

非同期式マイクロプロセッサの動向

A Perspective on Asynchronous Microprocessor by Takashi NANYA (Research Center for Advanced Science and Technology, University of Tokyo).

南谷 崇¹

¹ 東京大学先端科学技術研究センター

1. はじめに

今日までの情報処理技術の発展を支えてきたコンピュータ性能の飛躍的向上はVLSI技術の進歩に負うところが大きい。今後も、素子の微細化とチップの高集積化が一層進むと予測されている。しかし、コンピュータの誕生以来これまでずっと行われてきた「同期式」システム設計ではもはや今後のVLSI技術の進歩の恩恵を十分受けられず、プロセッサ性能が飽和点に近づきつつあることがはっきりしてきた。このため、この数年、これを解決する新しい設計アプローチとして「非同期式」システム設計の研究が急速に立ち上がり、非同期式マイクロプロセッサの研究開発もいくつか進んでいる。本稿では、現在の同期式設計が直面する問題点、非同期式設計の基本原則、非同期式マイクロプロセッサの開発事例を紹介し、その実用化への課題と将来の可能性を展望する。

2. 同期式プロセッサの性能限界

マイクロプロセッサに代表されるデジタルシステムあるいはデジタル電子機器がクロックと呼ぶ特別な方形波信号に同期してその動作を進める、というのは常識である。システムの動作解析、論理設計はすべて、クロック信号で定められる周期的タイミングを基にして行われる。アーキテクチャが同じならばプロセッサ速度はクロック周波数で決まる。クロック周波数は、プロセッサ内の任意の2つのレジスタ間のデータ転送に要する時間の最大値で決まる。したがって、プロセッサを速くしたければ、レジスタ間データ転送時間の最大値ができるだけ小さくなるように論理設計を行い、クロック周波数を上げればよい。実際、これまで、半導体製造技術の世代交替が起きるたびに、ゲート遅延の減少にほぼ見合ってクロック周波数が上がってきた。そうしてプロセッサの性能向上と大規模化が達成されてきた。

ところが、最近、様子が違ってきた。半導体集積回路構造の微細化が進むと、ゲート遅延は減少する。その一方、微細化で配線抵抗は増えるために同じ長さで

も配線遅延は絶対的に増大し、また、小さくなるゲート遅延に対して相対的にも増大する。さらに、チップ面積も大きくなり、搭載されるシステム規模が大きくなり複雑になってきたため、配線長自体も長くなる傾向にある。このため、VLSI設計に際して配線遅延が無視できなくなってきたのだ。いやそれどころか、むしろ配線遅延のほうが支配的要因になりつつある。図-1に示すように、現在のVLSIプロセス技術（0.25 μm ）でゲート遅延と長さ43 μm の（アルミ）配線遅延とはほぼ同程度である¹⁾。これからさらに微細化が進むにつれて、急激に配線遅延がチップ性能の支配要因になっていく。この傾向はアルミ配線が銅配線に置き換えられると若干緩和されるが、配線遅延が支配的になることに変わりはない。

配線遅延問題で真っ先に影響を受けるのがクロック配線である。LSIの微細化によってせっかく論理ゲートの速度が向上しても、その速度に見合った周波数でクロックをチップ全体に分配することができなくなるのだ。このため、すでに現在チップ性能は限界に近づきつつあり、クロック分配などのタイミング制御に多大の設計コスト、電力コストがかかるようになっていく。米国半導体産業協会（SIA）の予測¹⁾によれば、西暦2012年のシリコン技術による最先端マイクロプロセッサのオンチップ・クロック周波数は3GHzから10GHzに達するという。しかし、現在の同期式チップアーキテクチャでこれを達成するのは明らかに無理だ。真空中の電磁波信号でもチップ寸法と同程度の3cmを進むのに0.1nsを要するからだ。他の媒体ならもっと時間がかかることはいうまでもない。

