

シチズン時計株式会社におけるCADによるLSIの論理設計システム

LSI Logic Design System by CAD in Citizen Watch Co., Ltd

電子式時計(クォーツウォッチ)では、機能を豊富にするには複雑な論理回路設計を行なうことが必要になる。論理回路設計が複雑になればなるほど、その論理回路設計の過程をすべて人手で行なうことは困難となる。

そこで、論理回路設計の検証を容易にするための論理シミュレーションを中核とした、LSIの論理回路設計をサポートするトータルシステムの開発を行なった。

このトータルシステムにより、短期間、かつ正確な論理設計が行なえるようになった。

鈴木長建*	Chōken Suzuki
和方 淳*	Tōru Wakata
木野晴夫*	Haruo Kino
松島輝一**	Teruichi Matsushima
和田 崇***	Takashi Wada

1 緒 言

LSIを使用する製品では設計期間の短縮と工数削減を図るためのCAD(Computer Aided Design)システムの開発が重要な課題となっている。この課題に対処するために、論理回路設計の検証だけを目的としたバッチ形の論理シミュレータが開発されている。しかし、バッチ形の論理シミュレータは入力データの作成作業と出力の確認作業が人手によって行なわれる。そのため、一つの論理回路設計の作業が完了するまでのターンアラウンドタイムが長くなる。

そこで、入力データの作成と出力の確認作業を対話形システムでサポートする方式のLSIの論理シミュレーションシステムのトータルCADの開発を行なった。

2 開発の背景と概要

時計が電子式となり、デジタル時計が世に出て久しい。クォーツ化時計が市場に現われたころに比べると、主として論理ICの高集積化技術の進歩が著しい。1個のチップに搭載される論理的機能が豊富となり、いわゆる複合化商品としてのデジタル時計の需要が急成長の状況にある。

時計用のLSIは、主としてC-MOS(Complementary - Metal Oxide Semiconductor)が使用される。C-MOSは数千ゲートの規模まで広がり、更に高集積の新製品が開発されつつある。時計用論理回路設計も、それだけ複雑な論理回路を設計することが要求され、論理回路図を仕上げるのが精一杯のところきている。そして、このような大規模で複雑な論理回路の検証を、人手で行なうことは時間的に不可能である。

論理を検証する手法としては、論理シミュレーションによる方法と、市販のICなどを利用してLSIと等価な論理をもつ装置を作り、論理動作の検証を行なう方法とがある。従来、論理シミュレータとして開発された多くはコンピュータ設計のためのものであり、それがLSIの論理シミュレーションに流用されているのが実情である。

電子式時計で既存の論理シミュレータを利用する場合、時間の深さ(すなわち、20年から100年の間の時間を保証する。)から、電子計算機の計算時間が非常に長くなり、設計コスト面から、この種の論理シミュレータを導入することは難しい

という評価がシチズン時計株式会社の定説であった。

そこで、電子計算機の処理時間短縮のアルゴリズムを確立するとともに、利用面から見た使いやすさを重視した、LSI用論理シミュレータを開発した。次にバッチ形論理シミュレータでは手書論理回路図から論理シミュレーション用の論理回路網接続情報をコーディングして入力する。そのため、ゲート数が多いことによる初期トラブルが多発し、手書論理回路図と等価な論理回路網接続情報を作成するのに多大な時間を要すること、及びROM(Read Only Memory)、RAM(Random Access Memory)の採用によるトラブルも発生した。これらを解決する手段として、HITAC G-710システムを利用し、対話形による論理回路設計システムと論理回路網接続情報の自動変換システムの開発を行なった。

3 デジタルウォッチの設計・製作手順

デジタルウォッチの設計手順を図1に示す。論理設計の目的は、時計仕様を満たす論理構成をゲートレベルで明確に示すことにある。設計結果は、論理回路図として表現される。この論理回路図を基にして、ICマスク原画が作成され、ICの

