情報システム 27－6
（1990．1．16）

超流通アーキテクチャのためのプロトタイプII

河原正治*　森亮一**

*　筑波大学大学院理工学研究科
**　筑波大学電子・情報工学系

超流通は、ディジタル情報としてのソフトウェアの理想的な流通を目指す概念である。超流通アーキテクチャは、物理的保護機構と暗号による防御を用いることにより、著作権者に対しては権利の保護とソフトウェアの改変の範囲を限定することを可能にし、また、ユーザに対しては無料入手と随時使用の自由を提供する。これらは、既存の計算機に、(1) 実行制御機能、(2) 暗号化・復号機能、(3) 課金制御機能を備えた権利管理機構を付加することにより実現できる。権利管理機構を付加する方法として、コプロセッサに内蔵する方式を提案する。その論理的構造について解説し、超流通の導入期において、多くの利点を持つことを示す。

Prototype II for Superdistribution Architecture

Masaji KAWAHARA*　Ryoichi MORI**
* Master's Degree Program in Sciences and Engineering, University of Tsukuba
** Institute of Information Sciences and Electronics, University of Tsukuba

Superdistribution is a concept toward an ideal distribution of software as digital information. Superdistribution Architecture provides the way to protect the copyright and restrict modifications of software to software vendors, and provides free acquisition and free trial to users. To realize Superdistribution Architecture, SUM (Software Usage Monitor) is added to existing computers. SUM includes (1) monitoring mechanism, (2) encryption mechanism, and (3) accounting mechanism. We propose the way of adding SUM to an existing computer as a coprocessor. We describe the logical structure of the coprocessor and the merits of this way, especially in the introductory stage of Superdistribution.

1. まえがき

超流通（Superdistribution）が実現すれば、ソフトウェアの不正コピーによる様々な問題が解決され、著作権者とユーザ双方の利益を両立させることができる。超流通アーキテクチャ(SdA:Super-distribution Architecture)は、超流通を実現するための物理的手段として森亮一が提唱してきたものである。ＳｄＡは、ディジタル保護容器と暗号による防御機構をその基本構造として持ち、ユーザの手元でソフトウェアの実行に対応した料金の計算を行う機能を有する。流通されるソフトウェアは、暗号化によって充分保護されており、ソフトウェアの複製は自由となる（文献１、２参照）。

我々の研究室では、この超流通を実現するためのシステムの全体設計およびプロトタイプの製作を行なっている。

超流通が実現すれば、従来のソフトウェアの流通は大きく変化する。すなわち、ユーザはソフトウェアの入手に対してではなく、実行に対して料金を支払うようになる。このため、実用化を目指したシステム設計およびプロトタイプの製作においては、ソフトウェア・メーカ、ハードウェア・メーカ、ユーザにとって魅力があり、抵抗が小さい必要がある。このためには、

(1)既存計算機に対する変更が少ないこと、

(2)実行速度の低下が小さいこと、

(3)低価格で導入できること、

(4)ユーザやプログラマに透明であること、

(5)巨大な初期投資を必要とせず移行可能であること、

などが要求される。

また、将来のパーソナル・コンピュータの性能向上を考慮し、

(6)マルチ・プログラミングに対応可能なこと、

を満たす必要があると考える。

本論文では、上記の要件を満足するような超流通アーキテクチャをプロトタイプIIとして提案し、その論理的構造について解説する。

2 超流通アーキテクチャの実現方式

超流通アーキテクチャの実現のためには、既存の計算機に、次のような機能を付加する必要がある。すなわち、ソフトウェアの実行を制御するための実行制御機構、ソフトウェアの暗号化部分の復号と、必要に応じてソフトウェアを暗号化するための暗号化・復号機構、ソフトウェアの実行に応じた料金を徴収するための課金制御機構である。これらをあわせて権利管理機構と呼ぶことにする。また、権利管理機構を搭載した計算機をＳ計算機と呼び、権利管理機構の制御下で動作するソフトウェアをＳソフトウェアと呼ぶ。

