C-001

設計仕様解析によるハード／ソフト最適分割システムの構築と評価

Best Partitioning Methods of Hardware and Software with Design Specification Analysis

梅原　直人† 和田　智行† 山崎　勝弘†
Naoto Umehara Tomoyuki Wada Katsuhiro Yamazaki

1. はじめに

本研究ではハードウェアとソフトウェアの最適な分割を実現するために，対象となるアプリケーションの C 言語のソフトウェアプロトタイプが完成した段階で，それを解析して，設計の早期段階からハードウェアとソフトウェアの最適な分割を見出すことを目的とする[1].

本システムは C ソースコードからモジュール毎のコントロールデータフローグラフ（CDFG）などのパラメータを作成して，そこに対象のシステム固有の特徴量を考慮することで各モジュールの相対性能を導き出す．性能に速度重視や回路規模重視，または複数項目に対しユーザ要求の重みをつけることで，要求に沿った最適なシステム構成案を提示する．

本システムを MISTY1 暗号と AES 暗号に適用して，本システムの有効性を評価する．検証環境には Xilinx 社の Virtex 4 を搭載した FPGA ボードにソフトコア CPU「MicroBlaze」を実装して行った．システムの検証にはクロックサイクル数などの実測値とツールから合成される予測値を用いて行う．これにより，現実と最適分割システムが出す答えがどの程度乖離しているかを数値的に，または経験則からも検証する．

2. ハード／ソフト最適分割システム

2.1 システムの構成と機能

ハードウェアとソフトウェアを最適に分割する手法には経験則からの試行や，UML といった上位モデルから見つけ出そうとする方法があげられる．しかし，手法の抽象度の高さから決定的な解を発見するには至っていない．そこで本研究では，C 言語ソフトウェアプロトタイプといった具体的な内容を利用して，早期にかつ的確に分割を指定し，設計者を支援することのできるシステムを考案する．本システムの入力は C ソースコード，システムの特徴，ユーザの要求であり，これらを総称して「設計仕様」と本研究では呼ぶ．これら設計仕様を解析することで，最終的にはハードウェア／ソフトウェアの分割案を提案する．

システムの構成と流れを図 1に示す．まず，使用されるメモリ量や回路規模などを得るために，Cソースコードで使用されるデータ量や演算強度などを抽出して，ソースコードを中間表現に変換する．ここで，演算強度とは，演算の負荷の高さを表す．排他的論理和を 1 回実行する速度を基準とし，これの何倍になるかをモジュールの演算強度として導いている．次に，メモリ量や演算の速度などはシステムの特徴にも大きな影響を受けるので，命令セットやアーキテクチャなどから特徴量を導く．特徴量とは，システムの特徴を表す数値的な要素のことである．具体的には演算の種類と数，パイプライン段数，プロセッサのビット幅などがあげられる．これを中間表現に組み込むことで性能解析を行う．

ソースコードの中間表現とシステムの特徴量を組み合わ

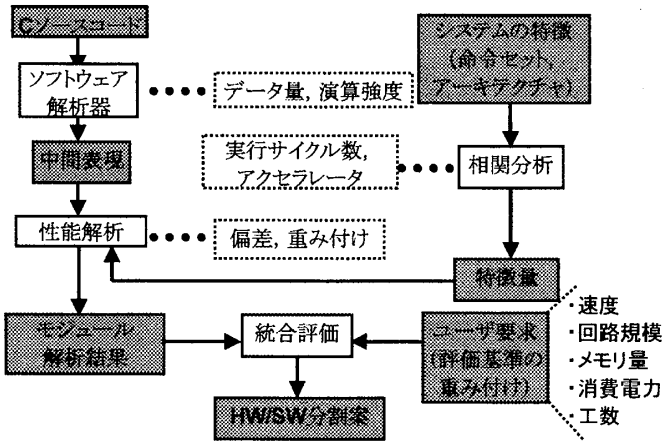

図 1 ハード／ソフト最適分割システム

せることで，モジュール毎の演算速度や回路規模を見積もり，モジュール間の性能と仕様を導出する．

その際に見積もる要素は，速度・回路規模・使用メモリ量・消費電力・工数である．最後にそれらを速度重視・消費電力重視などのユーザの要求を重み付けし，分割パターン毎に評価することで最終的なハード/ソフト分割案を提示する．