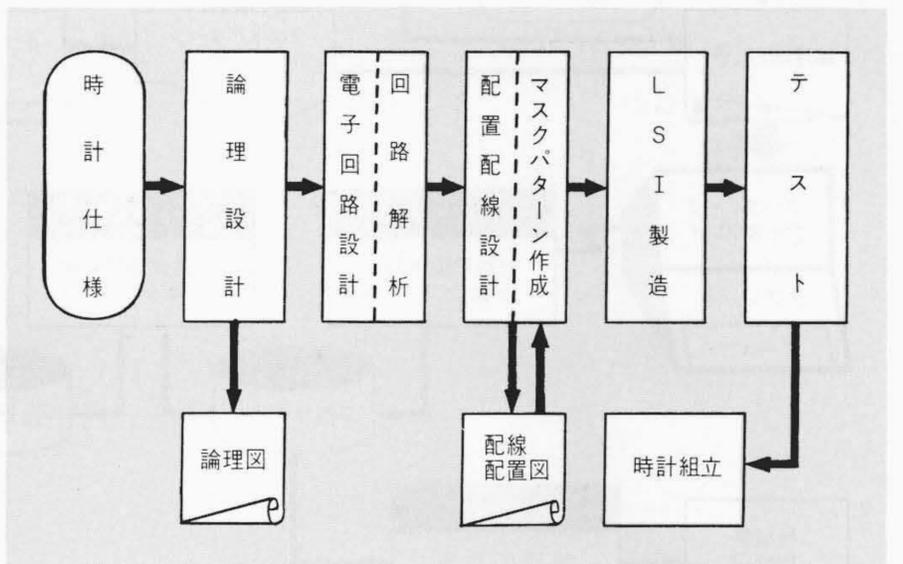


図1 デジタルウォッチのLSI設計・製作手順 電子回路設計及び回路解析は、特定のLSIについてだけ行なう作業である。

* シチズン時計株式会社開発部 ** シチズン時計株式会社電算室 *** 日立製作所ソフトウェア工場

チップが製作される。ここで時計用論理回路の特長を挙げると次に述べるようになる。

- (1) 論理的機能が豊富でゲート数が多い(2,000~5,000ゲート)。
- (2) 時間のファクターが長い(20~100年の時間としての保証)。

時計仕様に基いた論理設計が正しく行なわれていることの必要性は、ICプロセスの手番との関係から要求される。誤りのある論理に基づいて製作されたLSIを修正することは、現状の技術では不可能である。したがって、ICマスク原画を再製してLSIを作り直すとなると、何箇月もの時間的損失と場合によっては数千万円から数億円もの金銭的損失を生ずるとともに、新製品の出荷計画が大幅に遅れることになる。

このため、論理シミュレーションシステムによる設計確認をルーチンワークとして行なえることが、設計精度を上げ、かつ設計期間の短縮をもたらすことになる。

4 論理設計におけるCADシステムの開発

4.1 開発のねらいとシステムの全体構成

(1) 開発のねらい

時計用論理回路を論理シミュレータで処理実行する場合、時間系の深いことによる電子計算機の処理時間を小さくすることが重要な課題であった。そこで、まずシミュレーション時間の少ないアルゴリズムにするために、試行錯誤を行ないながら各種の工夫を凝らして、実用化できるアルゴリズムを見つけた。

更に、論理シミュレーションシステムとしてのトータル化を図り、利用が容易なCADシステムとするための条件整備を行ない、次のようなねらいをもたせることにした。

- (a) 手書論理図を対話形式で入力して、論理回路図ファイルを作成し、それに基づいて論理回路網接続情報を自動生成する。
- (b) テストパターン(シミュレーションの条件)の記述が簡単で、記述量が少なくできること。
- (c) 論理エラーを起こす箇所が会話形式で検索できること。
- (d) 論理設計ミスを見付けやすい出力形式を組み入れること。

- (e) シミュレーション完了時にICチップサイズなどの見積り資料が作成できること。
- (2) システムの全体構成

開発のねらいをシステムとして実現するために、図2に示すように大きく六つのフェーズに分けて、システムを構成することにした。本システムの開発は、段階別の展開を行なうことにした。開発の順序は同図に示すとおりで、システム効果の大きいフェーズから順に優先的に建設を行なった。以下に各フェーズの機能の概要について、開発順に従い述べる。

4.2 論理シミュレータの概要

4.2.1 論理シミュレータの構成

論理シミュレータは、次に述べる三つの機能から構成されている。

- (1) 論理接続情報記述言語変換プログラム
- (2) テストパターン情報記述言語変換プログラム
- (3) シミュレーションプログラム

4.2.2 論理接続情報記述言語変換プログラムの機能

論理シミュレータ用の回路接続情報などの入力体系の一般的な構成は図3に示すようになる。以下に、その概要について説明する。

(1) 機能構成

(a) TITLEプロセッサ

シミュレーションの結果をシステムがプリントアウトするとき、タイトル文字を指定する機能である。

(b) NO-PRINTプロセッサ

シミュレーションの結果を、システムがプリントアウトしないことを指定する機能である。

(c) TRACEプロセッサ

CRT(Cathode Ray Tube)ディスプレイからシミュレーション結果をトレーシングすることを、システムに指令する機能である。これにより、システムは全シミュレーションデータをディスクファイルにロギングし、設計者はCRTディスプレイを介して、ライトペンで対話的にトレースが行なえる。

