路を前記命令セルの回路にマップした前記プログラム命令を実行する」との事項が、ハードウェアに対する命令／処理の実行とされていること、及び、本願発明は、元々従属請求項とされた請求項29において、当該「復号モジュール」が採り得る態様として「コンパイラ」を含むとされていること、以上を総合すれば、「前記相互接続ネットワークおよび命令セルの少なくともいくつかのデータ経路を構成し、これにより前記データ経路を前記命令セルの回路にマップした前記プログラム命令を実行する」べく、プログラム言語命令をハードウェア実行コードに変換するもの同士である点で、両者は一致する。

イ 一致点及び相違点

**【一致点】**

プロセッサであって、  
相互接続ネットワークと、

それぞれが前記相互接続ネットワークに接続される異種の複数の命令セルであって、該命令セルの各々は少なくとも加算、減算、乗算、除算あるいはシフトの処理を行う該複数の命令セルと、

命令のデータ経路から構成命令を受け、前記相互接続ネットワーク及び命令セルの少なくともいくつかのデータ経路を構成し、これにより前記データ経路を前記命

令セルの回路にマップした前記プログラム命令を実行する復号モジュールと、  
を備えるプロセッサ。

### 【相違点】

本願発明の「プロセッサ」は、「実行する」とされる「命令」に関し、「従属および独立の両方のプログラム命令のデータ経路を有するプログラム命令」であり、その結果、「復号モジュール」も双方のプログラム命令を処理するとしているのに対して、引用発明の「プロセッサ」は双方のプログラム命令を取り扱うとしているか不明である点。

#### (3) 相違点についての判断

本願明細書の【0059】から、まず、相違点に係る本願発明の特定事項にある、「従属」プログラム命令、「独立」プログラム命令は、コンパイラ・プロセスにて、ネットリスト作成時に、「従属」命令については「直列に接続」される扱いとなり、「独立」命令については「並列に走る」扱いを受ける、とされていることが分かる。

ところが、かかる命令の性質に従って一連の命令をコンパイラで分析し、独立命令ならば並列処理が可能と扱い、従属命令は逐次処理とすること自体は、本願出願前にこの技術分野では常識的に扱われている（本願優先日前に頒布又は電気通信回線を通じて公衆に利用可能となった、特開2000-112758号公報(甲12。以下「甲12文献」という。)の【従来の技術】欄，【0005】を参照。)事項である。

また、かかる周知技術は、甲7文献における、コンパイラで命令を処理し、クロスバーを用いて処理エレメントをいかようにも結合可能とされる引用発明の「プロセッサ」について、逐次処理が処理エレメント間の結合で可能とする内容を示す記載やマッピングにより並列処理が可能とする内容を示す記載から見て、当該周知技術の適用を自然と惹起するにふさわしいものであると判断される。そして、引用発明のプロセッサに対し、その構成の一部であるコンパイラに、独立命令と従属命令を峻別し、前者は並列処理とし、後者は処理エレメント間をクロスバーで結合させ

ることで、逐次処理をさせることは、当該相違点に関する本願発明と同等のプロセッサを得ることとなる。

してみると、かかる相違点は、周知技術を考慮に入れると、引用発明においても当業者が通常採り得る態様の一つに十分になり得るといふべきであって、発明として当然に想定される自明な態様上の微差といえるので、かかる相違は格別なものではない。また、そのように構成することによる効果も、当業者が予測し得るものである。

したがって、本願発明は、引用発明及び周知技術に基づいて、当業者が容易に発明をすることができたものである。

### 第3 原告主張の審決取消事由

#### 1 取消事由1（手続違背）

審決は、本願発明と引用発明の相違点について、審決において初めて引用された甲12文献のみに基づいて周知技術と認定し、当該周知技術に基づいて容易想到として進歩性を否定したが、このような審判手続には、出願人に弁明の機会を与えることなく拒絶審決をしたという点において重大な手続的瑕疵があるから、審決は取り消されるべきである。

審査過程で示されていない文献を引用することが許されるのは、拒絶理由を構成する引用発明の認定上の微調整や、容易性の判断過程で補助的に用いる場合、ないし関係する技術分野で周知性が高く技術の理解の上で当然又は暗黙の前提となる知識として用いる場合に限られるところ、本件は、これらのいずれにも該当しないことが明らかである

したがって、審決は、「周知技術」に名を借りることで、実質的に新たな拒絶理由であるにもかかわらず、審決理由とすることにより、特許法50条の趣旨を潜脱したものであるから、違法である。

## 2 取消事由2（理由不備）

以下の点において、審決は、本願を拒絶すべきとする理由を十分に示していないから、特許法157条2項4号に反する。

(1) 審決において「周知技術」と認定された甲12文献には、中央演算装置（CPU）について、再構成可能プロセッサを対象とし得る旨の記載や示唆はなく、引用発明と対象とする技術分野が全く異なるため、「適用上の適正」がないにもかかわらず、審決は、そのような適正について説明していない。

また、審決では、「周知技術」を適用するに際し、引用発明のどの部分に、どのようにして、独立命令及び従属命令の処理を担わせることになるのかについての具体的な説明が一切なく、単に「自然と惹起するに相応しい」と一言、曖昧な説明があるにすぎない。したがって、審決には、当該周知技術を適用して特許出願に係る発明の構成を得ることが技術的合理性の見地からみて可能であり、また、相当であることの理由が、十分示されているとはいえない。

(2) また、審決は、本願発明における「符号化」「受け」「復号」「マップ」の四つの動作について、引用発明と対比して、引用発明の「再構成可能なアレイを異なる構成へと統合させるステップを有するプロセッサ用命令セット」と共通であるとして一致点を認定しているが、本願発明において、四つの動作で構成されている機能を、各々の動作と対応させることなく、引用発明の単なる一つの「統合」の機能と共通するものである、と曖昧に認定するのは、拒絶審決をする際の理由として余りに不十分である。

(3) さらに、審決は、本願の「請求項29において、当該復号モジュールが採り得る態様としてコンパイラを含むとされている」と認定しているが、請求項29はそのようなことは全く記載されておらず、認定の理由が示されていない。

## 3 取消事由3（引用発明の認定及び対比判断の誤り）

(1) 引用発明の認定の誤り

審決は、引用発明の認定において、「・・・コンパイラを含む主演算装置と、・・・」としており、主演算装置にコンパイラが含まれていると認定したが、かかる認定は以下のとおり、誤りである。

ア 本願発明及び引用発明は、ユーザ（アセンブリ・メーカー）が設計した任意のロジック回路を書き込んで、動作させることのできるデバイスの総称であるプログラマブル・ロジック・デバイス（以下「PLD」ともいう。）の一種であるFPGAを従来技術として挙げていることから分かるように、共にPLD分野に属する発明（技術）であって、当該分野の技術常識を当然の前提としているが、審決は、このような技術常識を踏まえず、主演算装置に「再構成可能なアレイを異なる構成へと統合させるステップを有するプロセッサ用命令セットを生成するコンパイラ」が含まれるとしたのは誤りである。

すなわち、通常、アセンブリ・メーカーが、何らかの機能を実現する装置を開発する場合、当該装置内に市販の汎用的なプロセッサを用いてアプリケーション・ソフトウェアを当該プロセッサで実行するが、より早くアプリケーションを実行させる必要がある場合には、アセンブリ・メーカーがプロセッサの代わりに、オリジナルのハードウェア（プロセッサのような汎用的なハードウェア（集積回路）ではなく、アセンブリ・メーカーの特定のアプリケーションを実現させるための専用のハードウェア（集積回路））を開発することがある。当該「オリジナルのハードウェア」を開発するために用いられるものの代表例が、FPGAを初めとするPLDである。FPGA内部には、ロジック（論理回路）を実現するための、入力されたデータに対して出力する信号を決定するためのテーブル（コンフィグレーション（設定）可能な要素のことで、FPGAの場合はLUT（ルック・アップ・テーブル。）と出力信号の状態を保持するための記憶素子（LUTと記憶素子の組を「ロジック・セル」という。）が無数に存在し、このテーブルの内容をアセンブリ・メーカーが様々に設定することにより、アセンブリ・メーカーの用途に従った多種多様な論理回路をFPGA内に多数設けることが可能となっている。さらに、この多種多様な論理回路

(ロジック・セル)同士をどのように接続するのかという情報(接続テーブル)をアセンブリ・メーカーが設定することにより、各論理回路(ロジック・セル)が多段階に組み合わせられ、アセンブリ・メーカーが意図する専用的なハードウェア(集積回路)が実現される。FPGAにおいて、ユーザが用途に従って設定する上記テーブル(LUT, 接続テーブル)はSRAMの技術を応用して実現されているため、当該テーブルの設定内容は、装置の電源を切ってしまうと消失してしまうことから、アセンブリ・メーカーは、当該データを書き込んでおくための専用のROMも装置に搭載し、装置の電源投入時に1度だけ、同ROMから同情報を読み出し、FPGA内部の上記テーブルにROM内のデータを設定する機構を設けている。このようなPLDの開発手順としては、一般に、①ソース・コードの作成、②論理合成、③配置配線、④コンフィグレーション・データの生成、⑤デバイス・プログラミング/コンフィグレーションにより行われるが、このうち、論理合成(又は②論理合成と次の③配置配線を合わせて)はコンパイルと呼ばれ、コンパイルのためのソフトウェアをコンパイラと呼び、これはPLDの開発用ツールに組み込まれている。