2.2 設計仕様解析

設計仕様の解析方法を図 2に示す．まず，Cソースコードを前処理して，各々の処理モジュールとシステム全体に関わる部分を分けておく．次に，モジュール毎にCDFGの生成や内部演算式の評価などを行って，各モジュールの中間表現を生成する．

中間表現はモジュール毎の内部状態を表すだけであり，そこから回路規模や処理速度と言った性能を正確に見積もる性能解析が必要となる．性能解析を行うためには中間表現によるモジュール毎の内部状態を使用するだけではなく，実装環境などのモジュール外部要素を把握しなければならない．例えば，CPU にバレルシフタがあるかどうかでシフト演算の速度は数十倍の差が生まれるし，乗算器など特殊演算装置について考えた場合も同様である．また CPU アーキテクチャによっては命令セットを考慮しなければならない．ARM プロセッサなどは条件付き演算という特殊な分岐命令を保持しているので，CDFG で分岐がある場合にその必要サイクル数が変わる．また，以後の処理を行いやすくするために，前処理部分で抽出されるモジュール名などを保持するパラメータも外部要素として規定している．

このようなソースコードのみでは分からない外部の環境を考慮することで，その場合にあった性能の解析・予測を行う．性能解析は，環境パラメータによる重み付けを中間表現に行うことで現実に適した性能数値を導出する．性能を予測する際には，ある中間表現が単一の性能値に結びつくわけではなく，相互に絡ませることでより

† 立命館大学大学院 理工学研究科, Graduate school of Science and Engineering, Ritsumeikan University

精度を高くする．例えば，本システムにおいて処理速度を見積もるには CDFG と演算式評価を組み合わる．プログラムの演算量を決める重要な要素に負荷が高いループであるタイトループの存在があるので，それを考えるだけならば CDFG を見るだけで処理速度をある程度決めることもできる．しかし，演算の内容に左右されるのも無論のことであるので，本システムではループや分岐の構造とそこにある演算の内容から処理速度性能を導き出す．

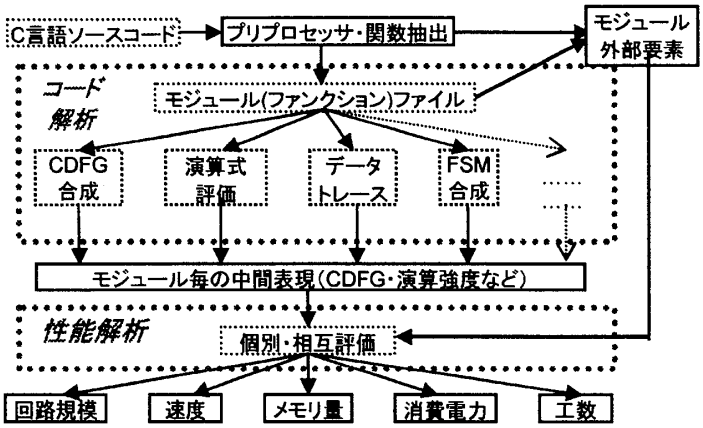

図 2 設計仕様の解析手法

2.3 分割パターン結果の評価方法

現時点での最適分割システムは CDFG 生成，演算式評価，及び各モジュールのメモリ量算出の 3 項目を導き出すことが出来る．これにより，処理速度・回路規模・メモリ量の評価はできるので，本論文では，これらの点から本システムの有効性を評価する．

実験結果はそれぞれの項目で数値を計算されているので，各々の傾向は大体見て取れるが，相互の影響を考慮して評価することは困難である．そこで，以下に示す独自に考えた評価式を導入することで定量的に評価する．

$$Eval = \sum \left[Wu_{item} \times \frac{Std(Val_{item}) - StdMin}{StdMax - StdMin} \right]_{pattern} \quad \cdots(1)$$

式(1)の Eval はそのパターンにおける総合評価結果であり，システムとして該当パターンのプライオリティの高さを表す．数値が大きいほど優秀である．Val はある項目（処理速度など）に関する解析値，または実測値である．Wu はユーザ要求で決定した重み付けであり，処理速度や回路規模などに応じた比率が記されている．Std()関数は偏差値を算出する関数であり，あるパターンでの処理速度などの偏差値を導く．偏差値は一般的に使用される”（ある値－平均値）×10/標準偏差+50”という方式を採用している．StdMax と StdMin は各々ある性能項目における偏差値の最大値と最小値を表す．これらを使用しているのは偏差値を独自に標準化し，0.0～1.0 の範囲に値を収めるためである．単純に偏差値だけを使ってしまうと項目によって値のばらつき方に著しい差が出てしまうので，このような措置を取っている．この評価式を使用することで，処理速度や回路規模と言った異なる基準の項目を同列にして定量的に，ひいては視覚的に優先度（最終評価値）を判断することが可能となる．