(d) MODELプロセッサ

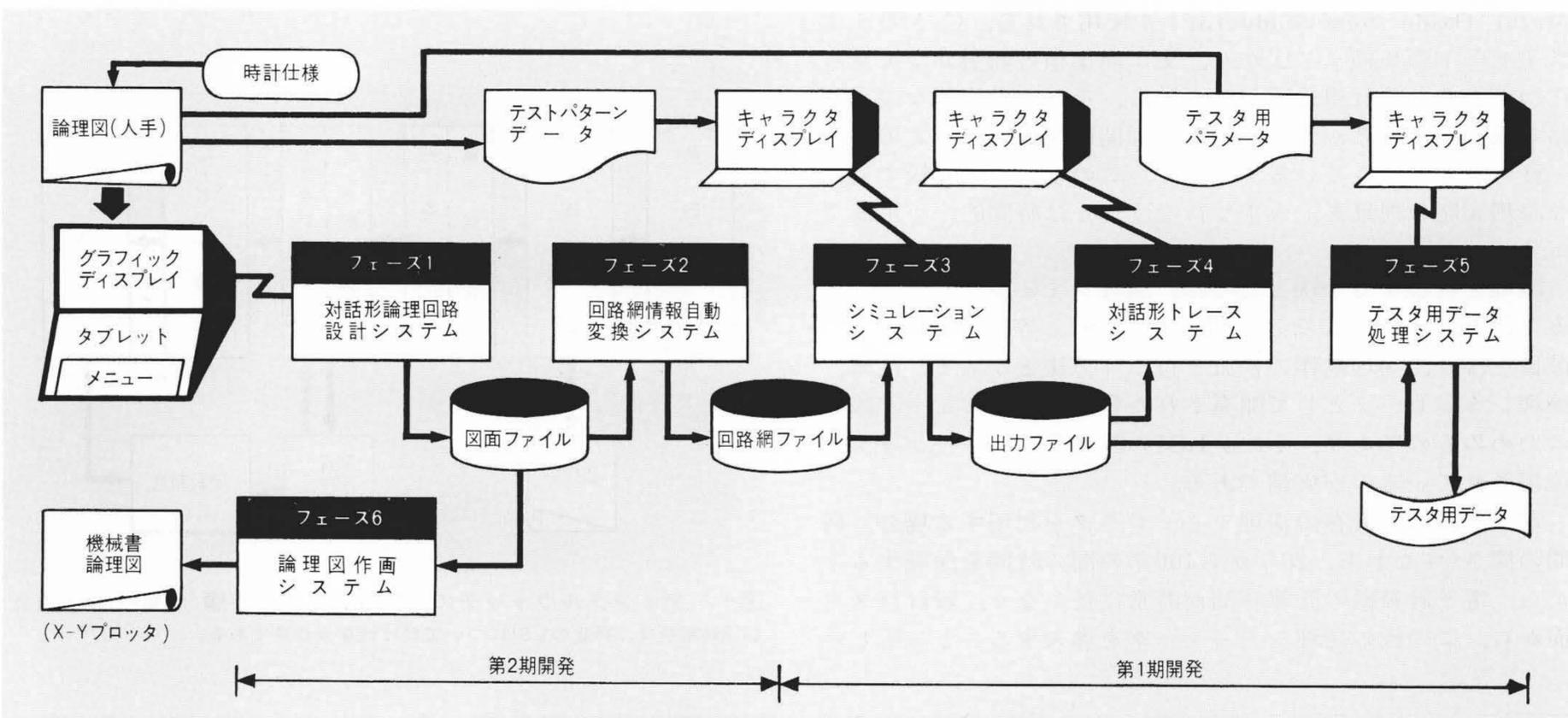


図2 システム全体構成 システムの全体構成と開発順序の関係を示している。第2期開発が完了後トータルシステムとなった。

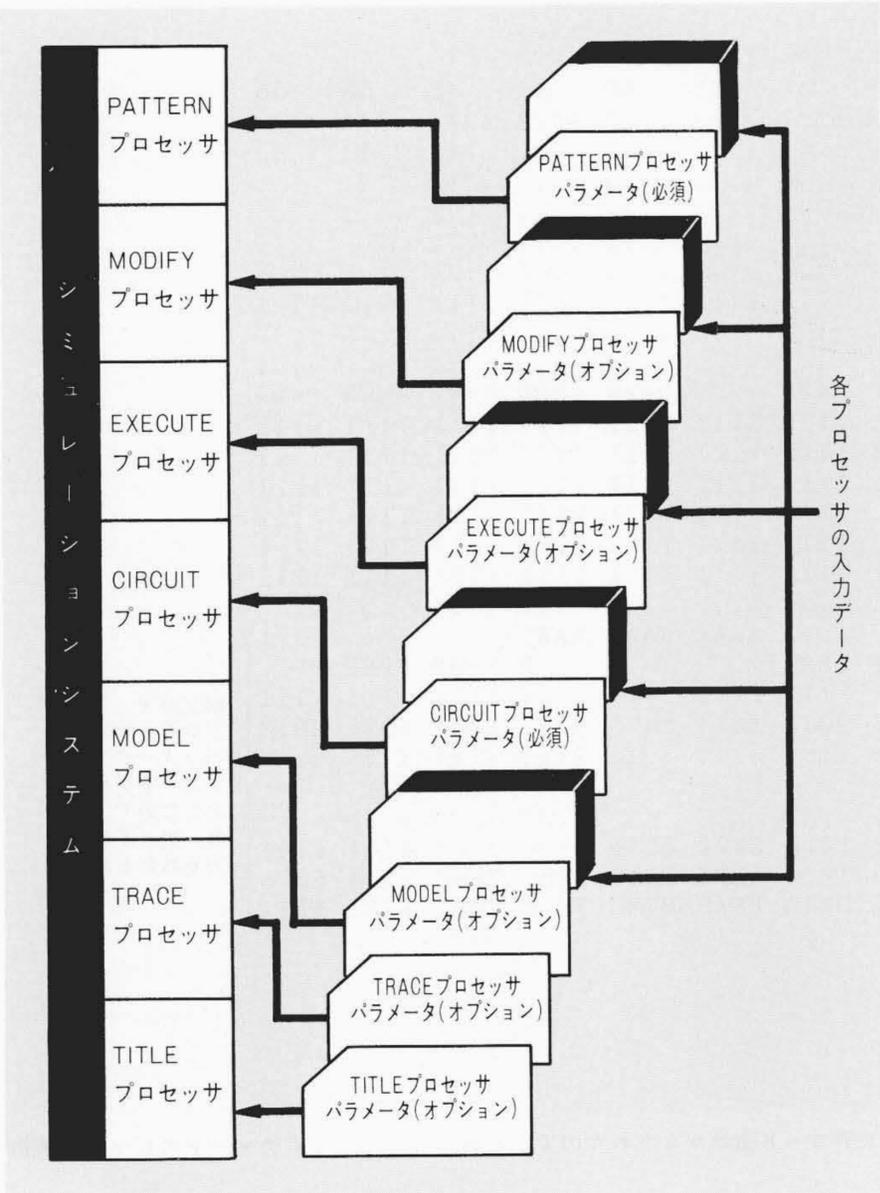


図3 回路接続情報記述言語の構成例 本ケースは、標準的な構成例である。シミュレーション結果のプリントアウトをしないことを指示するNO-PRINTプロセッサが必要なときは、TITLEプロセッサのパラメータの前にそのパラメータを置くことになる。

大規模回路を記述する際、簡略化のためにモデルとして回路の一部を定義する機能である。定義されたモデルは、名前前の参照だけで使用できる。モデルはシステムモデルとユーザーモデルとに分けられ、システムモデルは回路設計者に共通性があり、一般的なものに対してライブラリに登録されているので、個々に定義する必要はない。

(e) CIRCUIT プロセッサ

回路接続情報、初期化条件、固定値などを定義し、シミュレーションのための論理接続情報を用意する機能である。

(f) EXECUTE プロセッサ

シミュレーションの実行を行なう機能である。その結果は、ディスクファイルにロギングされる。ロギング方法は、本プロセッサで指定する。プリントアウトはNO-PRINTプロセッサに従い、プリントアウト後のロギングファイルを消去するかどうかはTRACEプロセッサがあるかないかによる。

(g) MODIFY プロセッサ

論理回路のシミュレーション結果を、システムがタイミングチャートとしてEXECUTEプロセッサの指定により、フォーマット化してプリントアウトするが、特に、デジタル系の論理回路は出力端子数が多く、かつ端子チャートでは実際の時計表示とのイメージにつながらないため、出力をモデファイしてチャートを見やすくする情報を指示する機能である(デコード記述の指示)。

(h) PATTERN プロセッサ

本プロセッサについては、後の4.2.3で説明するテストパターン情報記述言語変換プログラムを参照されたい。

図3は、バッチシステムで実行するときの論理接続情報記述言語の入力順序構成を示したものである。後述する対話形回路設計システムと論理回路網接続情報の自動変換システムのサポートにより、MODELプロセッサとCIRCUITの情報は自動生成される。