権利管理機構は、上述のような機能を実現するために、Ｓソフトウェアとあらかじめ定められたプロトコルによって通信し、Ｓソフトウェアの実行を制御する。また、実行に応じた料金の徴収と支払いを可能とするため、Ｓソフトウェアの実行記録である支払いファイルを作成しこれを保持する。この支払いファイルを回収することによって、収益の分配が可能となる。権利管理機構内部では、Ｓソフトウェアの実行の対価としてＳクレジットと呼ばれる数値が利用される。

権利管理機構を既存の計算機に付加する方式については、従来の研究から、次

のような方式が考えられている（図2.1参照）。

(1) I/Oインタフェースを経由して接続する方式、

(2) バスに直接接続する方式、

(3) バスとCPUの間に組み込む方式、

(4) CPUの内部に組み込む方式

(1)と(2)は、既存計算機の内部に全く変更を加える必要がない。我々の研究室においてすでに試作されているプロトタイプI（文献3参照）は、(1)の方式により実現されている。この方式は、オーバヘッドの大きさから、マルチ・プログラミングには適当ではない。(3)と(4)は、既存計算機の内部に変更を加える必要がある。

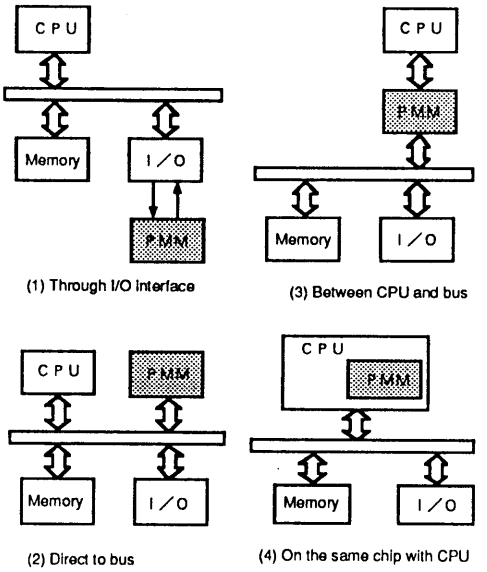

図2.1 権利管理機構の組み込み方式
Fig.2.1 Implementation of SUM (Software Usage Monitor)

3. プロトタイプIIの概要

2.で述べたような理由から、プロトタイプIIは、(2)のバスに直接接続する方式を採用する。

バスに直接接続する方式として、コプロセッサ・インタフェースを用いる方式を考案した。既存MPUのコプロセッサ・インタフェースを利用することによって、権利管理機構に対する処理は、既存MPUの拡張命令として扱われる。このため、プログラマにとって権利管理機構の付加は、ほぼ透明になる。また、既存の計算機のコプロセッサ・ソケットに権利管理機構を内蔵したコプロセッサを装着するだけで超流通に対応できるため、ユーザにとっても、ほぼ透明になる。

今回、提案するプロトタイプIIは、主として研究室の実験装置の都合により、MC68020を用い、そのコプロセッサ・インタフェースを利用する。すなわち、既存計算機のコプロセッサ・ソケットに権利管理機構内蔵のコプロセッサを装着する形で超流通アーキテクチャを実現する方式を設計した。設計においては、他のMPUへの移植が容易であるよう考慮した。

一方、Sソフトウェアの実行に伴う料金徴収を円滑に進めるためには、ユーザ・インタフェースの設計も重要である。

ユーザ・インタフェースは、「SdA ユーザ・ユーティリティ・プログラム（以下、ユーザ・ユーティリティ・プログラム）」と呼ばれるSソフトウェアによって実現することができる。プロトタイプIIでは、ユーザ・インタフェースも権利管理機構の機能として実装されている。これに対して、ユーザ・インタフェースを権利管理機構の外部で実現することにより、権利管理機構の複雑さが軽減され、高信頼化が促進される。ユーザ・ユーティリティ・プログラムの機能は、