2.4 検証システム

本システムの検証環境には，Xilinx 社のソフトコアプロセッサ”MicroBlaze”を中心に使用する．その構成を図 3に示す．MicroBlazeでは，CPUとのハード／ソフト協調動作ができる環境を構築するツールが豊富に揃っており，インタフェースIPとの連携も分かりやすく，利用しやすい．

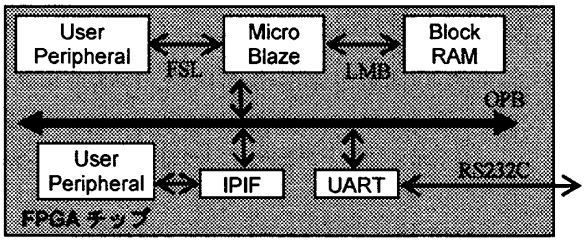

図 3 検証システムの構成

ユーザが作成したシステムのハードウェア処理に当たるモジュールや評価用のクロックカウンタなどは，各モジュールを Verilog-HDL で設計し，FSL(Fast Simplex Link)を通じて接続しており，レイテンシが 2 サイクル以内という高速データ転送を可能とする．今回は OPB(On-chip Peripheral Bus)にシリアル通信用 UART の IP を搭載して，PC と通信することで動作結果を確認した．

3. MISTY1 暗号での実験と考察

3.1 MISTY1 暗号の概要

MISTY1 は 128bit秘密鍵／64bitブロック暗号である．データを分割して排他的論理和やテーブル参照を行うことを変換処理の基本としている．条件分岐などは一切存在しない，非常にハードウェア化を意識したアルゴリズムとなっている．ループ回数は 4 の倍数であれば良いが，推奨ループ回数はN=8 とされている．図 4にMISTY1 暗号フローを示す．MISTY1 暗号は暗号化する関数が入れ子になっており，図 4 (a)のフロー中のFO関数(図 4(b))は更にFI関数(図 4 (c))を包含している．

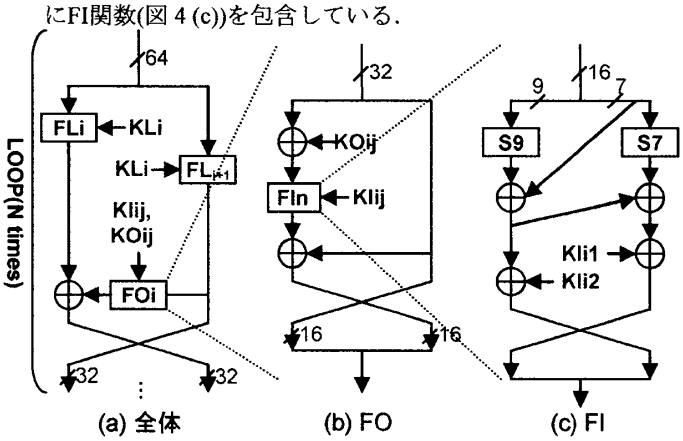

図 4 MISTY1 暗号基本構成

3.2 MISTY1 暗号へのシステム適用

MISTY1 暗号は論理演算とテーブル参照のみからなるシンプルな構成となっているが，ループ制御による入れ子構造が基本となっているため，モジュール分割が非常に行いづらい．そこで，MISTY1 暗号のループ 1 回分を基本モジュールとし，それを複数つなげることでモジュール数を増やす．単純に 100 回ループするならば，対応するモジュールを 100 個と見なす．その中でモジュールをインライン展開して任意の大きさのモジュールを作り出すことで，ソフトウェアとハードウェアの処理のバラ