要するに、今後のデバイス技術の進歩の果実を享受してシステム性能を向上させようとするなら、チップ全域に分配するクロック信号はもはや使えないのだ。

3. 初めは非同期式だった！

クロック分配に起因する同期式システムの性能限界を突破する解の1つは、「クロックを使わない」ことである。すなわち、非同期式設計スタイルを採用することである。クロックを使わなければ、局所計算の並

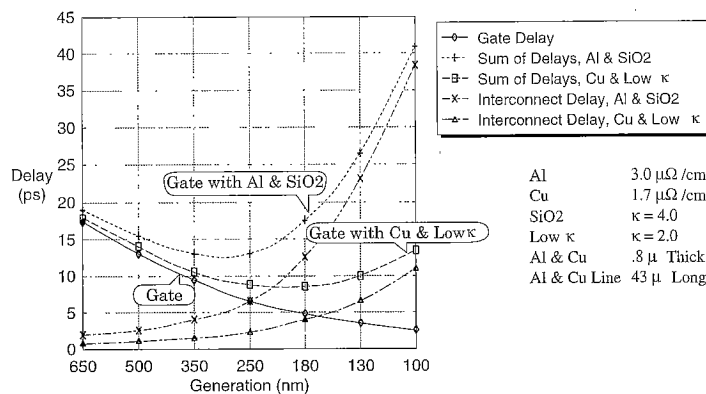


図-1 技術世代推移に対するゲート遅延と配線遅延¹⁾

列化と分散制御によって配線遅延の影響を最小化できる。ゲート遅延減少の効果を最大限に享受するアーキテクチャや回路構成の採用が可能になる。その結果、今後なお予想される微細化プロセス技術の進歩の恩恵をシステム側が十分に受けることができる。

非同期式設計は新しい技術ではない。歴史的には、初期のデジタル装置は非同期式で作られていた。電話交換の制御装置を電磁リレーで作っていた頃はクロックの概念はなかったはずだ。第2次大戦末期、米国ベル研やIBMなどで兵器研究のため電磁リレー式自動計算装置を開発していた頃、今のクロックのようなタイミング方式が生まれたいい。電子式になってからも1940年代には真空管を用いた非同期式演算装置あるいは計算機がいくつか試作されたようだ。しかし、コンピュータがいわゆる第2世代（トランジスタ時代）に移ってからは、周知の通り、今日までほとんどすべてのコンピュータは共通クロック信号を用いる同期式で設計されている。特に、1970年代以降のLSI時代になってからは、あらゆるデジタルシステムは同期式設計を基本とするようになった。

つまり、初めにまず非同期式回路があり、1940年代の半ばに誰かがクロックを発明して同期式回路が生まれたのだ。したがって、同期式が生まれる前の設計技術に対して「非」をつけて非同期式と呼ぶのは本来はおかしい。信号遷移の因果関係に従って動作するのだから、たとえば、因果式と呼ぶのがふさわしい。トランジスタの出現で素子特性が均一になったのでクロックを用いる非因果式が今日まで続いてきたが、VLSI技術の進歩で再び因果式による設計が必要になってきた、と考えれば分かりやすい。まさに「因果は巡る」ということである。しかし、ここでは今までどおり、クロックのない回路を非同期式と呼ぶことにしよう。

さて、非同期式設計スタイルを採用することによる利点をまとめると、次のようになる。

(1) 高速性：同期式システムの性能がレジスタ間デ

ータ転送の最悪ケース遅延で決まるのに対して、非同期式システムでは局所計算の平均遅延だけで決まる。したがって、信号伝播の平均距離を最小化するチップアーキテクチャの採用によって今後の素子速度向上の効果をそのまま享受した高速計算を実現できる。

(2) 信頼性：信号遷移の因果律に基づいて素子なりに動作を進めるため、予測不能な環境変化（電源電圧、温度など）やパラメータ変化に起因する遅延変動の影響を受けない高信頼性システムが実現できる。