(2) 論理シミュレータでサポートするゲートの種類

- (a) INVERTER
- (b) DELAY
- (c) NAND (Not AND)
- (d) AND
- (e) OR
- (f) C-MOS Transmission Gate
- (g) Exclusive NOR
- (h) Exclusive OR
- (i) Wired OR
- (j) 入力端子
- (k) 出力端子

(3) ゲートの真理値

本論理シミュレータでは、表1に示す真理値のサポートを行なっている。

(4) シミュレーション結果の出力

EXECUTEプロセッサのパラメータで、シミュレーション結果のプリントモードを指定する。プリントモードには次の3種類がある。

(a) タイムチャートの出力形式指定

- (i) CIRCUITプロセッサで使われる全素子を出力対象とする。
- (ii) システムモデル及びユーザーが定義したテデル以外の素子を出力対象とする。
- (iii) システムモデル以外の素子を全部出力対象とする。
- (iv) 特定の素子だけを出力対象とするとともに、High状態の出力文字を指定する。

(b) 動作率チェックの指定

ある素子の入力数が n のときは 2^n 個の事象の組合せが発生する。一方、シミュレーションの結果、ある素子の入力値の変化の組合せの事象発生数が得られる。そこで、次式により動作率が求められる。

$$\text{動作率} = P / 2^n \times 100 \quad (\text{ここで、} P: \text{発生組合せ事象数})$$

(c) 検出率チェックの指定

ある素子の出力値の変化の組合せは 2^1 個(High状態か、Low状態の二通り)あると考えて、シミュレーションの

表1 ゲートの真理値 シミュレータでの真理値の体系を示す。

5 値 レベル		そ の 他 の 状 態	
真理値状態記号	真 理 値	真理値状態記号	真 理 値
1	High状態	*	0又は1の状態
<	中間値状態	@	OPEN状態
=	中間値状態	?	0又は1、又は中間値又は発振の不安定状態
>	中間値状態		
0	Low状態		