ンスを自由に変えて，様々な分割パターンの実験を行う．その解析結果を計測・考察することで，最適分割システムの設計仕様解析ツールにフィードバックする．

3.3 実験結果

表 1にMISTY1 暗号のCソースコードの解析結果を示す．ここで，表 1の処理速度とはクロックサイクル数のことである．表 1はそれぞれの単位で数値を計算されているので，各々の傾向は大体見て取れるが相互の影響を考慮して評価することは困難である．そこで，表 1の結果と式(1)を用いて，分割パターンのプライオリティを計算した．表 2にはMISTY1 暗号の分割パターンA～Fの重み付け一覧を示す．これはHWループ回数のそれぞれに対してA～Fの 6 種類の性能項目の重視方法を表したものである．例えばパターン"A"なら，処理速度に 8 割，残り 2 つに 1 割ずつの重みを付けて評価している．

MISTY1 暗号ループ 100 回中のハードウェアの担当回数を変化させることで実験パターンを増やしており，そのハードウェア回路のループ回数は[0,8,32,64,100]となっている．重み付けは，解析結果と実機での評価を対比するために，2 グループ 3 パターンずつ，総計 6 パターンを用意する．パターンA,B,Cが処理速度・回路規模・使用メモリ量のうち 1 項目に対し 8 割の重み付けをした場合，パターンD,E,Fは 1 項目に対し 6 割の重みづけをした場合である．さらにこの解析結果の特徴的な部分として抽出し，グラフ化した物を図 5と図 6に示す．

表 1 MISTY1 暗号システム解析結果

HW ループ回数	0	8	32	64	100
処理速度	92330	85120	63056	24406	542
回路規模	0	13265	13270	13276	13282
メモリ量	38048	37668	25885	13728	64

※HW ループ回数"0"：全ソフト実行，"100"：全ハード実行

表 2 MISTY1 暗号分割パターンの重み付け

パターン名	A	B	C	D	E	F
処理速度	8	1	1	6	2	2
回路規模	1	8	1	2	6	2
メモリ使用量	1	1	8	2	2	6

3.4 評価と考察

まず図 5と図 6において，両者を比較して考察する．処理速度重視評価（AとD）とメモリ使用量重視（CとF）はほぼ同じ折れ線を辿っており，これらの間には非常に強い相関関係があると判断できる．逆に，回路規模重視（BとE）は前述の 2 組とは評価の傾向が反対となっている．即ち強い逆相関となっている．また，図 5に比べて図 6の方が少々，処理速度重視評価とメモリ使用量重視の組と，回路規模重視との間にある逆相関関係の差が縮まっている．これは評価におけるユーザの重視比率を変えている影響であり，評価に反映されているので，意図通りに結果が出た証左でもある．

次に，回路規模重視のパターンBとEの評価値を他と比べると，ループ 8 回以降があまり性能向上していない．これはHWのループ回数を増やしても，処理速度とメモリ量に比べると，回路規模性能では利点が無いことを示唆している．即ち，CPU単体でソフトウェア動作させた方が有利ということが言える．

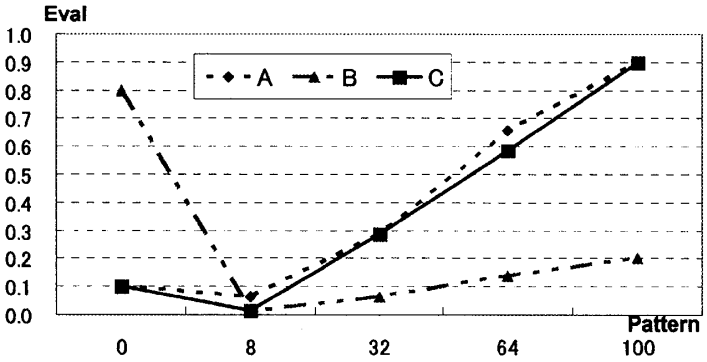

図 5 MISTY1 暗号解析結果評価グラフ(A,B,C)

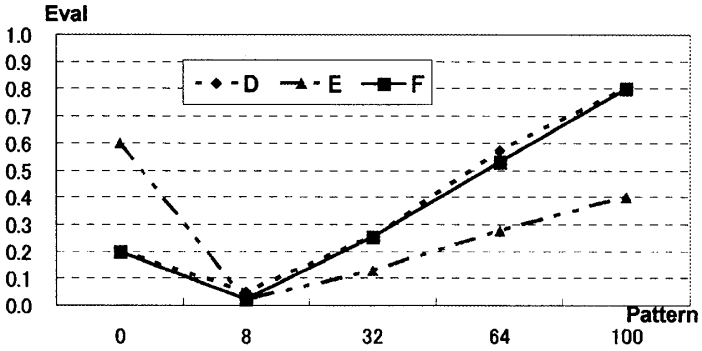

図 6 MISTY1 暗号解析結果評価グラフ(D,E,F)