(3) 低消費電力：現在の主流であるCMOSゲートの電力消費は、近似的には、回路中の信号遷移の量、すなわち、単位時間当たりに充電／放電の起きる配線数、に比例する。同期式システムでは、システム全域へのクロック分配による電力消費が大きな割合を占めるのに対して、非同期式システムではクロックがなく、与えられた計算の実行に必要な時に必要な場所でしか信号遷移は起きない。動作が要求されない部分回路の電力消費はごくわずかな漏れ電流によるものだけである。したがって、性能を犠牲にすることなく電力消費を低減できる。

(4) 設計容易性：異なる2つのモジュール間のインタフェースでは要求／応答のプロトコルに従う以外には同期式で要求されるようなタイミング調整の必要がない。そのため大規模システム設計の複雑さをモジュラー設計によって克服できる。

4. 非同期式システムの動作原理

デジタルシステムを構成する素子や配線の遅延はさまざまな要因で変動する。論理設計手法、チップレイアウト、プロセス技術、実装技術など、設計製造段階での変動要因に加えて、電源電圧、環境温度など、稼働する環境によって大きく変動する。さらに処理されるデータにも依存する。したがって、システムの生涯を通じて実際に起こり得る遅延変動の様子を推定

し、その振舞いを表す適切な遅延モデル（遅延仮定）に基づいて設計を進めることになる。

もし、現行の同期式システムのように、設計時に推定した最大遅延はシステムの生涯を通じて不変であると楽観的に仮定して設計を行うと、回路構成の効率はよくなるが、上記の要因で遅延変動が起きた場合にはクロックのタイミングがずれて誤った動作を引き起こしやすい。すなわち、信頼性は低下する。一方、DIモデル²⁾やQDIモデル³⁾のように、素子遅延や配線遅延は互いにばらばらで自由に変動し得ると悲観的に仮定して設計すると、遅延変動に対する信頼性は完全に保証するが、ある意味でムダが生じ、回路量が増えたり性能低下が生じる。

したがって、設計の前提となる遅延モデルは素子技術、システム実装技術、動作環境に対して十分なレベルの性能とタイミング信頼性を達成するものでなければならない^{2), 4)}。

同期式と非同期式の違いは、2値信号をレジスタ、メモリなどの記憶素子に書き込む際に共通クロックを用いるかどうか、という点にある。デジタルシステムの基本動作はレジスタ間データ転送である。同期式では、クロックパルスがデータを書き込むタイミングを定めている。このため、クロックパルスの到着する前に有効なデータが転送先レジスタの入力端子へ到着していなければならない。システム全体の動作も共通クロックに合わせた周期で進む。

これに対して非同期式では、データ転送が必要（かつ可能）になったら転送元レジスタへ要求信号を出す。その結果、転送元レジスタから送出されたデータは変換回路を経て転送先レジスタへ書き込まれ、その完了を示す応答信号が生成される。この応答信号が次のデータ転送を要求するトリガになる。

クロックなしで、Nビット幅のデータを転送先レジスタへ書き込むタイミングをどう決めるか？ これは非同期式論理設計における重要な選択の1つである。一般にデータパスの遅延は、実行される命令、データによって、またビットごとにも、ばらつきがある。そこでおおまかに次の2つの極端な場合に分ける。

(1) 遅延のばらつきが非常に大きい場合：

1ビットデータを2本の信号線で表す2線式データ表現を用いる。したがって、データ幅は2Nになる。ビットごとに初期状態を(0,0)として、有効データの到着（書き込みタイミング）を次のような遷移で表現する。

(0,0) → (0,1) "0"が到着

(0,0) → (1,0) "1"が到着

(0,1), (1,0) → (0,0) 初期状態へ復帰

データパス上のどのインタフェースでも、データ転送を実行するために初期状態(0,0)から有効データ(0,1)または(1,0)へ遷移する期間（稼働相と呼ぶ）

と次のデータ転送に備えるために初期状態へ復帰する期間（休止相と呼ぶ）を交互に繰り返す。これを2線2相式と呼ぶ。

(2) 遅延のばらつきが非常に小さい場合：

Nビットデータの遅延がほぼ一定で、その上限値Dが既知の場合、値Dを考慮した適当な遅延素子を挿入した1本のready信号をNビットデータに付加する。ready信号の0 → 1遷移が転送先レジスタへの有効データの到着を示す。これを束データ方式と呼ぶ。