したがって、コンパイラは、開発用パソコン内にのみ存在し、FPGA等のPLDが搭載された基板(商品)には存在しないことは自明である。

よって、引用発明の主演算装置にコンパイラが含まれるとする余地はない。

イ 仮に、審決の認定どおり、引用発明の主演算装置にコンパイラが含まれるとすると、アプリケーションの実行に際し、その都度、アプリケーションの動作に必要な再構成可能ユニットの構成(コード)を主演算装置によりコンパイルすることになり、引用発明の性能や消費電力にコンパイラの動作が大きく寄与することは明らかである。ところが、甲7文献において、4「Experimental Setup」及び他の章において、コンパイラが消費する電力、コンパイラの動作によるシステムの遅延時間等が一切記載されていないなど、引用発明の性能と消費電力を見積もる際に、コンパイラ(トライマラン型コンパイラを改良したもの)の動作分が対象とされていない。このことは、引用発明にコンパイラを含むとの審決の認定が誤っているこ

との証左である。

ウ そもそも、コンパイラの処理は、非常に多数のクロック・サイクルを必要とするため、1クロック・サイクルで再構成可能ユニットの構成が変更されることを予定している引用発明の動作を念頭にすると、アプリケーションの動作に必要な再構成可能ユニットの構成（コード）を、アプリケーションを動作させる都度、主演算装置によりコンパイルすることになる審決の認定は、以下のとおり不自然である。

第1に、甲7文献において事前にコンパイルした結果を構成メモリ、又は、Fig 1 図中の Unified L2 Cashe 等に保持しておけば、その都度コンパイルをして、余分に電力を消費してしまうことはないので、消費電力を抑えることを目的としている引用発明が、審決の認定しているような無駄な動作を選択するとは考え難いからである。

第2に、アプリケーションの動作時、引用発明では、主演算装置と再構成可能ユニットの両方が協働する機構となっているところ、その都度コンパイルをするということを前提にすると、アプリケーションの動作中に主演算装置がコンパイルをも行う必要があるため、アプリケーションの処理性能に悪影響が出てしまうからである。

エ さらに、コンパイラが、主演算装置に含まれている、又は、コンパイラを主演算装置で動作させる等との記載は一切なく、技術常識としても、そのような事実は存在せず、また、少なくとも、そのような技術常識があることを審決は立証していない。

## (2) 対比判断の誤り

引用発明は、本願発明の構成要件D～Gの点において相違しており、審決は、一致点の認定を誤り、相違点を看過したといえる。

### ア 構成要件Dについて

引用発明の「コンパイラ」と本願発明の「復号モジュール」が一致するとの認定

がなされているが、当該認定も誤りである。

引用発明のコンパイラは、前記のとおり、アプリケーションを引用発明で動作させるための命令セット・プロセッサ用コードを生成させるもの、すなわち、構成メモリに事前に書き込むコードを生成するものであり、本願発明においてもコンパイラに事前に格納しておく構成命令は、事前にコンパイラにより生成される（甲1の【0026】）。

これに対し、本願発明における復号モジュールは、構成メモリに格納されている構成命令を復号して、相互接続ネットワーク等を構成するものである（甲1の【0018】、【0020】）。

したがって、対比させるとしても、引用発明のコンパイラと本願発明のコンパイラを対象とすべきであって、引用発明のコンパイラと本願発明の復号モジュールを対比させることは誤りである。そして、引用発明の構成に「復号モジュール」に対応する部分はそもそも開示されていないのだから、引用発明の「コンパイラ」と本願発明の「復号モジュール」が一致するとの認定は誤りである。

#### イ 構成要件E2について

引用発明に構成要件E2相当の機能が備わっていることは認めるが、本願発明では、復号モジュールという復号を専用に行うモジュールが直接、構成命令を受けているのに対し、引用発明では構成命令（引用発明では「構成」又は「コンテキスト」とされる。）をどこで受信して、スライスに反映するのかについて開示されていないから、これらが一致するとはいえない。

#### ウ 構成要件F、Gについて

引用発明に構成要件F、G相当の機能が備わっていることを認めるが、それをどのように実現しているかについて記載がないことから、この点で本願発明と異なる。

### 4 取消事由4（本願発明の進歩性についての判断の誤り）

(1) 甲12文献は、本願発明（及び本願発明の先行技術等）が対象とする「再

構成可能なプロセッサ」とは無関係であり、同文献中の中央演算装置（CPU）が再構成可能プロセッサを対象としている旨の記載はない。また、甲12文献には、再構成可能なプロセッサにおいて、二つの命令に対応する各処理（並列処理、従属命令）を担わせる方法もおおよそ開示されていない。

したがって、本願発明のような再構成可能なプロセッサにおいて、「従属プログラム命令」及び「独立プログラム命令」の双方のプログラム命令を取り扱うことが、甲7文献と甲12文献を組み合わせても、容易に想到されるものとはいえない。

(2) 本願発明のRICAは、資源の制約の下で独立命令及び従属命令を実行できるので、費やすのは2クロック・サイクルだけである。したがって、RICAの処理能力は、VLW DSPより高くなる。同じ「周波数」で走る場合や、両者の内部の計算要素の遅れが同じである場合は、RICAの方が、処理能力が高くなる。したがって、「設計が容易で、多様な命令セルの演算のクロック・サイクルに対応できる柔軟性があり高性能な再構成可能プロセッサが得られる」という効果を奏する。

本願発明には、このような特有の独立命令の並列処理、及び従属命令の逐次処理による顕著な効果があり、審決はそれを看過したものである。

(3) したがって、当業者が本願発明を容易に想到できないこと、また、本願発明は、引用発明に対して有利な効果を有していることは明白であり、進歩性が欠如するとした審決は取り消されるべきである。

#### 第4 被告の反論

##### 1 取消事由1に対し

審決は、査定の理由と異なる拒絶の理由によりなされたものではないから、特許法159条2項にいう50条を準用する場合に該当せず、同条の準用はない。

審決の論旨に照らせば、甲12文献は、周知性が高く技術の理解の前提となる知識を補助的に示すものにすぎず、新たな拒絶理由として示されたものではない。

したがって、審判手続に手続違背があるとはいえない。

## 2 取消事由2に対し

### (1) 原告の主張2(1)に対し

審決は、周知性が高く技術の理解の前提となる知識を補助的に示すものとして甲12文献を例示したにすぎず、これに「再構成可能なプロセッサ」についての記載があることを不可欠の前提とした論旨を示したものではないから、原告の主張は、審決を正解しないものであって、失当である。

また、甲7文献の記載内容とその摘示を含めた審決全体の記載に照らせば、審決の記載が省略記載にすぎないことが明らかである。

### (2) 原告の主張2(2)に対し

審決において、引用発明の「主演算装置」が、「構成命令」を「受け」て「復号」して、相互接続ネットワーク及び命令セルの少なくともいくつかのデータ経路を構成する点において、本願発明の「復号モジュール」に対応しており、コンパイラにおける分析により決定されたデータ依存関係（「データ経路」）の「命令セルの回路」への「マップ」のために機能する趣旨において、引用発明の「主演算装置」が本願発明の「復号モジュール」と相当関係にある旨が、事実上示されているから、一部誤解を招きかねない表現があった点では不適切であったが、審決を取り消すべき理由不備があるとまではいえない。

また、プロセッサにおいて実行される命令である以上、その命令の実行によりプロセッサが行う一定の機能が「符号化」されたものであり、このように「符号化」された命令の機能を実現する際「復号」されることは、明示するまでもないことであり、これを明示した上で対比していないからといって、記載不備があることにならない。

### (3) 原告の主張2(3)に対し

請求項29は、本願発明において「マップ」するのが「コンパイラ」であること

を示す本願明細書の記載（【0073】，図6）を受けた技術的事項を記載したものであるところ，審決は，本願明細書の上記記載（【0073】，図6）を示す代わりに請求項29を示したものである。

また，「復号モジュール」が「コンパイラ」を含むとは，そもそも，「復号モジュール」がコンパイラが行う「マップ」のために機能する旨について，本願請求項1の記載の上では，「復号モジュール」が「これにより前記データ経路を前記命令セルの回路にマップ」すると表現されているのであり，これを踏まえて，審決では，「復号モジュール」がコンパイラと関係して機能する趣旨をこのように表現したものである。審決の記載がこれらの趣旨によるものであることは，審決全体の論旨とその記載ぶりに照らして明らかであるから，これらの点についても理由記載の不備はない。