(1) Sソフトウェアの課金情報の表示、

(2)支払いファイルの転送

である。

4. プロトタイプIIの論理的構造

プロトタイプI（文献3参照）は、権利管理ＯＳいう概念を用い、その制御下で計量プログラムが動作するという方式をとっている。この計量プログラムの柔軟さによって、多様な料金体系を実現している。プロトタイプIIでは、権利管理機構の複雑さを軽減する方向の試みとして、権利管理ＯＳという概念は用いず、時間料金と回数料金という単純な料金体系を実現する。

4.1 プロトタイプIIのアーキテクチャ

図4.1は、プロトタイプIIの論理的構造の概念図である。

権利管理機構は、ディジタル保護容器によって物理的に保護されたコプロセッサに内蔵された形となる。Ｓソフトウェア本体は、MC68020上で実行され、実行制御のためにコプロセッサと通信する。

インタフェース部は、MC68020のコプロセッサ・インタフェースの規格（文献4、5参照）を満足するものであり、MC68020との間であらかじめ定められたプロトコルによって通信を行う。

コマンド解析処理部は、コプロセッサ・インタフェース部がMC68020から受けとったコマンドを解析し、所定の処理を行う。

実行制御部は、Ｓソフトウェアの実行を制御する。

課金情報バッファは、複数のエントリからなり、ソフトウェア起動時に一つのエントリが使用されるようになっている。エントリには、対応するソフトウェアのソフトウェア番号、使用料金に対応するＳクレジットの値などが含まれる（4.2参照）。

支払いファイルは、支払い情報を保持する。

課金制御部は、課金情報バッファと支払いファイルを参照しながら、Ｓクレジットの減算を行う。また、違反行為のチェックと記録、違反行為の回数に応じて実行制御部の機能の制限も行う（例えば、違反行為が規定の回数を越えると、Ｓソフトウェアの実行を一定期間不能にする）。ユーザ・ユーティリティ・プログラムの機能を実現するための制御も行う。

暗号化・復号部は、課金制御情報部（4.2参照）やソフトウェア中の暗号化部を復号する機能を有する。逆の方向、すなわち、プログラマが作った平文のプログラムを、他者（例えば、ハードウェア・メーカ）に依存することなく、自己のパーソナル・コンピュータを用いて、暗号化できるような機能を提供することにより、超流通への新規参入を容易にすることが可能である。