すなわち、非同期式システムでは、共通クロックを使わない代わりに、Nビットのデータパスを実現するのに、最も楽観的な場合で(N+1)ビット、最も悲観的な場合には2Nビットのデータ幅が必要になる。実際には、この範囲内のデータ幅で性能と信頼性のトレードオフを考慮した種々の設計があり得る。

5. 非同期式マイクロプロセッサの開発事例

クロックをまったく用いない完全非同期式マイクロプロセッサの開発プロジェクトは現在、東大／東工大における筆者のグループのほかに、英国Univ. of ManchesterのAMULET 2e⁵⁾と、米国CalTechのMiniMIPS⁶⁾がある。AMULET 2eはARM社の32ビットRISCプロセッサの非同期版であり、0.5 μ m, 3層メタル, CMOSプロセスのフルカスタム設計で1996年10月にチップとして実現されている。電源電圧3.3VでのDhrystone V2.1ベンチマークで40 MIPSの速度性能であるが、その消費電力は0.15Wと少ない。一方、MiniMIPSは、MIPS R3000の命令セットを実現するが一部機能（TLB, 外部割り込みなど）を省略した非同期式簡易型MIPSである。まだ0.6 μ m CMOSプロセスによる設計段階にあり、チップはできていないが、シミュレーションによれば、3.3V電源、電力消費7Wで280MIPSの速度性能を達成している。非同期式システムの高速度性を具体的に示す評価結果といえる。

ここでは、筆者のグループで1997年2月に完成させ

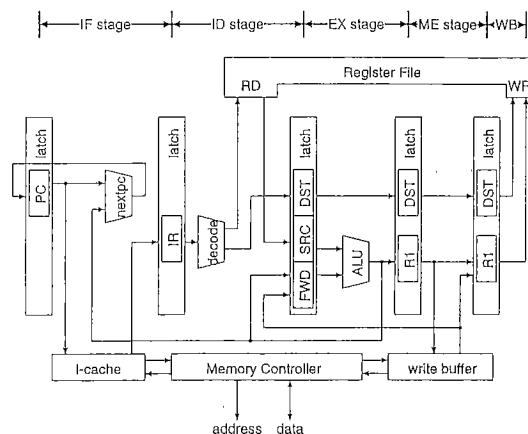


図-2 TIITAC-2のプロセッサ構成

た非同期式マイクロプロセッサTITAC-2^{7)~9)}を紹介しよう☆.

TITAC-2はMIPS-R2000の命令セットを実現した32ビットのマイクロプロセッサである. もちろんクロックはない. しかし, プログラマから非同期式かどうかは見えない. 同期式と同じようにCプログラムが走る. プロセッサ構成(図-2)は, R2000と同様, 命令フェッチ(IF), デコード(ID), 実行(EX), メモリアクセス(ME), ライトバック(WB)からなる5段パイプライン構造で, 40本の32ビットレジスタ(うち, 8本はカーネルモードのみ), 8KB命令キャッシュ, 例外処理, 外部割り込み, 記憶保護機構を備えている.

TITAC-2では, 楽観的すぎも悲観的すぎもしない現実的なモデルとして, Scalable-Delay-Insensitive(SDI)モデル⁹⁾と呼ぶ新しい遅延仮定を導入し, それを基に論理設計を行った. SDIモデルは, 回路要素の絶対的な遅延変動の大きさに上限はないが相対的な変動の幅には上限がある, と考えるもので, 以下のような仮定である.