### 3 取消事由3に対し

#### (1) 原告の主張3(1)に対し

審決は，以下に述べるとおり，本願発明の構成要件Gの表現ぶりを踏まえて説示する意図からとはいえ，引用発明について「コンパイラを含む主演算装置」等，技術常識を踏まえずに読むと誤解を招きかねない表現を用いており，表現の上では必ずしも適切でない部分があったが，技術常識を踏まえて通読すれば，以下の(2)に述べる論旨が示されていることは明らかであって，原告は，審決を正解しないものである。

原告も述べるように「プログラム命令を実行するプロセッサ」の「復号モジュール」に「コンパイラ」の機能が含まれるという技術常識はなく，また，審決が摘記した甲7文献の内容から明らかなように，甲7文献は，プロセッサに含まれたコンパイラが，プロセッサの実行する命令を命令実行段階で生成する技術（いわゆるJITコンパイラ等）を記載したものでないから，審決が摘記した甲7文献の記載は，プロセッサが自ら実行する命令を自身が含むコンパイラにより生成する技術に係る

発明を認定する根拠となり得ない。したがって、審決が摘記した甲7文献の内容を踏まえて審決を読めば、審決の「・・・コンパイラを含む主演算装置」等の記載を、プロセッサが自ら実行する命令を自身が含むコンパイラにより生成する旨を示すものと解する余地はない。

(2) 原告の主張3(2)に対し

ア 本願発明の構成要件D～Gと引用発明の対比について

引用発明の「再構成可能なアレイを異なる構成へと統合させるステップを有するプロセッサ用命令セット」である「特殊再構成可能命令」は、再構成に当たって構成(コンフィグレーション, コンテキスト)を示し、コンフィグレーションメモリにない構成を構成メモリへロードする等のプロセッサの機能を符号化したものであり、また、コンパイラがデータ依存関係の分析を行った上で再構成可能アレイを異なる構成へと統合するために、上記分析結果を踏まえて生成されたコードに含まれたものであり、この分析結果を反映した再構成を行うための構成を示すように符号化されたものであるから、「プログラム命令」の「データ経路」の「命令セルの回路」への「マッピング」を「符号化」するものであって、本願発明の「構成命令」に相当する。

そして、引用発明の「プロセッサ」も、コンパイラによるデータ依存関係の分析を経た上で命令実行するものであるところ、その「主演算装置」は、再構成可能な機能ユニットをコントロールする主体として命令キャッシュから読み込んだ「構成命令」を「復号」するものであり、これによって「符号化」された内容に従って相互接続ネットワークと命令セルのデータ経路が構成されることになるから、「構成命令」を「受け」て「復号」して、相互接続ネットワーク及び命令セルの少なくともいくつかのデータ経路を構成する点、すなわち、構成命令による相互接続ネットワークと命令セルの再構成を行う主体となる点において、本願発明の「復号モジュール」に対応している。

また、引用発明の「主演算装置」は、コンパイラがデータ依存関係の分析を行っ

た上で再構成可能アレイを異なる構成へと統合するために、コンパイラにおけるデータ依存関係の分析結果を反映した再構成を行うものであるのに対し、本願発明の「復号モジュール」は、「これにより前記データ経路を前記命令セルの回路にマップ」するものであって、命令実行前のコンパイラにおける分析により決定されたデータ依存関係（「データ経路」）の「命令セルの回路」への「マップ」のために機能するものであるから、両者は、「これにより前記データ経路を前記命令セルの回路にマップ」する点においても共通しているものである。

#### イ 構成要件D～Gに係る審決の対比の論旨について

審決は、引用発明の認定に当たって、主演算装置が「レベル1の命令キャッシュから命令を読み、前記処理エレメント及び相互接続クロスバーと共に設けられたコンフィグレーションメモリへスライスのデータ経路コンポーネントのための構成をロードする」ものであること（以下「前者」という。）に加え、「トライマランをベースとしたコンパイラであって、再構成可能なアレイを異なる構成へと統合させるステップを有するプロセッサ用命令セットを生成するコンパイラを含む」（以下「後者」という。）とした。そして、引用発明の「主演算装置」と本願発明の「復号モジュール」との対比の説示では、まず、前者に係る対比において、これ以前の「プロセッサ」、「相互接続ネットワーク」及び「複数の命令セル」に係る対比における、「クロスバー」及び「処理エレメント」がそれぞれ本願発明の「相互接続ネットワーク」及び「複数の命令セル」に相当する旨の説示を踏まえ、引用発明の「クロスバー」及び「処理エレメント」以外の対象物である「主演算装置」へ「命令キャッシュ」から命令が供給される旨を示している。そして、審決では、引用発明の「プロセッサ」においては、コンパイラにおける分析結果を反映した再構成を行うための構成を示す「構成命令」による再構成可能な機能ユニットのコントロールを行う主体が「主演算装置」であるという前提を踏まえて説示されており、命令が供給された「主演算装置」は、供給された命令による「相互接続ネットワーク」と「複数の命令セル」の再構成を行うことになる。

そして、後者に係る対比に当たって、引用発明の「コンパイラ」を対比において参照する旨を示した上で、本願発明の「復号モジュール」が行うとする特定事項の中に「前記相互接続ネットワークおよび命令セルの少なくともいくつかのデータ経路を構成し、これにより前記データ経路を前記命令セルの回路にマップした前記プログラム命令を実行する」との事項が、ハードウェアに対する命令／処理の実行とされている旨を示しつつ、「プログラム言語命令をハードウェア実行コードに変換する」点で、引用発明の「主演算装置」と本願発明の「復号モジュール」の両者が一致する旨を説示している。

この後者に係る対比の説示のうち「プログラム言語命令をハードウェア実行コードに変換する」点で本願発明と引用発明とが一致する旨の説示は、本願発明の「プログラム命令を実行するプロセッサ」が、データ依存関係の分析のみならずその分析結果の「命令セルの回路」への「マップ」がコンパイラにおいて命令実行前に行われることを前提としたものであることを踏まえ、プログラム言語を用いて作成されたプログラム（プログラム言語命令）をハードウェアにおいて実行可能な命令の列としてのプログラム（ハードウェア実行コード）に変換する際のコンパイラにおける分析を前提としたものである点において、引用発明の「主演算装置」が本願発明の「復号モジュール」と相当関係にある旨を述べているものである。

なお、命令処理資源（「命令セル」）をどのように割り当てるかの決定は、命令実行前にコンパイラにおいてデータ依存関係の分析の結果を踏まえて（静的に）行われることが前提とされているにもかかわらず、本願特許請求の範囲においては「前記データ経路を前記命令セルの回路にマップ」する（構成要件G）と記載されており、あたかもハードウェアである復号モジュールが主体的に割り当てを行うかのように解釈することもできる。審決は、このことを踏まえて、本願発明の「復号モジュール」の特定事項中に「・・・これにより前記データ経路を前記命令セルの回路にマップした前記プログラム命令を実行する」との事項が、ハードウェアに対する命令／処理の実行とされている旨を示し、引用発明の「コンパイラ」を対比におい

て参照する趣旨を示したものである。

#### 4 取消事由4に対し

##### (1) 原告の主張4(1)に対し

審決は、甲7文献において「マッピングにより並列処理が可能」であることを摘示した上で、コンパイラにおけるデータ依存関係の分析により依存関係がないとされたものについて、複数の命令が並列実行可能であることが技術常識（周知技術）であることの例示として甲12文献を示したにすぎない。

したがって、甲12文献が「再構成可能なプロセッサ」について記載していないことは、審決の容易想到の論旨とは無関係である。

##### (2) 原告の主張4(2)に対し

処理能力について、引用発明は、「従属」する命令を先行命令の結果の書込みを待たず並列に実行し得るのであり、本願発明と同様、「従属」命令につき先行命令の結果の書込みを待つ必要がある（「従属」命令につき命令レベルの並列化が行われ得ない。）VLIWに比して、処理能力が高くなり得るという効果を奏するものである。

また、「設計が容易で、多様な命令セルの演算のクロック・サイクルに対応できる柔軟性があり高性能な再構成可能プロセッサが得られる」という効果については、本願明細書の記載、特に段落【0041】～【0046】の記載によれば、「多様な命令セルの演算のクロック・サイクルに対応できる柔軟性」は、「再構成レート・コントローラ（RRC）24」の効果であるから、原告の主張は、本願特許請求の範囲に基づくものでない。

#### 第5 当裁判所の判断

##### 1 本願発明について

(1) 本願明細書（甲1）及び本件補正書（甲4）によれば、本願発明は、以下のとおりのものである。

本願発明は、コンピュータ・プロセッサ、特に、再構成可能プロセッサに関するものである（【0001】）。

現在の半導体デバイスの構造は、大量のオーディオ及びビデオ・データを処理する必要のある次世代ネットワーク用の移動体装置におけるなど、将来のデータ処理及び処理能力の要求に対処できないので、将来の新しいハードウェア構造として、電力消費及びシリコン面積に関して効率を犠牲にしても高い処理能力を保持するために、プロセッサの性能に対する新たな要求に応え、ハードウェアも将来の変化する標準に対して高い適応性を備える必要がある（【0002】）。