4.2 プロトタイプIIにおけるＳソフトウェアの構造

プロトタイプIIにおけるＳソフトウェアの構造は図4.2の通りである。

課金制御情報部は、Ｓソフトウェア起動時にコプロセッサに転送され、コプロセッサでの課金制御に用いられるもので、次のような情報を含んでいる。

(1)ソフトウェア番号

Ｓソフトウェアに対してユニークに割り当てられた番号である。

(2)アクセス制御鍵

Ｓソフトウェア本体中のチェック・

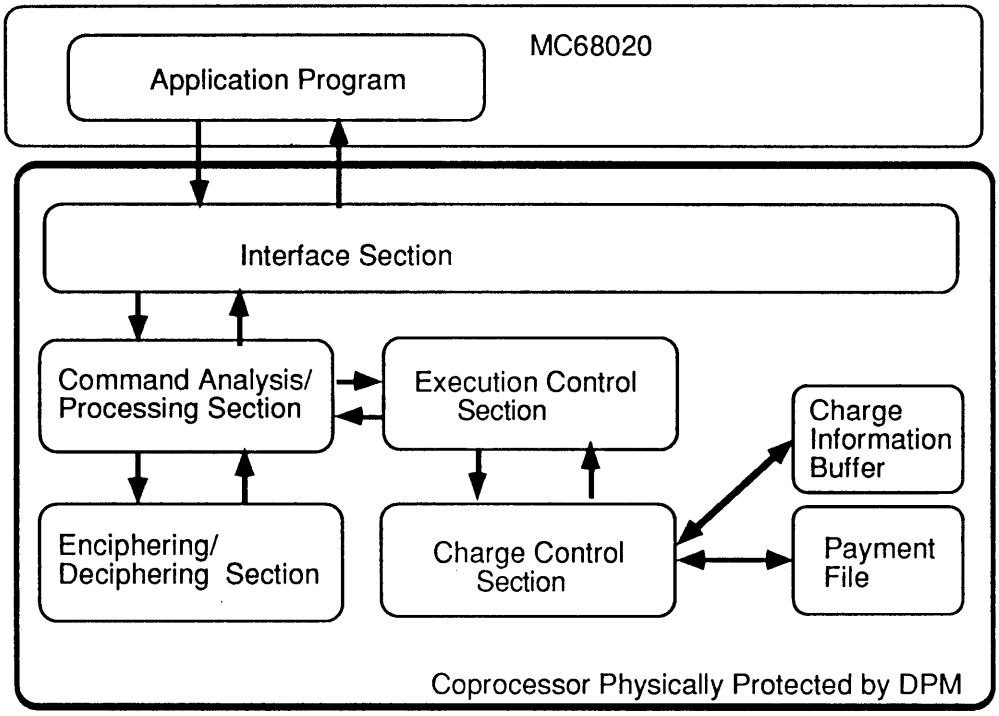

図4.1 プロトタイプIIの論理的構造
Fig.4.1 Conceptual Diagram of Architecture of Prototype II

ルーチンが、コプロセッサによる認証を得るために使用する。

(3)暗号化鍵

ソフトウェア中のMC68020の命令は、この暗号化鍵を用いて暗号化され、ソフトウェア本体中に埋め込まれている。ソフトウェア実行時に、暗号化されたブロックがコプロセッサに転送され、この鍵を用いて復号しながら実行される。

(4)単位料金時間

単位Sクレジット当りのソフトウェアの使用可能時間である。課金制御部が、この時間が経過するごとにSクレジットの減算を行う。

(5)回数使用料金

回数料金を採用しているSソフトウェアの1回の実行に対して、この料金分だけSクレジットが減算される（例えば、ファイルの保存1回につき2クレジット、印刷1回につき5クレジットなど）。

(6)コプロセッサ内に確保する自由メモリの大きさ

実行制御のために、Sソフトウェア本体が用いることのできるコプロセッサ内のメモリの大きさが指定される。

この課金情報部の暗号化には、公開鍵暗号系を用いる方式と、慣用暗号系を用いる方式がある。

初期化ルーチンは、上記の課金情報を

コプロセッサ命令（文献4、5参照）を用いて、コプロセッサ側に転送するものである。

また、ソフトウェア本体中には、後述するようなチェック・ルーチンが埋め込まれており、このチェック・ルーチンのコプロセッサに対する通信が、あらかじめ規定された内容である場合のみ、ソフトウェアの実行を継続する構造になる。また、ソフトウェア本体の少なくとも一部は高速な慣用暗号系（文献6参照）を用いて暗号化されており、暗号化ブロックはコプロセッサに転送され、復号・実行される。

4.3 ソフトウェアの実行制御方式

Sソフトウェアとコプロセッサが通信を行なうことによりソフトウェアの実行を制御することになる。通信が正しく成立しない場合は、Sソフトウェアの実行を停止する必要がある。また、Sソフトウェアの構造の解析が困難であるような構成が必要である。以下のような手法が考えられる。