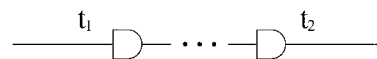
「任意の2つの回路要素(ゲートまたは配線) C_1 , C_2 の遅延を d_1 , d_2 とすると, $D=d_1/d_2$ を C_2 に対する C_1 の相対遅延と呼ぶ. 回路の設計段階で推定される相対遅延を De で表し, システムの生涯を通じて起こりうる実際の相対遅延を Da で表すと, $R=Da/De$ は相対遅延変動率を表す. このとき, 上限値(定数) K が存在して, $1/K < R < K$ が常に成り立つ, と仮定する.」

SDIモデルを適用した設計の効果を説明しよう. 今, 回路中の2つの信号遷移 t_1 と t_2 に関する動作仕様で, t_1 が t_2 より必ず先に起きなければならないとする. これを実現するために, DIモデルのもとでは, 遅延に上限値がないので図-3 (a)に示すように, t_1 を原因としなければ t_2 は決して起きないように回路を構成する必要がある. QDIモデルのもとでは, 分岐配線には遅延差はないとするため図-3 (b)に示すように, 原因となる幹を t_1 と共有する枝によって t_2 が起きてよい. これらに対してSDIモデルのもとでは, 図-3 (c)に示すように, t_1 と t_2 の共通原因となる幹からの経路の遅延をそれぞれ d_1 , d_2 としたとき, $K \cdot d_1 < d_2$ が成り立つような回路構成であればよい. したがって, 遅延変動に対して十分なレベルの信頼性を確保しつつ, システムの速度性能を向上させることができる.

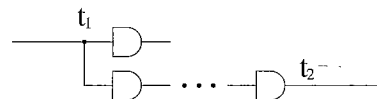
SDIモデルを適用する設計は以下の指針に従う.

- (1) システム全体を複数の機能ブロックに分割する.
- (2) 各ブロックおよびその接続をQDIモデルに基づいて設計する.

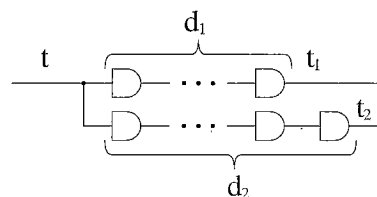
☆ TITAC-2の完成は過去数年に渡る東工大/東大・南谷研究室メンバーの努力の結晶であるが, 特に上野洋一郎, 桑子雅史, 高村明裕, 今井雅, 藤井太郎, 小沢基一, 深作泉, 石川誠, Rafael Morizawaの諸君の奮闘によるところが大きい.



(a) DI実現



(b) QDI実現



(c) SDI実現

図-3 SDIモデルによる回路実現

- (3) 使用するプロセス技術およびブロックサイズから相対的遅延変動率の上限値 K を決定する.
- (4) 各ブロックで $K \cdot d_1 < d_2$ が成り立つような局所変換を行う.

SDIモデルを正当化するために, レイアウトには一定の制約が課される. TITAC-2では $K=2$ とした. これを正当化するために, 使用した $0.5\mu\text{m}$ CMOSプロセスに対して, 各ブロックは $1.93\text{mm} \times 1.93\text{mm}$ の範囲に収まるようなレイアウト設計を行った.

TITAC-2チップは, $0.5\mu\text{m}$, 3層メタル, 3.3V電源CMOSプロセスのスタンダード・セル・ライブラリ, および新規に作成した専用マクロ・セル(C素子, アービタ, 2線式全加算器など)を用いて設計した^{☆2}. 図-4にそのチップ写真を示す. 12.15mm (コアサイズは 10.55mm)四方のチップ上に約50万トランジスタと8.6Kバイトのメモリマクロが集積されている. Dhrystone V2.1ベンチマークによる実測性能は, 室温, 電源電圧3.3Vで54.1VAX MIPS, 消費電力2.11Wである.

TITAC-2の特徴はそのDelay-Insensitivityにある. 遅延変動を生じさせるような環境の変動, たとえば, 電源電圧やチップ温度の変動に対しても正常に動作する. ただし, 当然のことながら, 速度性能と消費電力は変化する. 図-5に電源電圧を1.5Vから4.0Vまで連続的に変化させた場合のTITAC-2の速度性能(MIPS)と消費電力(W)の実測値を示す. このように, 計算パワーを要求される場合には, 消費電力を上げて高速計算を実行し, そうでない場合は計算性能

☆2 チップ実現に関してNEC(株)の高田正日出, 山田八郎, 福岡雅夫, 山品正勝, 藤田修, 藤本裕樹, 岡本冬樹の諸氏に助言やご協力をいただいた.