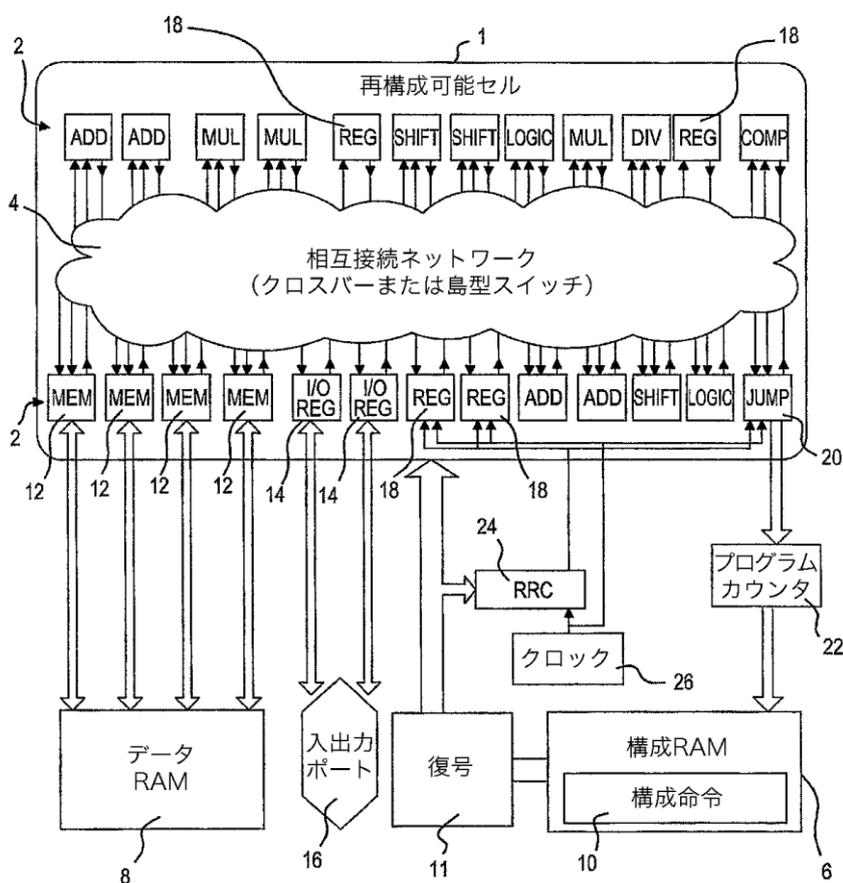
ハードウェアの再プログラム可能性及び柔軟性は、製作後にシステムを変更することができるので、設計コストを下げることができ、標準C/C++などの高水準プログラム記述によりプログラム可能なこの柔軟性を有することは、設計サイクルを減らし、また、大きなシステムを実現するのに重要である（【0003】）。

提案されている書き替え可能ゲート・アレイ（FPGA）などの再構成可能なハードウェア・インフラストラクチャは、コストが低く、設計時間が短く、電力消費が少なく、性能が高い、というニーズに十分応えておらず、どれも広く用いられていないため、本願発明は、コストが低く、設計時間が短く、電力消費が少なく、性能が高いプロセッサを提供することを目的とする（【0004】～【0016】）。

そこで、本願発明は、従属及び独立の両方のプログラム命令のデータ経路を有するプログラム命令を実行するプロセッサを提供するものであり、プロセッサは、相互接続ネットワークと、それぞれが相互接続ネットワークに接続される異質の複数の命令セルと、復号モジュールであって、従属プログラム命令のデータ経路と独立プログラム命令のデータ経路の少なくとも一つから命令セルの回路へのマッピングをそれぞれ符号化された構成命令を受け、更に構成命令を復号して相互接続ネットワーク及び命令セルの少なくともいくつかを構成し、これによりデータ経路を命令セルの回路にマップし、また、プログラム命令を実行する、復号モジュールと、を備えるものである（請求項1，【0018】，図1参照）。

これにより、再構成可能プロセッサは、設計時に完全にカスタマイズすることができ、アプリケーションの要求に従って設定することができ、また、実行の並列処理レベルを高め、操作を終えるのに必要な時間を減らすことができる（【0037】、【0046】）。

【図1】



(2) 本願発明の復号モジュール及びコンパイラについて

本願明細書においては、構成命令の生成に関し、プログラミング・フロー（図5）及び設計法の略図（図6）が示され、標準GNU Cコンパイラ（gcc）を用いてC/C++コード50をコンパイルして、どの命令セル（IC）を用いる必要があるかを記述するアセンブリ・フォーマット52を作ること、RICAコンパイラは、このアセンブリを処理して、一連のネットリスト54を作り、各ネットリストは1

クロック・サイクル内に実行する命令のブロックを含み、ネットリストへの分割は命令をスケジューリングして命令の間の依存性を分析した後に行うこと、ネットリスト内では、従属プログラム命令は直列に接続され、独立プログラム命令は並列に接続され、スケジューリング・アルゴリズムはアレイ内の I C 資源や相互接続資源やタイミング制限を考慮に入れ、最も多くの I C を用い、また、同時に最長経路遅れを減らして最小にすることにより、プログラム処理能力を最高にすることを旨とし、さらに、R I C A コンパイラは、g c c が生成した一時的レジスタを簡単な配線で置き換えて除くなどの厳しい最適化も行い、一連の命令セル 5 6 の構成を生成し、これを構成命令 6 として符号化して構成メモリ 1 0 内に記憶することが示されている (【0 0 5 2】、【0 0 5 3】、【0 0 5 9】)。そして、復号モジュールは、従属プログラム命令のデータ経路と独立プログラム命令のデータ経路の少なくとも一つから命令セルの回路へのマッピングをそれぞれ符号化する構成命令を受け、更に構成命令を復号して相互接続ネットワーク及び命令セルの少なくともいくつかを構成し、これによりデータ経路を命令セルの回路にマップし、プログラム命令を実行するものである (【0 0 1 8】)。また、命令セルの少なくともいくつかを相互接続ネットワークを通して直列に接続することにより、相互接続ネットワーク及び命令セルの少なくともいくつかを構成したり (【0 0 1 9】)、複数のプログラム・スレッドのデータ経路から命令セルの対応する複数の独立回路へのマッピングを符号化する構成命令を受け、更に構成命令を復号して相互接続ネットワーク及び命令セルの少なくともいくつかを構成し、これにより複数のプログラム・スレッドのデータ経路を命令セルの対応する複数の回路にマップし、プログラム・スレッドを互いに独立に同時に実行したりする (【0 0 2 0】) ものである。

これによれば、構成メモリに格納された構成命令は、R I C A コンパイラによって生成され、符号化されたものであるところ、復号モジュールは、この符号化された構成命令を受けて復号化し、命令セルの少なくともいくつかを相互接続ネットワークを通して直列に接続したり、複数のプログラム・スレッドのデータ経路から命

令セルの対応する複数の独立回路へのマッピングを符号化された構成命令を受け、更に構成命令を復号して相互接続ネットワーク及び命令セルの少なくともいくつかを構成し、これにより複数のプログラム・スレッドのデータ経路を命令セルの対応する複数の回路にマップし、プログラム・スレッドを互いに独立に同時に実行したりするなど、相互接続ネットワーク及び命令セルの少なくともいくつかを構成し、これによりデータ経路を命令セルの回路に現実にマップし、プログラム命令を実行するものであるといえる。

## 2 取消事由3（引用発明の認定及び対比判断の誤り）について

### (1) 引用発明について

ア 甲7文献には、以下の記載がある。

(ア) 表題「低消費電力 粗粒度結合型再構成可能なインストラクションセットプロセッサ」

(イ) 「概要：近年の組み込み型マルチメディアのアプリケーションは、処理時間および消費電力の制約が厳しい。要求性能を満たす粗粒度型プロセッサについて紹介されてきた。しかしながら、このようなプロセッサの消費電力についてはあまり研究がなされていない。本論文では、画期的な粗粒度結合型プロセッサと、WATTCHから導き出された消費電力モデルをつかった消費電力についての研究を紹介する。いくつかのプロセッサコンフィギュレーションにおいて、複数のマルチメディアアプリケーションを評価した。評価結果から、本論文の粗粒度結合型プロセッサは、RISCプロセッサ単体に対して平均2.5倍の処理能力を達成したが、消費電力は18%増加に留まる。」

### (ウ) 「1. イントロダクション

3Dレンダリング、映像圧縮、物体認識技術などの現在および将来のマルチメディアアプリケーションは、非常に複雑な計算を必要とするアルゴリズムによって特徴づけられ、深いネストループ構造（何重ものループになっている構造）は、非常に厳しいリアルタイム処理が必要となる。組み込みシステムにおいて（例えば、マルチメディア端末、PDA、

携帯電話など),このようなアプリケーションを実行するには,実行時間と面積の制約下で,消費電力を最適化してゆくとの問題が生じる。これらのアプリケーションでのループ処理において可能な限り並列性を高くすることで,再構成可能なインストラクションセットのプロセッサ(標準のプロセッサと再構成可能ないくつかのロジックが密結合したものは,時間的な制約を満たす計算能力を持つことができる,といくつかの論文で紹介されている[7, 6, 4]。しかしながら,上記の再構成可能なプロセッサはこれらのアプリケーションに要求される消費電力を満たすか否かについては,示されていない。

本論文は, CRISP(a coarse-grained reconfigurable instruction set processor, 粗粒度結合型インストラクションセットプロセッサ)について示す。CRIPS は, 低消費電力でマルチメディア処理を高速に行うことができるよう, マルチメディアのアプリケーション向けに開発されたものである。本アーキテクチャの消費電力は, 再構成可能なロジックに起因するものである。ここで再構成可能なロジックは, ALU や乗算器などの複雑な処理を行うブロックにより構成され, マルチメディアアプリケーションにおいて典型的に見られるデータサイズ(8ビット~32ビット)で動作し, 消費電力と再構成期間を減らすために, 独立して動作可能化できるスライス単位に分割される。さらに重要なことは, 再構成可能なロジックはメインプロセッサに密結合していて(再構成可能なロジックは, 追加機能ユニットとして位置づけられる), プロセッサと再構成可能なロジックの間で瞬時的な制御とデータ通信が可能になる。

...