(1) Sソフトウェアからの問い合せがあらかじめ定められたものと一致しているかどうかをコプロセッサが確認する方式。

(2) Sソフトウェアから受取ったデータに基づいて、コプロセッサがジャンプ・アドレスを返す方式。

(3) あらかじめ定められた時間内に通信が成立するかどうかを確認する方式。

(4) ソフトウェア本体中の一部に、MC68020命令を暗号化して埋め込む方式。

暗号化ブロックは、コプロセッサに転送され、復号・実行され、結果がMC68020に返される。

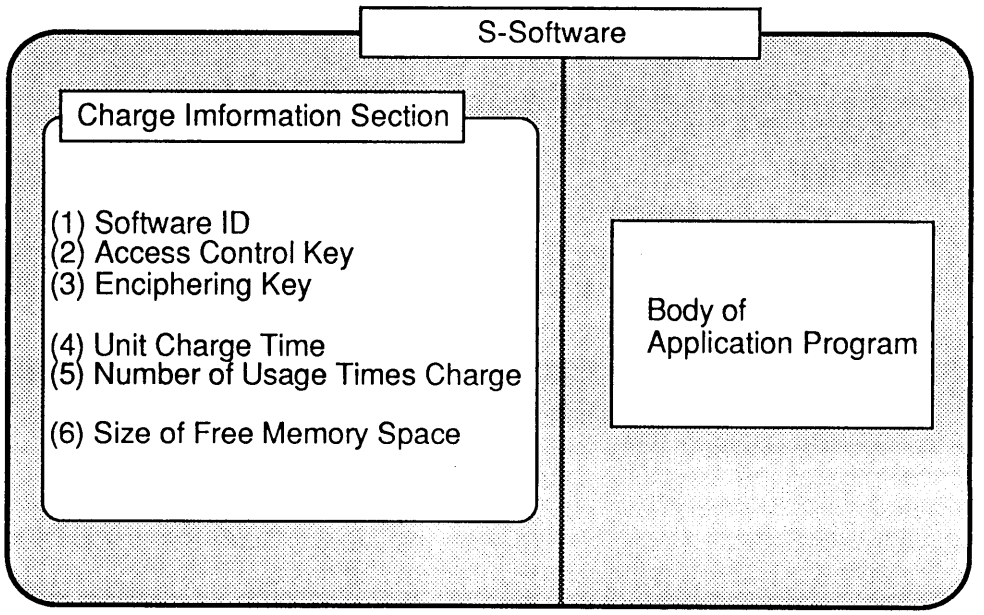

図4.2 Sソフトウェアの構造
Fig.4.2 Structure of S-Software

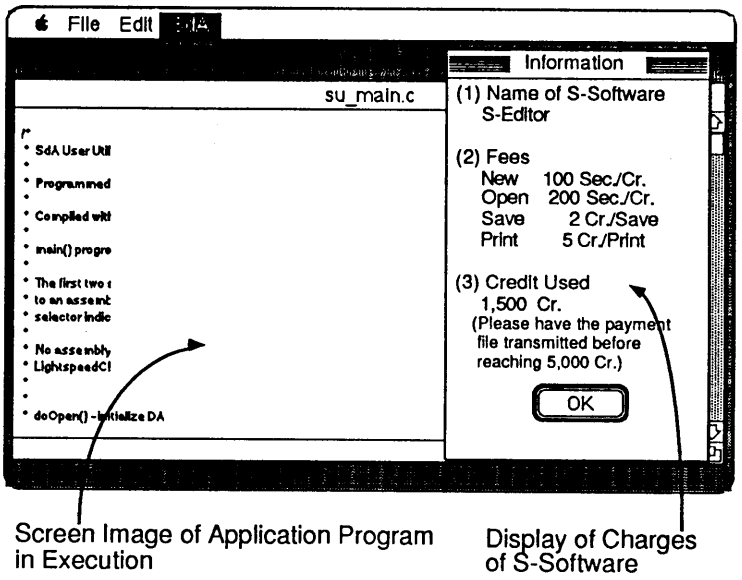

図4.3 SdAユーザ・ユーティリティ・プログラムの実行例
Fig.4.2 Example of Execution of SdA User Utility Program

上記の方式を適当に組合せることにより、Sソフトウェアの構造の解析をより困難にすることができる。

4.4ユーザ・インタフェース

図4.3にユーザ・ユーティリティ・プログラムの例を示した。ユーザは、このプログラムを通じて、Sソフトウェアの課金情報、Sクレジットの残額などを確認することができる。マルチ・プログラミングの提供されていない場合においても、ユーザの便宜を図るため、ユーザ・ユーティリティ・プログラムは、アプリケーション・プログラムが動作中の場合も実行可能であるように配慮するべきであり、近く実現可能であると考えている。

