

C-001

## TCP/IP ハードコアの設計とその消費電力解析 Designing a TCP/IP Core for Power Consumption Analysis

棚町 健一†1  
Kenichi Tanamachi

井上 弘士†2  
Koji Inoue

モシニャガ・ワシリー†2  
Vasily G. Moshnyaga

### 1. はじめに

近年、コンピュータを取り巻く様々な環境の中でインターネットは非常に重要な位置を占めるようになった。ネットワーク環境は急速に普及し、そのバンド幅は近い将来1Gbpsにも達することが予想される。それに伴い、パケット処理の高速化に対する要求も高まってきた[2]。この課題を解決する1つの手段として、TCP/IP処理のハードウェア化がある(以降、TCP/IPハードコアと呼ぶ)。通常はソフトウェアで実行されるTCP/IPを専用ハードウェア化することで大幅な性能向上を期待できるためである。一方、ノートPCに代表される携帯電子機器システムの普及に伴い、コンピュータ・システムの低消費電力化は重要な設計制約の1つとして位置づけられている。一般に、CMOS回路の消費電力は動作周波数に比例するため、今後より高速動作が要求されるTCP/IPハードコアにおいてもその低消費電力化が重要になる。

そこで本稿では、TCP/IPハードコアの設計を行い、その回路面積ならびに消費電力の解析を行う。具体的には、TCP/IPコアの機能分割、ハードウェア記述言語による設計ならびに論理合成を行い、消費電力を測定する。また、自動配置配線ツールを用いたレイアウトに基づき、配線負荷容量を調査する。

### 2. TCP/IP ハードコアの設計

TCP/IPプロトコルは図1に示すようにスタック構造となっている。各層は互いに独立しており、ある層で構成に変更が生じても他の層に影響は及ばない。TCPはトランスポート層に属しており、信頼性のあるコネクションを確立することがその役割となる。一方、IPはネットワーク層に属しており、その役割は正しい相手にデータの受け渡しや受け取りを保障することである。ここでパケット処理をハードウェア化するためには、TCPとIPの機能分割を行う必要がある。そこで、本設計ではTCPを以下のように4つの機能ブロックに分割した。

- **port\_ctr**: 本モジュールは、上位層であるアプリケーション層との間でデータの受け渡しを行う。アプリケーション・ポート番号を指定することで特定のアプリケーション間でのデータ通信を実現する。
- **data\_ctr**: 本モジュールは、シーケンス番号や応答確認番号を用いて、データの分割(送信側)や統合(受信

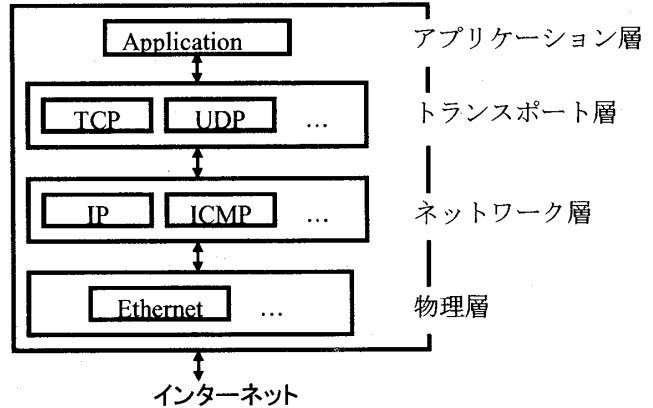


図1: TCP/IPプロトコルスタック

側)を行う。また、タイマーを用いてデータ再送が必要であるか否かを判定する。

- **window\_ctr**: 本モジュールでは受信バッファの管理を行う。バッファの空き具合を通信相手に知らせることで受信するデータ量を調整する。
- **checksum**: 本モジュールは、送信データと受信データが異なっていないかどうかを検出する。もし、データに誤りがある場合は送信側に再送要求する。

一方、IPは次の4つの部分に機能分割できる。

- **service\_info**: 本モジュールでは、IPヘッダ情報(IPバージョン、ヘッダ長、パケット長、サービスタイプ)を決定する。サービスタイプには通信の品質を設定する。
- **frag\_ctr**: 本モジュールではTCPからのデータを更に細かく分割(フラグメント)するかどうかを決定し、フラグメントする場合はそのデータの識別子を割り当てる。
- **checksum**: 本モジュールはTCPのチェックサムの場合とは異なり、データに関してではなくIPヘッダに関してのみのエラーの検出を行う。
- **add\_ctr**: 本モジュールでは送信先のIPアドレスや送信元のIPアドレスを指定することで通信相手同士でのデータ通信を可能にしている。