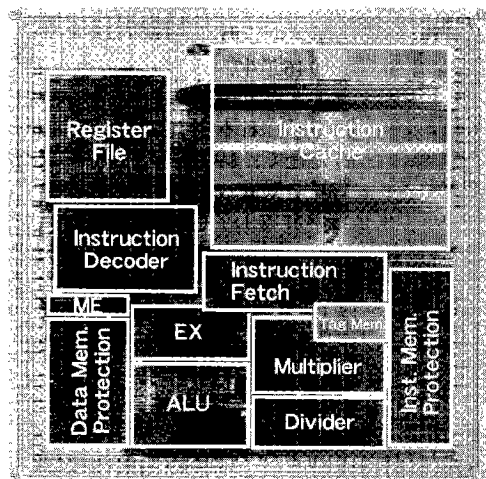


図-4 TITAC-2のチップ写真

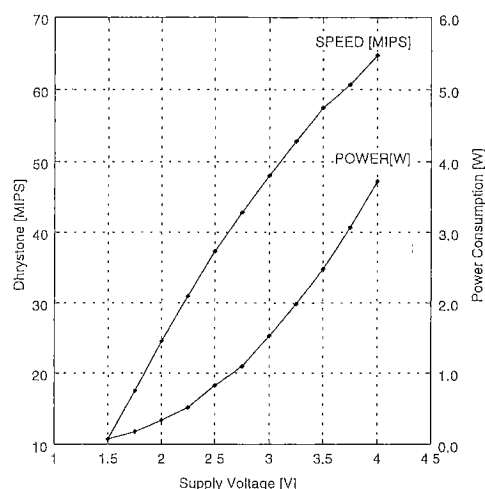


図-5 電源電圧に対するTITAC-2の速度性能と消費電力

を落として消費電力を節減できる。

6. 実用化への課題と将来展望

非同期式システムの実現可能性について論じる時期はすぎた。実用レベルのマイクロプロセッサをクロックなしで実現する設計技術はすでに用意されている。今必要なことは、素子性能の向上をシステム性能の向上に直線的に反映し得る非同期式システムの特性を十分に引き出すアーキテクチャの開発と、それを実現する非同期式設計技術を現在の成熟した同期式設計と並ぶ選択肢の1つとして設計者に提供できる設計環境を構築することである。そのためには、具体的課題として、レジスタ転送レベル記述段階での性能予測手法の開発、設計支援／検証ツールの開発、テスト手法、過渡性フォールト（ノイズなど）への対策などがあげられる。

高性能マイクロプロセッサは、ソフトウェアからプロセス技術まであらゆるレベルで最先端の技術を総合して実現される。現在の最先端の同期式マイクロプロセッサと大学の研究室で開発されたTITAC-2, AMULET2e, MiniMIPSなどの非同期式マイクロプロセッサとでは設計の環境、資源、スキルなどの条件がまったく異なるので、簡単には比較できないが、これらの非同期式プロセッサの性能は、現在の技術水準でもすでに同期式プロセッサと十分比較し得るレベルに達しているといえることができる。今後、配線遅延が支配要因となる傾向が進めば、クロックを用いない非同期式システム設計の優位性が一層顕著になると思われる。実際、前述のSIAのロードマップでも、2003年頃に実現が予想される0.13 μ m技術からは、非同期式チップアーキテクチャが必要になると予測している¹⁾。