5. むすび

権利管理機構をコプロセッサに内蔵し、既存計算機のコプロセッサ・ソケットに装着することにより、超流通アーキテクチャを実現する方式について解説した。現在、浮動小数点演算用のコプロセッサのためのソケットを備えているパーソナル・コンピュータが一般的になっており、これを利用することができる。超流通アーキテクチャの導入期においては、浮動小数点演算機能と権利管理機構とを同一のチップ上に実装することより、ユーザの便宜を図ることができる。コプロセッサ・ソケットを備えていないパーソナル・コンピュータについては、拡張スロットに挿入する方式が採用できる。この場合、プログラマがコプロセッサ・インタフェースをシミュレートするソフトウェアを用意しなければならないが、大きなオーバヘッドにはならない。1チップのコプロセッサを提供することによって超流通を実現することができ、また、巨大な初期投資を必要としない運用形態も検討されている。従来の同種のシステ

ム（電気、水道、電話など）の構築には不可欠であった巨大な初期投資を必要とせず、かつ、その将来の規模は、それらのシステムと同程度まで成長する可能性を持っていることは、興味深いことである。

今回提案したプロトタイプIIでは、プロトタイプIのような多様な料金体系は提供されていない。これに対しては、支払いファイルを回収先で処理することにより、多様な料金体系を実現する方式などを設計中であり、別の論文で述べる。

マルチプログラミングへの対応は容易である。MC68020のコプロセッサ・インタフェースは、コプロセッサ状態のセーブ・リストア命令を備えており、プロセスのスイッチ時における、コプロセッサ側の処理の中断・再開に対処することができる。また、コプロセッサの処理が短時間で終了する場合は、コプロセッサ命令処理中の割り込みをペンディングにすることも可能である。チェック・ルーチンによる認証は、通常、汎用コプロセッサ命令を一つ挿入するだけであるので、この手法を用いることができる。

これまでの記述は、アプリケーション・プログラムを対象としているが、基本的に機械語命令を用いるため、システム・プログラムに対する応用も可能である。将来のパーソナル・コンピュータの性能向上にともない、仮想計算機システムによる、複数オペレーティング・システムの動作にも充分対応可能である。

今回は、MC68020の特長を利用した形でプロトタイプIIを提案したが、図4.1のインタフェース部のみがMC68020のコプロセッサ・インタフェースに依存した部分である。このため、インタフェース部のみを変更することで他のMPUへの適応が可能である。

今後は、権利管理機構の実行制御機能と課金制御機能をシミュレーションした後、ASICによるコプロセッサの試作と、その運用形態の詳細についての検討とを並行して進める予定である。このコプロセッサを限られたメンバに配布して、種々の機能の検証を行なうことも考えている。

文献

1）森亮一、河原正治、篠崎雅英、大瀧保広：「超流通」研究計画の現在、第3回「利用者指向の情報システムシンポジウム」論文集, pp.173-180

2）森亮一、田代秀一：ソフトウェア・サービス・システム（ＳＳＳ）の提案、電子情報通信学会論文誌 Vol.J70-D No.1, pp.70-81, 1987

3）田代秀一、森亮一：ソフトウェア・サービス・システム（ＳＳＳ）の小規模な試作、 同上Vol.J70-D No.2, pp.335-345, 1987

4）MC68020ユーザーズ・マニュアル, ＣＱ出版社,Motorola Inc.(1986)

5）MC68881ユーザーズ・マニュアル, 電波新聞社,Motorola Inc.(1988)

6）栗原定見、宮口庄司：ＦＥＡＬ暗号方式の開発と適用、1989年情報理論とその応用、暗号と情報セキュリティジョイントワークショップ講演論文集、pp.159-166

7）日本電子工業振興協会：第3部 ソフトウェア流通の為の基盤技術、

マイクロコンピュータに関する
調査報告書［II］、89-パ-5, pp.10
7-221, 1989年3月
8）超流通アーキテクチャSdA報告
書、筑波大学森亮一研究室、19
88,1989