本論文では, 以下の順に説明を行う。第2章および第3章では, 提案の CRISP アーキテクチャと関連するコンパイラ技術/合成技術について議論する。...

#### (エ) 「2. 低消費電力再構成可能なアーキテクチャ

CRISP (Coarse-grained Reconfigurable Instruction Set Processor 粗粒度結合再構成可能なインストラクションセットプロセッサ) は, メインのプロセッサコアといくつかの粗粒度結合した再構成可能なロジックからなるインストラクションセットプロセッサからなる。粗粒度再構成可能なロジックは, 他の機能ユニットと同様に再構成可能な機能ロジック

ク (RFU) の中に位置し、動作は毎クロック発行される。RFU は、メインのレジスタファイルに対しデータのアクセス (読み書き) を行う。メインプロセッサは、どのようなタイプのプロセッサでもよいが、本論文では、単純な RISC (Reduced Instruction Set Computer) プロセッサを仮定する。

図 1 は、完全なプロセッサの全体を示す設計思想 (=アーキテクチャー) である。主演算処理装置コアはレベル 1 の命令キャッシュからその指示を読み、レベル 1 のデータキャッシュによってデータを得る。双方のキャッシュは、外部メモリに接続され、かつ、レベル 2 の統合キャッシュに接続される。図の中央に図示され、かつ、直接主演算処理装置コアによってコントロールされている、再構成可能な構成部は、2 次キャッシュ (これは、外部メモリからロードした構成の再使用を可能とし、再構成を行う回数を減らす) によってロードされるコンフィギュレーション (=構成) メモリを含んでいる。再構成可能な構成部は、直接いくつかのデータポート経由でデータキャッシュにアクセスすることができる。さらに、再構成可能なロジックは機能ユニット・インターフェース経由で主演算処理装置コアと通信する。

図 1 に示されるように、再構成可能な機能ユニットは再構成可能なスライスに分割される。そのうちの 1 つは、図 2 によってより詳細に示されている。各々のスライスは、いくつかの粗粒子状の処理エレメント (PE) や、レジスタファイルや、相互接続や、小さなコンフィギュレーション (構成) メモリを含んでいる。各々の処理エレメントは、ALU や、シフタや、乗算器や、あるいはメモリ素子である。こういった複雑な処理エレメントは、マルチメディア・アプリケーションの分野で典型的に見られる操作の実行向けとして、ワード・オリエンティドであって、ビット・オリエンティドではないルックアップテーブル (LUT) をベースとした従来の論理ブロックよりも、適したものになる。これらの処理エレメント (PE) 複合体は、従来の FPGA と比較すると、より低い電力消費量で、かつ、より高周波数で作動することを可能とする、再構成可能ロジックになる。

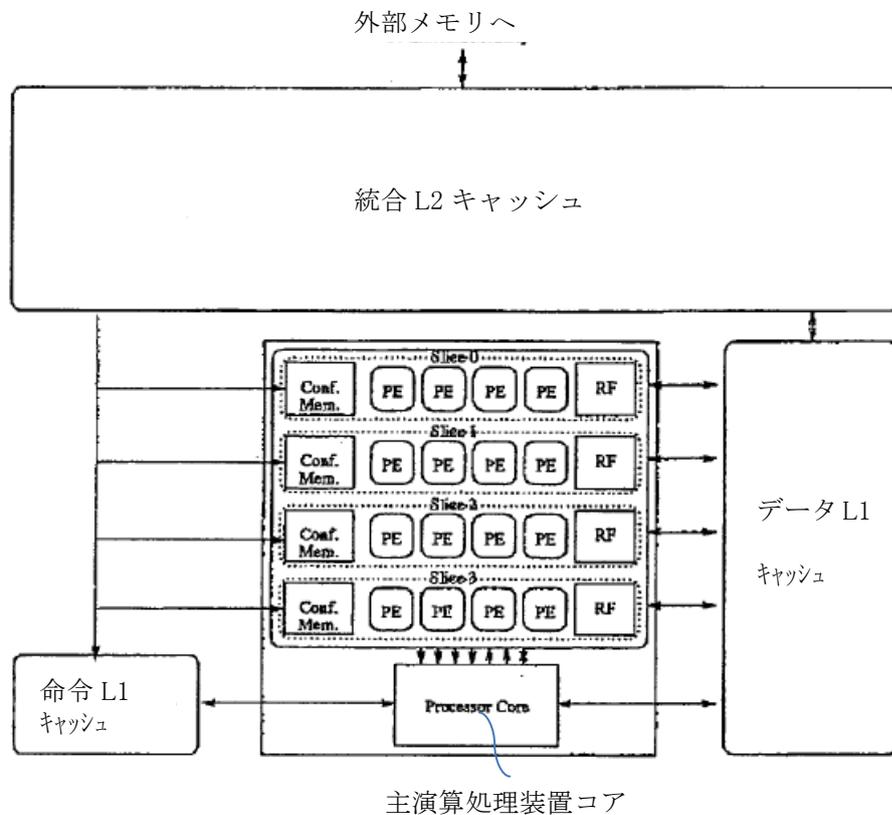


図 1. CRISP の例 (RFU: 再構成可能な機能ユニット, PE: 処理エレメント, FU: 機能ユニット)

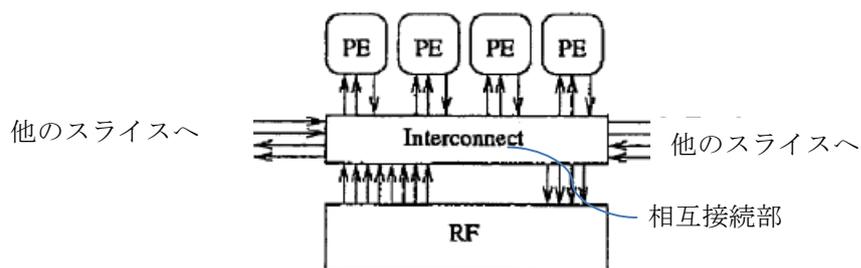


図 2. 再構成可能なスライスの内部

スライスの内部にある処理エレメントは、プログラム可能な相互接続部を通じて、互いに接続状態にある。この相互接続部は、完全なクロスバーでワード単位に作動し、VLIW (超

長命令語) マイクロプロセッサで今日典型的に見られるバイパス・ネットワークと同程度の複雑さがある。このクロスバーは、任意の処理エレメントの結果出力を、その他の処理エレメントの入力へと接続することができる。加えて、このクロスバーは、処理エレメントを、レジスタファイルや、他のスライスへと接続もする。ほとんどの場合、処理エレメントはそれぞれスライスのレジスタファイルにその出力を書き込む。しかし、従来の FPGA の中で見られるのと同じで、この動作は任意にバイパスすることができ、また、結果は異なる処理エレメントに送られる。この他のスライスのレジスタ書き込みと、相互接続クロスバーを組み合わせることによって、データ・フロー鎖の中の要素がクロスバーによってともに接続されるように、空間的な計算を行なうことが可能である。チェーンの終端の処理エレメントはレジスタファイルに接続される。

処理エレメントの数は、相互接続とレジスタファイルの複雑さを縮小するために少なくしておかなければならない。私たちのシミュレーションの結果では、処理エレメントとの数として4が、電力消費量対パフォーマンスの間で形成されるトレードオフの関係上、良い数値になる（スペースの理由で、これらの結果はここでは示さず）。

再構成可能なスライスはそれぞれさらに構成メモリを含んでいる。この構成メモリは、スライスのデータ経路コンポーネントのための構成を格納する。典型的なループ処理は、ディレイなく切り替えるため複数のコンフィグレーションを用意する必要があり（第3章で議論されるが）、そのためコンフィグレーションメモリは、マルチコンテキストである必要がある（すなわち、いくつかのコンフィグレーションを保持しておく必要がある）。1つのコンテキスト（又はコンフィグレーション）を別のコンテキストに切り替えるのに1クロックサイクルで可能であり、すなわち、浅くて（アドレス線方向の幅がせまい）、ワイド幅な（データ線方向に幅が広い）メモリから読むのと同じである。4つの処理エレメントをもったスライスの場合、コンフィギュレーションメモリの幅は、128ビット程度であり、典型的なFPGAのスライスに必要なビット幅よりかなり少ない。コンフィギュレーションメモリ内のコンフィギュレーションの数は、通常8～32のコンテキストの範囲である。理想的には、コンテキストの数は、コンフィギュレーションメモリの消費電力削減のため、