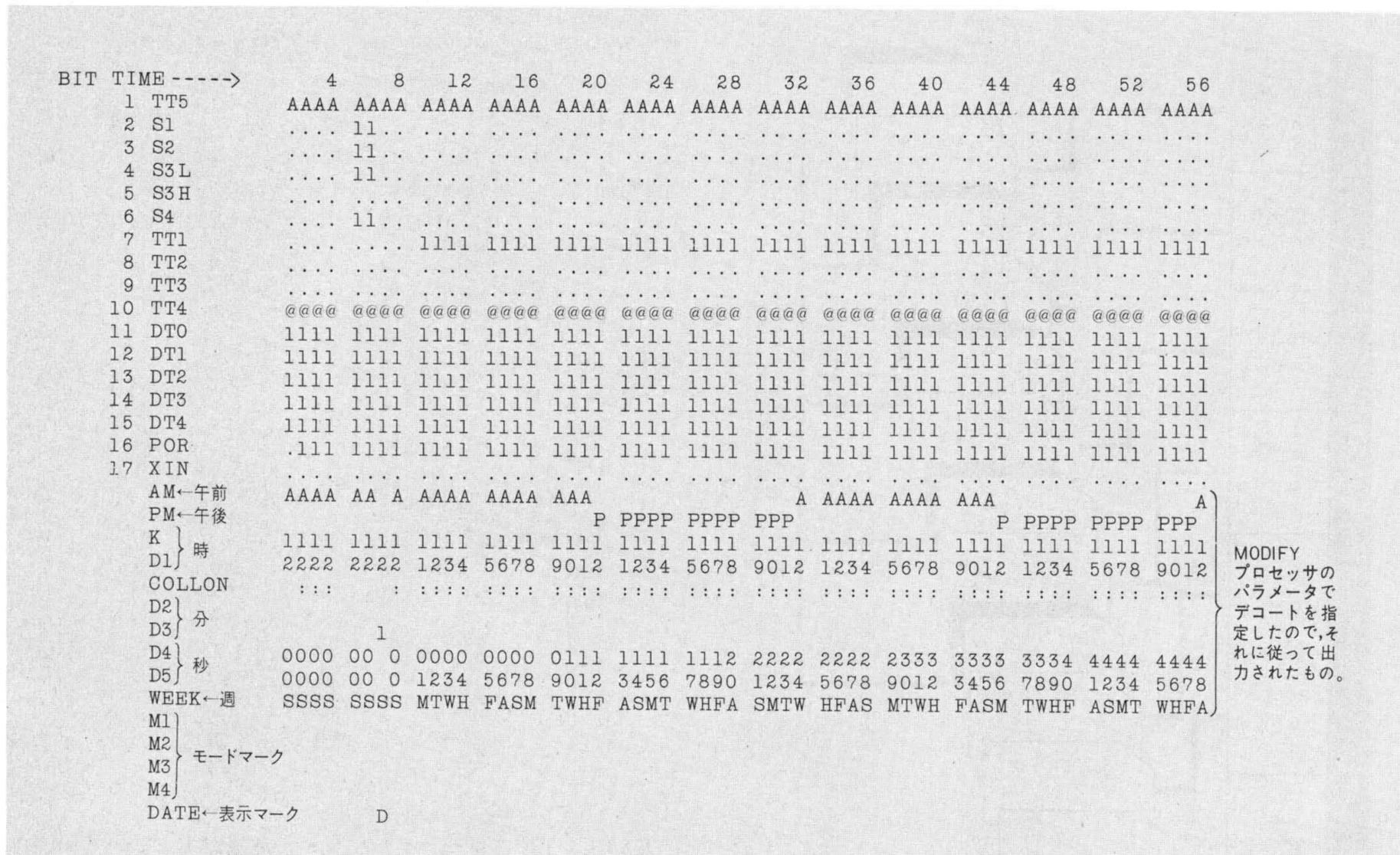


図4 シミュレーション結果の出力例 MODIFYプロセッサのパラメータでデコード指示がなされたので、それによってタイムチャートのビットを解析して表示したものである。

結果、その素子の出力値の変化数を得ることにより、ある素子の検出率とする。

図4は、タイムチャート形式の出力例を示す。

4.2.3 テストパターン情報記述言語変換プログラム

本プログラムは、入力端子に対する状態変化情報を論理シミュレータに指示する入力データの作成の簡略化をサポートするものである。本プログラムでは、簡略形式で作成されたテストパターンデータと論理回路網接続情報(CIRCUITプロセッサにより与えられる。)を入力し、シミュレート系列の全状態に変換するものである。

本プログラムの特色は次に述べるとおりである。

- (1) CLOCKはA~Zの26種の指定ができる。
- (2) REPEATの指定ができる。

REPEATとは、テストパターン系列の前のテストパターンの繰返しを、n回指定できるものである。

4.2.4 シミュレーションプログラム

本プログラムは指定された論理回路情報とテストパターン情報に基づいてシミュレーションの実行をするものである。「シミュレーション時間の短縮を実現することが、最も重要な課題である。」と前に説明したが、電子計算機での処理時間については、表2に示すように目的を達成したものと考えている。同表の処理時間は、次の条件によるものである。