まだ配線遅延がそれほど支配的とはいえない現在の技術領域でも非同期式設計が有効だと考えられる応用分野としては、携帯電子機器などの電力消費の低減が

必要なシステム、環境変動に対する耐性が要求されるシステム、多数の機能が複合する大規模集積システムなどがあげられる。

非同期式設計技術は汎用デジタル技術である。しかし、配線遅延が真に支配的になると予想される2003年以降でも非同期式が同期式に完全にとって代わると考えるのは現実的ではない。最もあり得るシナリオは応用分野に依存した同期式設計と非同期式設計の住み分けと融合だろう。すでに、外見はこれまで通りの同期式プロセッサであるが、内部の命令デコードや、演算部などは高速化を狙って非同期式設計スタイルを取り入れたチップなども出現し始めている。さらに微細化と高集積化が進んだ段階での先端VLSIでは、システム全体が非同期式設計になることは必須となろうが、局所的には高周波クロックを用いた同期式設計が用いられることも大いにあり得る。いずれにしても、配線遅延が支配的になるにつれて、VLSI設計スタイルは大きく変わっていくことになろう。応用分野によって性能、消費電力、信頼性のトレードオフは異なる。そのトレードオフに応じた同期／非同期融合アーキテクチャの開発が今後の課題である。

謝辞 本研究の一部は科研費補助金基盤研究（B）09480049、および（株）半導体理工学研究センターとの共同研究によるものである。

参考文献

- 1) The National Technology Roadmap for Semiconductors, 1997 Edition, SEMATECH, Inc. (Dec. 1997).
- 2) 南谷 崇：非同期式プロセッサ／超高速プロセッサを目指して、情報処理, Vol.34, No.1, pp.72-80 (Jan. 1993).
- 3) Nanya, T., Ueno, Y., Kagotani, H., Kuwako, M. and Takamura, A.: TITAC: Design of a Quasi-Delay-Insensitive Microprocessor, IEEE Design & Test of Computers, Vol.11, No.2, pp.50-63 (Summer 1994).
- 4) Nanya, T.: Challenges to Dependable Asynchronous Processor Design, in Logic Synthesis and Optimization, Kluwer Academic Publishers (Nov. 1992).
- 5) Furber, S.B., Garside, J.D., Temple, S., Liu, J., Day, P. and Paver, N.C.: AMULET2e: An Asynchronous Embedded Controller, Proc. of ASYNC'97, pp.290-299 (Apr. 1997).

- 6) Martin, A.J., Lines, A., Manohar, R., Nystrom, M., Penzes, P., Southworth, R., Cummings, U. and Lee, T.K.: The Design of an Asynchronous MIPS R3000 Microprocessor, Proc. ARVLSI'97 (Sep. 1997) .
- 7) 高村, 桑子, 南谷: 非同期式プロセッサTITAC-IIの論理設計における高速化手法, 電子情報通信学会論文誌, Vol.J80-D-I, No.3, pp.189-196 (Mar. 1997) .
- 8) Nanya, T., Takamura, A., Kuwako, M., Imai, M., Fujii, T., Ozawa, M., Fukasaku, I., Ueno, Y., Okamoto, F., Fujimoto, H., Fujita, O., Yamashina, M. and Fukuma, M.: TITAC-2: A 32-bit Scalable-Delay-Insensitive Microprocessor, Symp. Record of HOT Chips IX, Stanford, pp.19-32 (Aug. 1997) .
- 9) Takamura, A., Kuwako, M., Imai, M., Fujii, T., Ozawa, M., Fukasaku, I., Ueno, Y. and Nanya, T.: TITAC-2: A 32-bit Asynchronous Microprocessor based on Scalable-Delay-Insensitive Model, Proc.ICCD'97, pp.288-294 (Oct. 1997) .

(平成9年12月22日受付)



南谷 崇 (正会員)

1946年生。1969年東京大学工学部計数工学科卒業。1971年同大学院修士課程修了。日本電気(株)中央研究所勤務を経て、1981年東京工業大学情報工学科助教授、1989年同電気電子工学科教授。1995年東京大学計数工学科教授。1996年同大学先端科学技術研究センター教授。主として、非同期システム論、VLSI設計論、フォールト・トレランス技術に関する研究に従事。工学博士。電子情報通信学会論文賞、大川出版賞受賞。著書「順序機械」(岩波書店)、「フォールトトレラントシステムの構成と設計」(楳書店)、「フォールトトレラントコンピュータ」(オーム社)など。IEEE, 電子情報通信学会, 情報通信学会各会員。

E-mail:nanya@hal.rcast.u-tokyo.ac.jp