できるかぎり少ない方がよい。

再構成可能な機能ユニットは、図3に示されるような特殊再構成可能命令により活性化される。この再構成可能命令は、主に2種の情報からなる。1つは、多くの利用可能な構成のどれが使用されるべきかを指定する再構成可能命令識別子 (RID) を含んでいるということである。この識別子は、構成メモリにおいて利用可能なコンテキストの数より多数の構成の中から選択することができる。必要な構成が構成メモリに現在ロードされない場合、システムは停止される。それはこの RID によってインデックスを付けられた小さなキャッシュのように作用する。また、適切な構成は一体になった2次キャッシュからロードされる。

再構成可能命令識別子 RID に加えて、再構成可能命令は、どのスライスが活性化されるかを指定する1ビット長の数個のフィールドを含んでいる。図3は、これらのスライス・イネーブル・フィールド (図中では、EN<sub>x</sub>と名付けられている) を示している。少数の処理エレメントを要求するアプリケーションの部分については、最初のスライスだけが活性化されるだろう。より高い並列必要条件を備えたそれらの部分については、より多くのスライスが使用されるだろう。このメカニズム (それは部分的な再構築の形式と見なすことができる) は、構成ミスの場合にはロードされなければならない構成流れのサイズを縮小する。さらに加えて、必要とされない再構成可能なデータ経路のスライスは、スイッチをオフとすることができ、それにより、エネルギー消費コントロールの有効な形を提供することができる。

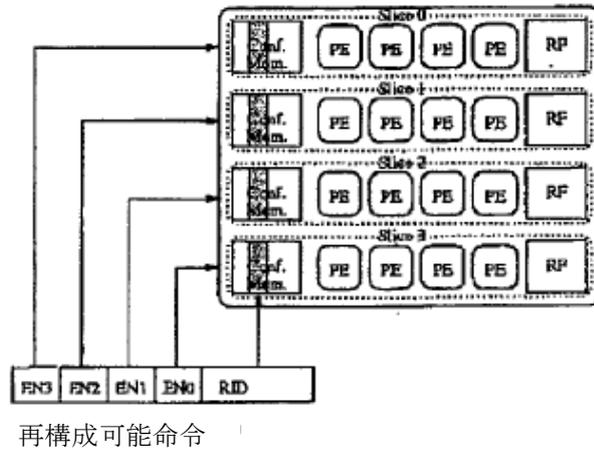


図 3. 再構成可能命令のフィールド EN: スライス・イネーブル・ビット, RID: 再構成可能命令識別子」

(オ) 「3. コンパイラ技術

全ての再構成可能なインストラクションセットプロセッサの命令生成は、主に次の2つのステップを含む。

(一つは) 再構成可能アレイ用の様々なコンフィギュレーションの合成であり、(もう一つ) は、メインプロセッサ用の命令生成 (再構成可能なアレイへマッピングされない命令生成) である。CRISP の場合、標準の機能ユニットと同様に複雑な要素を処理するので、従来の VLIS 技術が使われている。

我々の研究用コンパイラ (Trimaran[1]をベースにしている) においては、ループの命令生成は、ソフトウェアのパイプライン処理に基づいている。ソフトウェアのパイプライン処理では、イタレーション (繰り返し) は、一定の間隔をあけて開始され、同時に、しかし、計算の異なるステージにおいて実行される。これらにより、CRISP のハードウェアリソースに可能な限り並列性を展開する。これらの技術[2]により、ループ向けに生成された命令は、ループ (ループカーネルのサイクル数) の初期化間隔分のコンフィギュレーションを含んでいる。そのため、繰り返しが、コンフィギュレーションのキャッシュで可能なコンテキストの数以上にならないことをチェックすることは重要である。超えてしまった場

合、生成された命令は、不断の再構成が必要で、処理性能は落ちるであろう。

ソフトウェアのパイプラインは、命令をチェインすることで[2]、空間的な計算を行うために修正することができる。これにより、実行時間を削減することで、内側のループのクリティカルパスの長さを削減できる。空間的な計算の命令生成処理は、処理構成部と相互連結の時間の適切な遅延モデルが必要である、なぜならこの処理は、FPGA の “place and route” (配置配線) と同じであるからである。

さらに、我々のコンパイラは、各ループ処理を分析し、必要なスライス数を割り出す。再構成時間と消費電力の双方を削減するため、必要なスライス数しか使わない。」

#### (カ) 「4. 実験セットアップ

本提案のアーキテクチャでの性能および消費電力を評価するために、シミュレーションにてマルチメディアアプリケーションをいくつか動作させてみた。表1に、ベンチマークのリストを示す。コンパイラは、前セクションで説明した”Compilation Techniques”を含んだ Trimaran コンパイラの改良したバージョンでコンパイルした。本コンパイラは、再構成可能なプロセッサにおいて、粗粒度結合型再構成可能なユニットを利用して開発することができる。特別な計算手法は、我々のプロトタイプのコパイラには現状組み込んではいないため、使用してはいない。」

(キ) 「イントロダクションで記述したように、本提案のアーキテクチャは、低消費電力である。その理由は以下にいくつかあげる。

— 粗粒度のデータパスの演算部：データパスの演算部は、ビットレベルの構成要素よりも、制御のオーバーヘッドが少なく済むため、より消費電力の面で効率的である。

— 少量およびスライスされたコンフィギュレーションメモリ部：コンフィギュレーションメモリを定期的にスイッチングすることは必要であるが、コンフィギュレーションメモリに対する消費電力は全体に比較して微量である。さらに、コンフィギュレーションメモリがスライス化していることで、選択的に動作することになり、より低消費電力を実現できる。

— スライスされたデータパスとすることによって、異なる複数の要素の複雑さ及び消費

電力が制限される。

— 消費電力を考慮し、さらに処理性能を低下することなく動作させるスライス数を最小限にするインテリジェントコンパイラ。」

(ク) 「7. 結論

本論文では、低消費電力マルチメディアアプリケーション用の粗粒度結合アーキテクチャについて言及した。再構成可能なロジックは、プロセッサの消費電力を削減することを独立に動作させるためにスライスに分割している。本プロセッサは、標準の RISC プロセッサと比較して、18%消費電力が増加するが平均 2.5 倍の処理性能が得られることが分かった。・・・」

イ 上記記載によれば、引用発明について、次のとおり認められる（甲7）。

甲7文献は、粗粒度結合型再構成可能なインストラクションセットプロセッサ（CRISP）の消費電力について研究した結果を記載した論文である。

近年、組込型マルチメディアのアプリケーションは、処理時間及び消費電力の制約が厳しく、このような要求性能を満たす粗粒度結合型プロセッサが開発されてきた。例えば、マルチメディア端末、PDA、携帯電話などの組込型システムにおいて、3Dレンダリング、映像圧縮、物体認識技術などのマルチメディアアプリケーションは、非常に複雑な計算を必要とするアルゴリズムによって特徴付けられ、深いネストループ構造は、厳しいリアルタイム処理が必要となる。これらのアプリケーションでのループ処理において可能な限り並列性を高くすることで、再構成可能なインストラクションセットプロセッサ（標準のプロセッサと再構成可能ないくつかのロジックが密結合したもの）は、時間的な制約を満たす計算能力を持つことができる。

引用発明は、低消費電力でマルチメディア処理を高速に行うことができるよう、マルチメディアのアプリケーション向けに開発されたCRISP（粗粒度結合型再構成可能なインストラクションセットプロセッサ）であり、これは、単純なRISCプロセッサであるメインプロセッサ（主演算処理装置コア。以下「主演算装置」

という。)と、いくつかの粗粒度結合した再構成可能なロジックからなるインストラクションプロセッサとで構成される。

図1に示されるように、主演算装置は、レベル1の命令キャッシュからその指示を読み、レベル1のデータキャッシュによってデータを得、双方のキャッシュは、外部メモリに接続され、かつ、レベル2の統合キャッシュに接続される。そして、主演算装置により、直接コントロールされている再構成可能な構成部は、2次キャッシュ（外部メモリからロードした構成の再使用を可能とし、再構成を行う回数を減らす。）によってロードされるコンフィグレーション（構成）メモリを含み、直接、いくつかのデータポート経由でデータキャッシュにアクセスすることができ、さらに、再構成可能なロジックは、機能ユニット・インターフェース経由で主演算装置と通信する。

再構成可能な機能ユニットは、再構成可能なスライスに分割され、各々のスライスは、いくつかの粗粒子状の処理エレメント（PE）、レジスタファイル、相互接続や、小さなコンフィグレーションメモリを含んでいる。