- (1) 電子計算機は、HITAC M170システムを使用した。
- (2) 実行時間は単独処理の時間である。

4.3 対話形トレースシステム

4.3.1 概要

LSIの設計では、論理シミュレーション結果の検証、回路情報及びテストパターンの修正、変更などの諸作業の効率化

と小規模回路のテストなど、小回りの利くシステム運用を行なえることが必要である。

そこで、LSIの論理シミュレーションシステムのトータル化を図る観点から、対話形システムに要求される機能として、次に述べる機能をサポートしている。

(1) トレースシステム

本システムの機能は任意のビットタイムをCRTディスプレイ又はプリンタに出力する機能である。

(2) エディットシステム

本システムは回路網の変更や、テストパターンの1ビットの変更をCRTディスプレイから行なえる機能である。

(3) マージシステム

本システムの機能は、回路網とテストパターンの組合せを、CRTディスプレイから自由に行なえることをサポートするものである。

(4) テストシステム

表2 シミュレーション時間例 実行時間は、HITAC M170システムでの単独処理の場合の時間である。

ゲート数	パターン数	実行時間	
		チェックなし	チェックあり
2,000	3,036,000(5,520×550)	7分24秒	8分18秒
2,100	7,392,000(12,320×600)	15分31秒	—
2,600	1,996,800(4,160×480)	4分36秒	—
2,900	5,920,000(5,920×1,000)	10分15秒	16分12秒

本システムは、小規模回路の手入力及びテストパターンの手入力をサポートするものである。

(5) 詳細トレースシステム

本システムの機能は、任意の1ビットタイムの全システムタイムのトレースをサポートする機能である。

以上の対話形システムは、HITAC M170システムのオペレーティングシステムであるVOS2 (Virtual Storage Operating System 2) のタイムシェアリングシステムの管理下にあり、(2)のエディットシステム、(3)のマージシステム、(4)のテストシステムなどの操作はタイムシェアリングシステムの操作モードに準じている。

テストパターンの1ビット変更はタイムシェアリングシステムのCALLにより起動されるアプリケーションプログラムによって行なう。

4.3.2 トレースシステム

論理シミュレータの実行後、EXECUTEプロセッサのパラメータで指定した内容に基づいて、出力結果がプリントアウトされる。この出力データの確認で何か異常があった場合、その原因追求のためタイムチャートをトレースすることになる。そのトレース作業の効率化を実現するために、HITAC M170システムに接続されたCRTディスプレイを利用して、対話的に指定ビットタイム(又はシステムタイム)単位に全素子のタイムチャートをCRTに表示させ、異常タイムチャートの発生原因追求の手助けを行なうのがトレースシステムである。

図5にトレースシステムの検索例を図示する。その操作手順の概要は次に述べるとおりである。

- (1) ビットタイム又はシステムタイムのいずれかをCRTから入力する。
- (2) トレースしたいゲート名をCRTから入力する。
- (3) 参照したいビットタイム又はシステムタイムをCRTから入力する。

(4) 以上の操作により、シミュレーションで実行されたロギングファイルから該当するデータを検索して、CRT上に表示する(指定タイムを中心に前後10タイム)。

(5) CRTに表示されたデータに対してCRTのライトペンで?マークをピックアップする。ピックアップされたゲートに関するデータがCRT上に表示される。これらの操作を繰返し行なう。

これらのトレースシステムのサポートにより従来手法(プリントアウトされたタイミングチャートから異常箇所の発見)よりも格段に使いやすくなった。従来手法では、設計担当者に女子の設計援助者を付けて、タイミングチャートの検索を行っていたが、本システムのサポートにより約3割の工数削減ができた。