†1 福岡大学大学院工学研究科電子情報工学専攻

†2 福岡大学工学部電子情報工学科

表1: TCPハードコアの設計結果

TCP	面積 [ $\mu m^2$ ]		消費電力 [mW]		配線負荷容量 [pF]	
	Transmit	Receive	Transmit	Receive	Transmit	Receive
port_ctr	0.17408 (5%)	0.12672 (4%)	1.1145 (6%)	0.8516 (4%)	4.58 (4%)	3.56 (3%)
data_ctr	0.94837 (25%)	0.91584 (27%)	4.7279 (23%)	6.0286 (30%)	27.4 (22%)	30.1 (26%)
window_ctr	0.43508 (11%)	0.23409 (7%)	1.9972 (10%)	1.4553 (7%)	11.1 (9%)	7.41 (6%)
checksum	2.25205 (59%)	2.15203 (62%)	12.2927 (61%)	12.042 (59%)	8.06 (65%)	74.3 (65%)
TOP	3.77407 (100%)	3.33206 (100%)	18.8503 (100%)	19.669 (100%)	148 (100%)	137.2 (100%)

表2: IPハードコアの設計結果

IP	面積 [ $\mu m^2$ ]	消費電力 [mW]
Service_info	0.352 (16%)	1.8465 (14%)
Frag_ctr	0.6254 (29%)	3.5368 (26%)
Checksum	0.9036 (43%)	6.4956 (47%)
Add_ctr	0.2534 (12%)	1.7016 (13%)
TOP	2.1196 (100%)	10.7182 (100%)

### 3. 評価

TCP/IP ハードコアの消費電力を解析するため、第2節で示した機能分割に基づき設計を行った。実際には、各機能ブロックをハードウェア記述言語 (Verilog-HDL) で記述し、synopsys 社の Design-Compiler によって論理合成を行った。なお、本設計では、東京大学大規模集積システム設計教育研究センターから提供される  $0.35 \mu m$  CMOS プロセス用ライブラリを用いた。

TCP/IP ハードコアの設計結果をそれぞれ表1, 2に示す。表1において、面積ならびに消費電力は論理合成ツールのレポートファイルから得た値であり、配線の影響は含まれていない。一方、表1の配線負荷容量に関しては、自動配置配線ツールを用いたレイアウトに基づき得た値である。表2の面積ならびに消費電力についても同様であるが、レイアウトに基づく配線容量の抽出は行っていない。

まず、TCPハードコアの消費電力について考察する。表1より checksum モジュールは全体の60%~65%を占める結果となり、全体に与える影響は極めて大きいことが分かる。このモジュールにはデータ送受信の信頼性を向上するために、エラー検出用チェックサムが含まれている。チェックサムでは、全てのデータ(ヘッダも含む)を15ビット毎に区切り、それらの足し込みを行う必要があり、多くの演算が必要となる。その結果、checksum モジュールの回路規模が大きくなり、引いては、多くの電力を消費する。checksum モジュールの次に大きな割合を占めるモジュールは data\_ctr である。このモジュールは全体の22~30%の消費電力を消費する。これは、本モジュールが、シーケンス番号ならびに応答確認番号を用いたデータの分割/復元といった複雑な処理を有するためである。

次にIPハードコアの消費電力について考察する。表2よりTCPハードコアと同様にchecksumモジュールにおいて最も電力を消費しており、全体の約半分ほどの割合を占めている。この理由はTCPハードコアの場合と同じように、エラー検出には多くの演算量が必要となるためである。2番目に全体に占める割合が大きいのはfrag\_ctrモジュールである。このモジュールでは全体の26%を占めている。その理由はfrag\_ctrのフラグメント機能はTCPのdata\_ctrの機能に似ており、非常に複雑な処理を必要とするためだと考えられる。

最後にTCPハードコアとIPハードコアの消費電力を比較する。TCPハードコアで消費される電力はIPハードコアで消費される電力の約2倍である。これはTCPがデータの送信用と受信用それぞれの回路が必要になるためである。以上より、TCP/IPパケット処理を低消費電力化するにはTCPのチェックサムモジュールを低消費電力化する事が最も有効である事が分かる。

### 4. おわりに

本論文ではTCP/IPハードコアの設計を行い、その消費電力解析を行った。本研究の最終的な目標は、将来、より高速化の進むネットワークに対応できる高速かつ低消費電力なTCP/IPハードコアを開発することである。今後、スイッチング確率を考慮したより正確な消費電力解析を行う予定である。

### 謝辞

本研究は東京大学大規模集積システム設計教育研究センターを通しローム(株)および凸版印刷(株)の協力で行われたものである。なお、本研究は一部、科学研究費補助金(課題番号: 14GS0218, 14702064)による。

### 参考文献

- [1] 村島 修一, "詳説図解TCP/IPエキスパートガイド", 秀和システム, May, 2003.
- [2] Srihari Makineni, Ravi Iyer, Communications Technology Laboratory Intel Corporation, "Architectural Characterization of TCP/IP Packet Processing on the Pentium M microprocessor", *10<sup>th</sup> International Symposium on High Performance Computer Architecture*, p.152, Feb. 2004.
- [3] 鈴木 憲一, 土屋 信明, "TCP/IPをマスターしよう", 株式会社アスキー, Feb. 2002.