各々の処理エレメントは、ALU（算術論理演算）、シフタ、乗算器、あるいはメモリ素子であり、これらの処理エレメント複合体は、従来のFPGAと比較すると、より低い電力消費量で、かつ、より高周波数で作動することを可能とする、再構成可能ロジックになる。これらの処理エレメントは、プログラム可能な相互接続部を通じて、互いに接続状態にあり、この相互接続部は、完全なクロスバーでワード単位に作動し、このクロスバーは、任意の処理エレメントの結果出力を、その他の処理エレメントの入力へと接続することができ、この他のスライスのレジスタ書込みと、相互接続クロスバーを組み合わせることによって、データ・フロー鎖の中の要素がクロスバーによって共に接続されるように、空間的な計算を行うことが可能である。

また、スライスにあるコンフィグレーションメモリは、スライスのデータ経路コンポーネントのための構成を格納している。典型的なループ処理は、遅滞なく切り

替えるために複数のコンフィグレーション（構成）を用意する必要があり，そのため，コンフィグレーションメモリは，マルチコンテキストである必要がある。再構成可能な機能ユニットは，コンフィグレーションメモリにロードされた構成を用いることができる。

以上によれば，引用発明は，

「プロセッサであって，

プログラム可能な相互接続を提供するクロスバーと，

それぞれがクロスバーに接続される処理エレメントであって，処理エレメントは，

A L U（算術論理演算），シフタ，乗算器を含んだ処理エレメントと，

処理エレメント及びクロスバーと共に設けられ，スライスのデータ経路コンポーネントのための構成をロードされたコンフィグレーションメモリと，

クロスバー及び処理エレメントを直接コントロールする主演算装置と，

を備えるプロセッサ。」であると認められる。

## (2) 審決の引用発明の認定について

審決は，引用発明について，「・・・前記主演算装置は，トライマランをベースとしたコンパイラであって，再構成可能なアレイを異なる構成へと統合させるステップを有するプロセッサ用命令セットを生成するコンパイラを含む主演算装置・・・」と認定し，主演算装置に「・・・プロセッサ用命令セットを生成するコンパイラ」を含むとしていることから，引用発明における「コンパイラ」について検討する。

ア まず，前提として，P L D（プログラマブル・ロジック・デバイス又はプログラマブル・デバイス）の分野における「コンパイラ」について検討する。

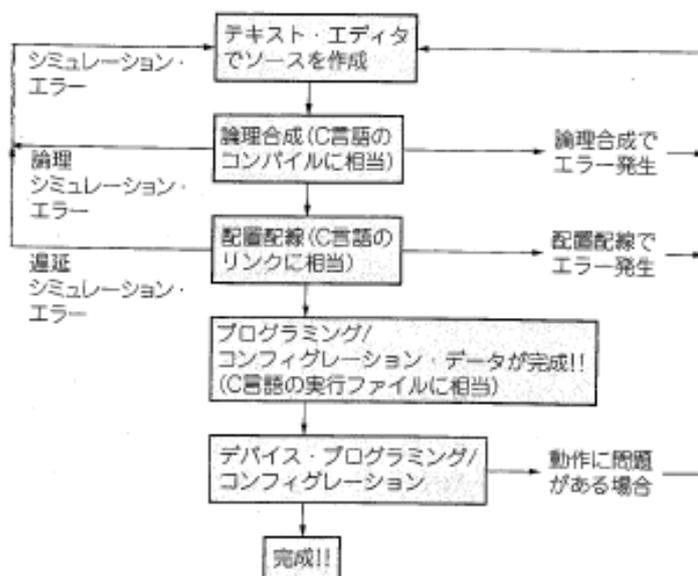
証拠（甲 1 9， 2 0）及び弁論の全趣旨によれば，P L Dの開発に関し，以下のとおり認められる。

P L Dは，ユーザが設計した任意のロジック回路を書き込んで，動作させることのできるデバイスである。P L Dの一つが，F P G A(Field Programmable Gate Array)であり，L U T (Look Up Table)と呼ばれる入力の組合せによってあらかじめ決めら

れた出力が得られるメモリを用いるものであり，ハードウェア記述言語（HDL;Hardware Description Language)を用いて作成したハードウェアイメージを専用のデバイス（FPGA）に書き込むだけでオリジナルのハードウェアを作ることができる。このLUTは，構造がSRAMと似ており，高集積化しやすい一方，回路情報を記憶するベースがSRAMベースであるために，電源を切ると回路情報が失われることから，ROMを外付けし，電源投入と同時に回路情報をFPGAにダウンロードする必要がある。この動作をコンフィグレーションと呼び，PLDの開発段階で，ROMに格納するコンフィグレーション・データを作成しておく必要がある。

PLDの開発（FPGA等の各テーブルに設定する内容の決定）は，通常のスフトウェアの開発同様，開発用言語を利用して行われ，開発の手順としては，一般に，①ソース・コードの作成，②論理合成，③配置配線，④コンフィグレーション・データの生成，⑤デバイス・プログラミング/コンフィグレーションにより行われる。

例えば，甲19の図4には，以下のように示されている。



すなわち，まず，ソース・コードの作成が，HDLなどの開発用言語の文法に従って行われ(上記①)，次に，開発用ツール（ソフトウェア）によって，ソース・コードからネットリスト（回路構成情報）を生成する論理合成が行われ（上記②），こ

ここで生成されたネットリストに従い、実際の素子に組込み可能なように配置と配線を決める配置配線が行われ（上記③）、この配置配線の結果は、コンフィグレーション・データ（又はビット・ファイル）として生成され（上記④）、開発用パソコンの中に保管されているコンフィグレーション・データがケーブルを介して、FPGA等のPLDが搭載されたボードのPLD又はROMにダウンロードされるという、デバイス・プログラミング／コンフィグレーション（FPGAへのデータ書込み）の操作が行われる（上記⑤）。

このように、ソースコードは、HDLなどの高級言語で作成されており、これを0と1とからなる機械語（アセンブラ）に変換すること、あるいは、プログラミング言語で書かれた、プログラムのソースコード（原始コード）を、機械語、ないしバイトコードなどの中間言語によるオブジェクトコード（目的コード）に翻訳（変換）することを、「コンパイル」、その動作をさせるソフトウェア又は場所を、「コンパイラ」と呼ぶ（以下、「第1の意義」という。）一方、PLDの開発手順における②論理合成、又は②論理合成と③配置配線とを合わせて、上記と同様に「コンパイル」、「コンパイラ」との用語が用いられる（以下「第2の意義」という。）。

イ 甲7文献に記載された「コンパイラ」について

上記(1)ア(ウ)のとおり、甲7文献において、コンパイラ技術は、「コンパイラ技術／合成技術」と記され、また、上記(1)ア(オ)のとおり、「3 コンパイラ技術」として、すべての再構成可能なインストラクションセットプロセッサの命令生成は、主に二つのステップを含み、そのうちの 하나가再構成可能アレイ用の様々なコンフィグレーションの合成であること、甲7文献の研究で用いられたコンパイラは、トライマランコンパイラをベースとしており、再構成可能なプロセッサにおいて、粗粒度結合型再構成可能なユニットを利用して開発することができ、ループの命令生成は、ソフトウェアのパイプライン処理に基づいており、この処理は、「FPGAの“place and route”（配置配線）と同じである」こと、また、このコンパイラは、各ループ処理を分析し、必要なスライス数を割り出すことができ、再構成時間と消費

電力の双方を削減するため、必要なスライス数しか使わない、「消費電力を考慮し、さらに処理性能を低下することなく動作させるスライスの数を最小限にするインテリジェントコンパイラ」であること（上記(1)ア(キ)）が示されている。

ウ 以上によれば、甲7文献に記載された「コンパイラ」は、PLDの開発段階で、ROMに格納するコンフィグレーション・データを作成するために用いられるものであり、上記の第2の意義を示すものと認められる。

そうすると、前記のとおり、引用発明の「プロセッサ」は、スライスのデータ経路コンポーネントのための構成をロードされたコンフィグレーションメモリを備え、「主演算装置」は、クロスバー及び処理エレメントを直接コントロールするものであるところ、甲7文献の「コンパイラ」は、PLDの開発手順において、上記コンフィグレーションメモリへロードされるデータコンポーネントを生成するために用いられるものであるから、審決の述べるように、引用発明の主演算装置に「トライマランをベースとしたコンパイラであって、再構成可能なアレイを異なる構成へと統合させるステップを有するプロセッサ用命令セットを生成するコンパイラを含む」と解する余地はない。

したがって、審決の引用発明の認定は誤りである。

(3) 被告は、命令処理資源（「命令セル」）をどのように割り当てるかの決定は、命令実行前にコンパイラにおいてデータ依存関係の分析の結果を踏まえて（静的に）行われることが前提とされているとして、「コンパイラ」が命令実行前に命令セルの割当てを決定するものであることを認めつつ、審決の認定について、上記のような引用発明の「コンパイラ」を参照しつつ、引用発明の「主演算装置」と本願発明の「復号モジュール」の両者が一致する旨を説示したものであると主張する。