5 対話形回路設計システム

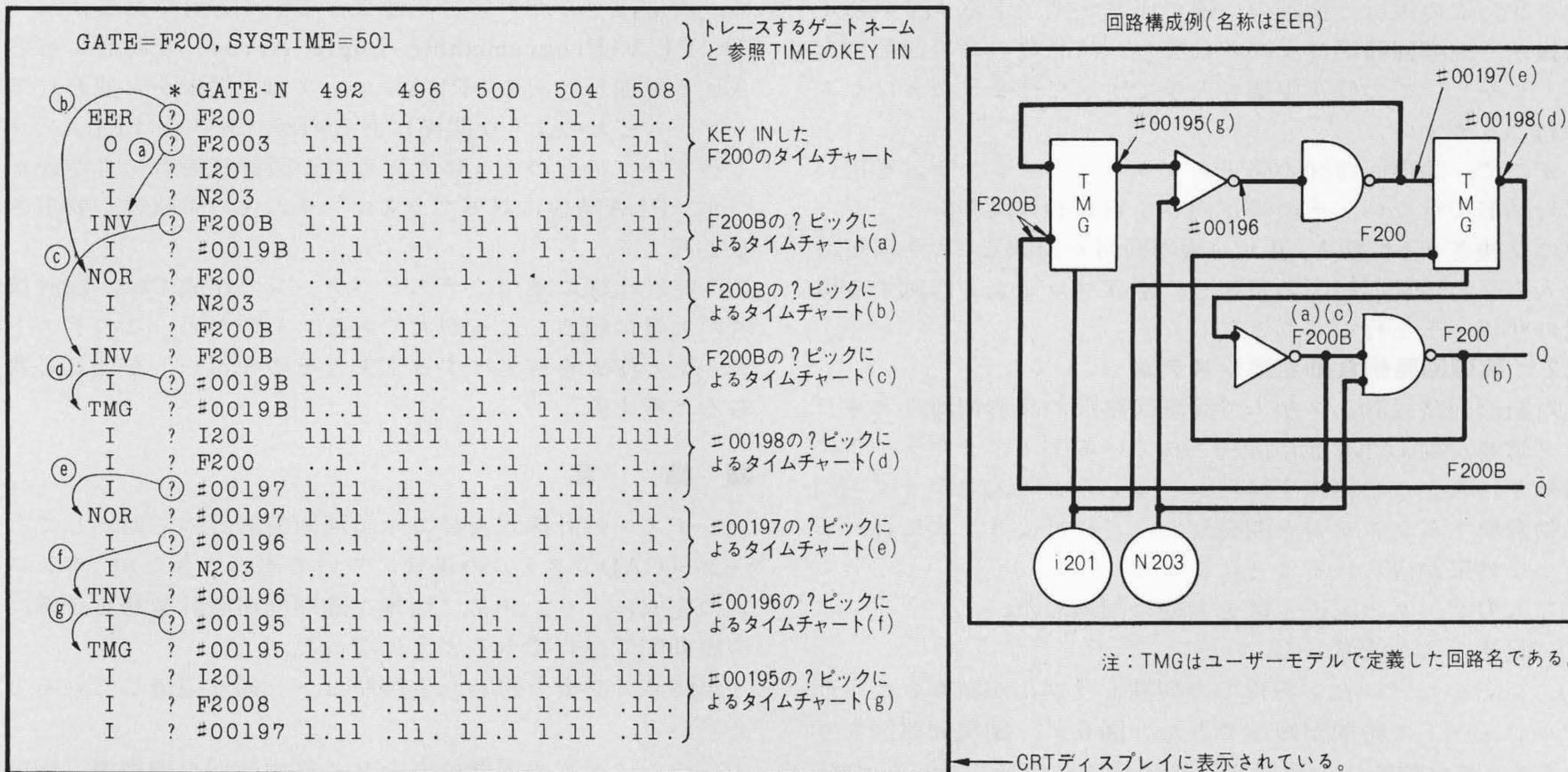
5.1 開発の背景

(1) RAM及びROMのサポート

デジタル時計にRAM及びROMを搭載するLSIが検討されている。これに対する手法としては、RAM及びROMを機能ブロックとしてシステム内の組込関数として定義する方法と、RAM及びROMもゲートとして展開して記述する方法とが基本的に考えられる。論理シミュレータとしては、ゲートレベルで記述されてもシミュレーション実行時間に対して影響を与えないことが判明したため、シチズン時計株式会社では後者のゲートとして展開する方法を採用した。

(2) 設計工数の削減

LSI回路設計工数は、回路網の複雑さから相当時間を要する。設計者は、機能ごとの回路図(以下、ブロック図という)を作成すると、ペーパー上にブロック図のレイアウトを行ないながら回路図を完成させている。そこで、既作成のブロック図については電子計算機に登録しておき、必要の都度対話形式で参照できれば設計工数の削減が図れることになる。