最後に処理速度とメモリ量重視だが，これらは非常に似通った結果になっており，また評価値についてもループ 8 回を除けば全てのパターンにおいてハードウェア化した方が勝っている． このことからもMISTY1 暗号は処理速度とメモリ量の削減ではハードウェア化の利点が非常に大きく，回路規模の面ではハードウェア化は不利になると判断できる．

4. AES 暗号での実験と考察

4.1 AES 暗号の概要

MISTY1 暗号はループ回数を調整することで任意のサイズのモジュールを作ることが出来るが，性能が線形的になり，結果もそれに応じた形になりやすい．そこで，複雑な演算を行うアプリケーションとして，すでに AES 暗号を設計・実装して計測を行っている[3]ので，これを利用して追加実験を行った．

AES 暗号は入力データに一定回数のデータ変換を行うことで暗号化する．今回は 128 ビットの暗号強度のみに限定している．データ変換には AddRoundKey，SubBytes，ShiftRows，及び MixColumns の 4 種類がある．AddRoundKey では入力データと鍵データの排他的論理和を取る．SubBytes は，テーブル参照で実現している．ShiftRows は入力データを行列と見て，各行でシフトを行う．ただしシフト量は行によって異なる．MixColumns は，数学的なガロアフィールド理論に基づいた行列演算をしなければならない．

従来研究では，協調設計に最も重要と見られる処理速度に関係するソフトウェアのCPU処理負荷を見て，その比率から処理負荷の大きいモジュールを優先的にハードウェア化する方針を立てて，分割パターンを数通りに絞った[4]．そのパターンを表 3に示す．

表 3 AES 暗号分割パターン

	ハードウェア処理部	ソフトウェア処理部
ア		SubBytes, MixColumns, ShiftRows, AddRoundKey
イ	MixColumns	SubBytes, ShiftRows, AddRoundKey
ウ	SubBytes	MixColumns, AddRoundKey, ShiftRows
エ	SubBytes, MixColumns	ShiftRows, AddRoundKey
オ	SubBytes, MixColumns, ShiftRows	AddRoundKey
カ	SubBytes, MixColumns, AddRoundKey	ShiftRows
キ	SubBytes, MixColumns, ShiftRows, AddRoundKey	

4.2 実験結果と考察

実験に使用する重みは表 2と同一である．表 3の各パターンに対してMISTY1 暗号と同様にユーザ要求の重み付けを変更して実験を行い，その推移を解析評価結果と実測評価結果とを比べることでどれだけ解析結果が実際の数値に迫ることができているかを調べた．

MISTY1 暗号での結果と同様に評価式を用いて，実際の測定値と本研究のシステムでの予測値を比較する．図 7に実機から得た性能評価をグラフ化したもの，図 8に本システムで予測した性能評価をグラフ化したものを示す．

両者を比較すると，A～F のそれぞれの動きを比べると似たような形になっており，特に”ウ”の時にプライオリティが急激に落ちるところや，”キ”の方向に遷移するときに”B”と”E”がやや低迷している点などが類似している．このことは考案した本研究のシステムが実際にある程度有効だということの視覚的な証明である．しかし，実測値からの評価の方が比較的，起伏の激しい折れ線グラフになっているのに対し，予測値からの評価結果は少々滑らかになっている．これは回路規模の見積もりが不十分だったため，及び評価方法が完全ではないためだと考えられる．