被告の主張は、その趣旨が必ずしも明らかではないが、認定した引用発明における「コンパイラ」は、「甲7文献に記載された『コンパイラ』を参照し」たにすぎないもので、主演算装置に「コンパイラ」を含むと認定したものではない旨述べるものと解される。

しかし、審決が認定した引用発明は、その字義から、主演算装置が「コンパイラ」を含むと認定したことが明らかである。

そして、審決は、前記(1)ア(㊦)の部分を摘記し、「『命令』を処理する『主演算装置』が行う処理の一部が、上記C（裁判所注：前記(1)ア(㊦)の部分）に記載された『コンパイラ』であると看取できる。」と述べ、引用発明について「・・・コンパイラを含む主演算装置・・・」と認定し、対比において、引用発明の「コンパイラ」の元々の意味がプログラミング言語で書かれた、プログラムのソースコード（原始コード）を、機械語、ないしバイトコードなどの中間言語によるオブジェクトコード（目的コード）に翻訳(変換)する機能単位を指す用語であるとした上で、引用発明の「コンパイラ」と本願発明の「復号モジュール」について、「プログラム言語命令をハードウェア実行コードに変換するもの同士である点で、両者は一致する」と認定し、一致点を「・・・復号モジュールと、を備えるプロセッサ」と認定しているのである。

以上によれば、審決は、甲7文献に記載された「コンパイラ」が、PLD開発手順における構成命令生成段階における上記の第2の意義の「コンパイラ」であるのに、その技術的意義を誤って理解し、上記第1の意義における「コンパイラ」と同様に捉え、引用発明の認定において、後記(5)のとおり、構成を復号化するモジュールが甲7文献に記載されていないにもかかわらず、第1の意義の「コンパイラ」が主演算装置に含まれると認定したと解される。この点において、審決は、引用発明の認定を誤ったものといわざるを得ず、被告の主張は採用できない。

(4) 被告は、審決のした対比について、「プログラム言語命令をハードウェア実行コードに変換する」点で、引用発明の「主演算装置」と本願発明の「復号モジュール」の両者が一致する旨を説示したもので、本願発明の「プログラム命令を実行するプロセッサ」が、データ依存関係の分析のみならず、その分析結果の「命令セルの回路」への「マップ」がコンパイラにおいて命令実行前に行われることを前提としたものであることを踏まえ、プログラム言語を用いて作成されたプログラム

(プログラム言語命令) をハードウェアにおいて実行可能な命令の列としてのプログラム (ハードウェア実行コード) に変換する際のコンパイラにおける分析を前提としたものである点において、引用発明の「主演算装置」が本願発明の「復号モジュール」と相当関係にある旨を述べたものである旨主張する。

しかし、審決の認定は、上記したとおりであり、引用発明の「コンパイラ」が本願発明の「復号モジュール」と一致すると明確に述べているのであるから、被告の上記主張は採用できない。

(5) もっとも、引用発明の「主演算装置」が、符号化された構成命令のプログラム言語命令をハードウェア実行コードに変換する (復号化する) ことが、甲7文獻に記載されており、引用発明の「プロセッサ」が「復号モジュール」を備えるといえる場合、審決の認定した「・・・命令のデータ経路から構成命令を受け、前記相互接続ネットワーク及び命令セルの少なくともいくつかのデータ経路を構成し、これにより前記データ経路を前記命令セルの回路にマップした前記プログラム命令を実行する復号モジュールと、を備えるプロセッサ。」との一致点の認定に誤りがないと解される余地があることから、この点について検討する。

まず、前記のとおり、引用発明は、

「プロセッサであって、

プログラム可能な相互接続を提供するクロスバーと、

それぞれがクロスバーに接続される処理エレメントであって、処理エレメントは、ALU (算術論理演算)、シフタ、乗算器を含んだ処理エレメントと、

処理エレメント及びクロスバーと共に設けられ、スライスのデータ経路コンポーネントのための構成をロードされたコンフィグレーションメモリと、

クロスバー及び処理エレメントを直接コントロールする主演算装置と、を備えるプロセッサ。」であると認められる。

そして、引用発明の「プロセッサ」は本願発明のA2「プロセッサ」に、引用発明の「プログラム可能な相互接続を提供するクロスバー」は、本願発明のB「相互

接続ネットワーク」に、引用発明の「それぞれがクロスバーに接続される処理エレメントであって、処理エレメントは、ALU（算術論理演算）、シフタ、乗算器を含んだ処理エレメント」は、本願発明のC「それぞれが前記相互接続ネットワークに接続される異種の複数の命令セルであって、該命令セルの各々は少なくとも加算、減算、乗算、除算あるいはシフトの処理を行う該複数の命令セル」にそれぞれ相当する。

ところで、本願発明の「復号モジュール」は、「復号モジュールであって、・・・前記データ経路を前記命令セルの回路にマップした前記プログラム命令を実行する、復号モジュール」である。そして、前記1(2)において述べたとおり、「復号モジュール」は、RICAコンパイラによって生成され、符号化され、構成メモリに格納された構成命令を受けて復号化し、相互接続ネットワーク及び命令セルの少なくともいくつかを構成し、これによりデータ経路を命令セルの回路に現実にマップし、プログラム命令を実行するものと認められる。また、本願明細書においては、構成命令が符号化されたものであることが明確に示されている（【0018】～【0026】、【0032】、【0053】。図6等）。

これに対し、引用発明の「主演算装置」は、「前記相互接続クロスバー及び前記処理エレメントを直接コントロールする」ところ、当該「コントロール」に「前記相互接続ネットワーク及び命令セルの少なくともいくつかのデータ経路を構成し、これにより前記データ経路を前記命令セルの回路にマップした前記プログラム命令を実行する」ことが含まれることは、甲7文献には明示されていない。仮に、上記「コントロール」は、「・・・プログラム命令」を実行することを含むと解したとしても、引用発明の「プロセッサ」が、スライスのデータ経路コンポーネントのための構成をロードされたコンフィグレーションメモリを、処理エレメント及びクロスバーと共に備えることは認められるものの、このコンフィグレーションメモリから「構成」（本願発明の「構成命令」に相当）を受けるのが「主演算装置」であることは、甲7文献に明示されていないし、そのことを示唆する記載もない。

また、「主演算装置」は、コンフィグレーションメモリにロードされた、スライスのデータ経路コンポーネントのための「構成」をコンフィグレーションメモリから受けていると更に仮定したとしても、甲7文献においては、当該「構成」が符号化されているか、既に復号化されたものであるかについては、一切記載がない。甲7文献の「・・・ディレイなく切り替えるため複数のコンフィグレーションを用意する必要があり・・・、そのためコンフィグレーションメモリは、マルチコンテキストである必要がある（すなわち、いくつかのコンフィグレーションを保持しておく必要がある）。1つのコンテキスト（またはコンフィグレーション）を別のコンテキストに切り替えるのに1クロックサイクルで可能であり、・・・コンフィグレーションメモリ内のコンフィグレーションの数は、通常8～32のコンテキストの範囲である。」、 「再構成可能な機能ユニットは、図3に示されるような特殊再構成可能命令により活性化される。この再構成可能命令は、主に2種の情報からなる。1つは、多くの利用可能な構成のどれが使用されるべきかを指定する再構成可能命令識別子（RID）を含んでいるということである。この識別子は、構成メモリにおいて利用可能なコンテキストの数より多数の構成の中から選択することができる。必要な構成が構成メモリに現在ロードされない場合、システムは停止される。それはこのRIDによってインデックスを付けられた小さなキャッシュのように作用する。また、適切な構成は一体になった2次キャッシュからロードされる。」との記載（前記(1)ア(エ)）からすれば、保持された複数のコンフィグレーションを別のコンフィグレーションに切り替えるのに1クロックサイクルで可能であると示されており、このことを考慮すると、引用発明において、主演算装置が「構成」を受けるとして、その段階において、当該「構成」が既にハードウェア実行コードであることも想定される。

そうすると、引用発明において、「構成」を主演算装置において受けるとしても、この「構成」が符号化されており、主演算装置において「復号化」しているか否かは不明であるといわざるを得ない。そして、他に、このような復号機能を有するモ

ジュールの存在は窺われず、当事者も主張していないから、「復号モジュール」を引用発明のプロセッサに含むと認定することはできない。

そうすると、本件において、引用発明と本願発明とは、「・・・復号モジュールと、を備える(プロセッサ)」である点において一致しているとはいえず、引用発明の「構成」は符号化されたものであるか否かは不明であるから、審決は、引用発明が、モジュールが受けた「構成」を復号する機能を有するか、すなわち、「復号モジュール」を含むか否か不明である点を本願発明との相違点として認定すべきである。

(6) 以上によれば、審決のした引用発明の認定、一致点の認定には誤りがあり、これは相違点の看過につながることから、原告の取消事由3には理由がある。

## 第6 結論

以上によれば、原告主張の取消事由3に理由があるから、その余の点について判断するまでもなく、原告の請求を認容することとして、主文のとおり判決する。

知的財産高等裁判所第2部

裁判長裁判官

---

清 水 節

裁判官

---

中 村 恭

裁判官

---

中 武 由 紀