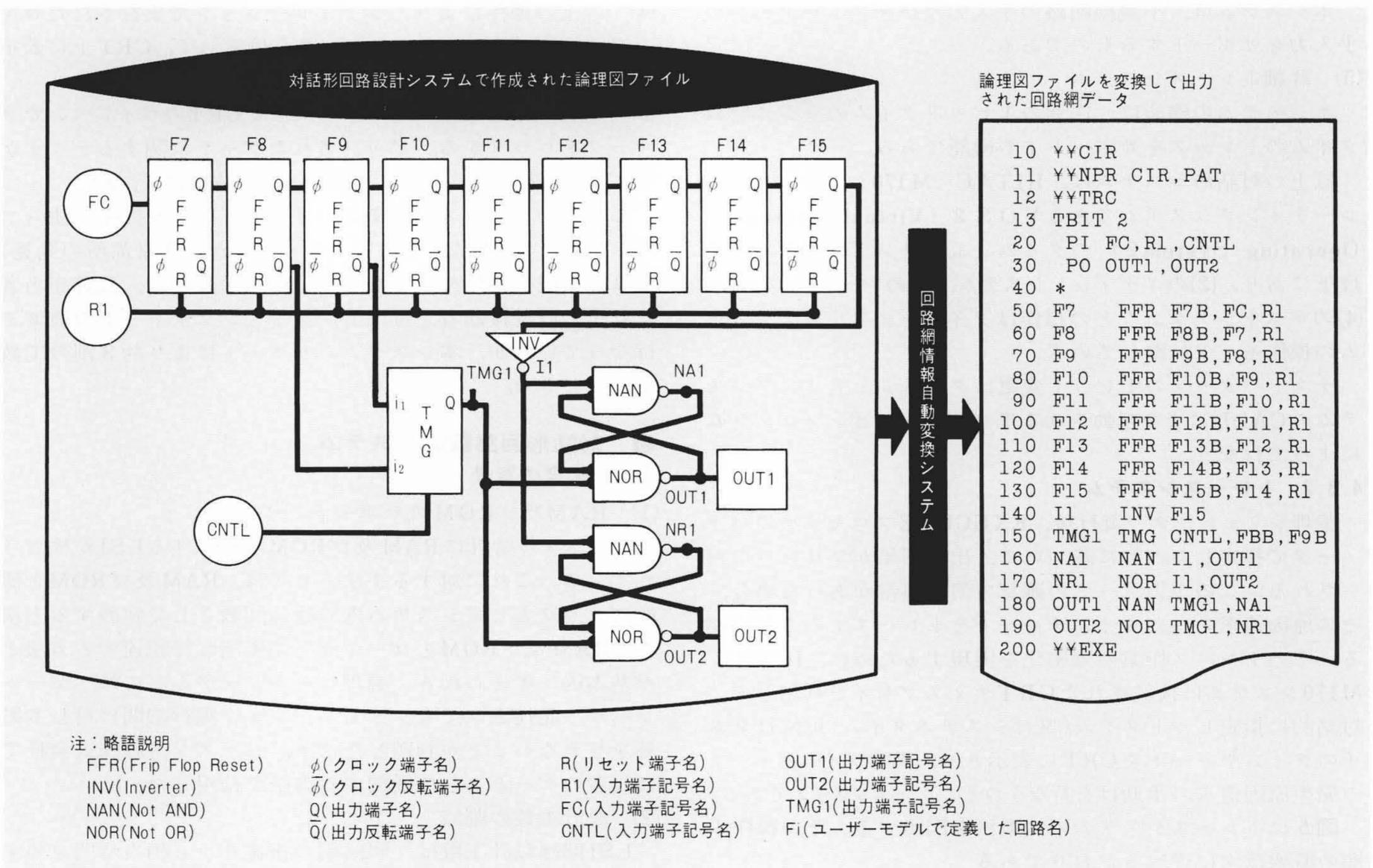


図6 回路網情報自動変換システムの例 論理ファイルの内容は、図形データとしてディスクファイルに格納されている。

(3) 図面品質の向上

上記(1), (2)と更に次のことをサポートするために、日立グラフィックシステムHITAC G710の導入を行なった。

論理回路図の情報に基づいて、論理シミュレーションの実行と結果の検証を行ないながら、正しい論理回路図の設計を行なう。この実行と検証の一連の作業が完了するまでの修正情報が、論理回路図に反映されていけば品質の高い図面になる。しかし、この修正作業を人手ですべて行なうことはミスが起こりやすい。

そこで、論理回路図の修正をグラフィックシステムを用いて対話形で行ない、その修正内容を自動的に論理シミュレータに反映させる機能と、正しい論理回路が完成したときに対話で入力した論理回路図の情報を、X-Yプロッタで作画する機能のサポートを行なうことにした。

5.2 論理回路網自動変換システム

対話形回路設計システムで論理回路図の図形情報とキャラクタ情報が電子計算機に接続されているディスクファイルに格納される。この情報を論理シミュレータの入力データ用に自動変換するシステムを開発した。これにより、次に述べるような効果が得られることになった。

- (1) 入力データの記述工数を大幅に削減した。
- (2) 記述ミスが低減した。
- (3) シミュレーション実行での初期トラブルが減少し、トラブルシュートの時間が短縮できた。図6に、論理回路図を本システムで自動的に論理回路網接続情報に変換した例を示す。

6 今後の課題

LSI設計でのトータルCADシステムを開発した。本システ

ムが対象としているのは、基本的にはスタティックLSI用のシステムである。

しかし、マトリックス駆動回路及びダイナミック駆動回路にも適用できる機能を備えている。

本システムを更に使いやすく、かつ適用領域を広めるために、次に述べる点の検討を行なっていく方針である。

- (1) PLA(Programmable Logic Array)が時計にも応用される見通しにある。PLAについては、従来から使われているゲートとどのような関係にあるのか、あるいはPLAのテストパターンはどのように考えるのか設計技術者にも定説がないが、PLA時代に対処できるシステム化の柔軟性が要求されると考える。
- (2) 時計仕様に基づくテストパターンの作成では、設計技術者の力量に頼るところが大である。そのため、このテストパターンの作成を容易にする工夫とそのサポートを図る必要があると考える。

7 結 言

シチズン時計株式会社での論理回路設計の一助としてのトータルCADシステムの現状について述べてきたが、本システムの完成により、非常に煩雑な論理回路設計業務を正確、かつ短期間で遂行できるようになった。

現時点で必要と思われる機能は、一通り完備していると考えている。

このシステムの開発に当たり、終始適切な御指導、御協力をいただいた関係各位、及び特に対話形回路設計システムの開発について御協力を得た日立製作所日立研究所の関係各位に対し感謝する次第である。