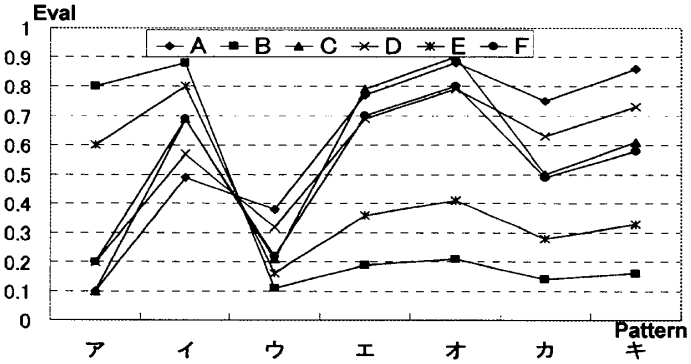

図 7 実機による評価結果

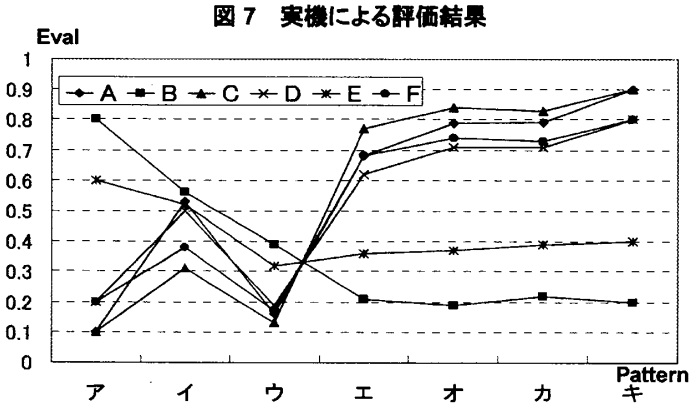

図 8 本システムの解析による評価結果

このままでは両結果が似ているとしか言えない．そこで，2 つのグラフがどれだけ類似しているかを定量的に証明する．両方の数値は共に，間隔尺度以上の単位を有しているので，ピアソンの積率相関係数によって関係を導き出せる．その結果を図 9に示す．積率相関係数は 1.0 に近ければ正の相関関係，0 に近ければ無関係を表す．最適分割システムの結果に適用すると，おおよそ 1.0 に近い数値を示していることが見て取れる．平均すると約 0.85 となっている．総じて，解析結果は現実に即していると言える．しかし，”8：1：1”と”6：2：2”の両方の場合において，処理速度を重視した結果の時に一番相関係数が 1.0 に近く，メモリ使用量を重視した場合，一番相関係数が低い結果になっている．これは処理速度に関する解析は非常に現実に近く，精度が高いと言え，メモリ量の見積もりにおいては精度が一番低いと言える．

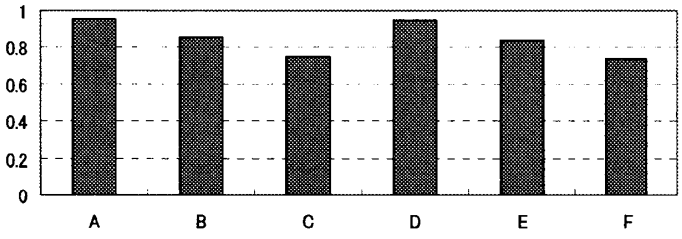

図 9 実測・解析の積率相関係数

5. おわりに

本論文では協調設計におけるハードウェア／ソフトウェアの最適分割を行うシステムを提案し，その構成や処理の内容などについて述べた．また暗号アプリケーションである MISTY1 を対象として，ソフトコア CPU を用いて実装・計測を行い，処理速度に着目してその結果を評価して，本システムのテストを行った．また，以前に実機計測などを行った AES 暗号のデータを利用することで更に詳細な評価を行い，実測値と比較して本システムの解析結果が現実的に有効であることを示した．今後の課題として，本システムの妥当性の検証やツールの完成，及びシステムの評価精度向上などがあげられる．

参考文献

[1] 梅原直人,山崎勝弘:設計仕様の解析によるハードウェア/ソフトウェア最適分割手法の検討,情報処理学会関西支部大会 VLSI システム研究会,C-08,2006.

[2] 梅原直人,和田智行,山崎勝弘:設計仕様の解析によるハード/ソフト最適分割手法の実現と評価,第 69 回情報処理学会全国大会 システム LSI 設計技術(2),1L-3,2007.

[3] 梅原直人,古川達久,的場督永,山崎勝弘,小柳滋:ハード/ソフト最適分割を考慮した AES 暗号システムと JPEG エンコーダの設計と検証,FIT2005,C-034,2005.

[4] 梅原直人,古川達久,的場督永,山崎勝弘,小柳滋:ソフト・マクロ CPU を用いた回路設計とハードウェア/ソフトウェア最適分割法の検討,情報処理学会関西支部大会 VLSI システム研究会,C-07,2005.