

1E-7

# データフローマシン "TOPSTAR-II" によるコンカレント故障シミュレーション

中田恒夫, 田中英彦, 元岡 達  
(東京大学 工学部)

## §1 はじめに

コンカレント故障シミュレーションは、タイミングに関する故障を扱えること、多値論理にも容易に対応できることなど優れた特徴を持つ方式であるが、1パスですべての故障を処理するために結果を得るまでに相当時間がかかるという難点を持っている。

今回、アロシージャレベル・データフローマシンTOPSTAR-II<sup>(\*)</sup>に実装を行なったコンカレント故障シミュレータは、処理を並列化することにより、実行速度の向上をねらったものである。

## §2 シミュレータの実現法

コンカレント故障シミュレーションをTOPSTAR-II上で実行するには、問題をデータフローグラフで表現する必要がある。この際に様々なレベルでの表現法が考えられるが、今回採用した方式は、回路の各論理要素をひとつのノードに割り当て、ノード内で行なわれるプロシージャとして、その論理要素における1クロック分の処理を与えるというものである。このようにすることにより、回路からデータフローグラフへの変換が容易に行なえる。

なお、回路全体に対する入力と出力は、それぞれまとめておくと入力データを変えたい場合や結果をコンソールに出力させたい場合に便利なので、別に入力ノード、出力ノードをひとつずつ設けている。

## §3 シミュレータの構成

シミュレータは以下に示す3つのプログラムにより構成されている。

### (1) システム・プログラム

データ駆動によるデータフロー処理を実現するもので、論理シミュレーション<sup>(\*\*)</sup>用に開発されたものをほとんどそのまま用いている。

### (2) 制御テーブル生成コンパイラ

テーブル記述言語で書かれている、プロシージャとノードの対応関係およびノード間の接続情報を、システム・プログラムで用いられる制御テーブルに変換するものである。

### (3) コンカレント故障シミュレーション・プログラム

各ノード内での処理を行なうものである。ユーザは端子数、論理関数等を記述したものを、本プログラムとリンクさせることにより実際に実行されるプロシージャを作り出す。

なお、主にTOPSTAR-IIのメモリ容量の制限から、シミュレータの機能に以下のような制約を加えている。

- ・論理値は、(0, 1)の2値とする。
- ・対象回路を同期回路に限定し、カフクロック部の故障はないものとする。
- ・タイミング故障は扱わない。
- ・単一縮退故障のみを対象とする。
- ・論理要素の入出力間の遅延は最低でも1クロック分はあるものとする。

## §4 シミュレーション例

対象回路として図1のような10進同期カウンタを用いシミュレーションを行なった。この回路をデータフローグラフに変換したものが、図2である。

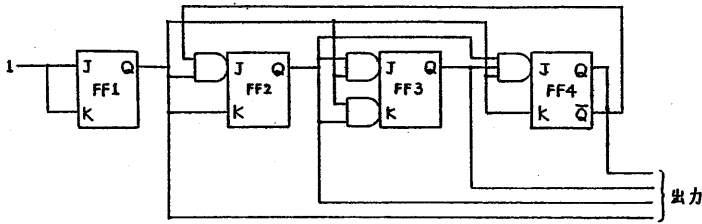


図1 対象回路 — 10進同期カウンタ

### §5 評価・検討

TOPSTAR-IIでは、Pモジュールが実際の処理（今回の場合はシミュレーション）を行なうので、Pモジュールの台数が並列動作台数にほぼ等しいと考えて良い。§4の例についてPモジュール数を1台から6台まで変化させて実行速度を測定した。その結果を図3に示す。

コンカレント故障シミュレーションの場合、問題自体の並列度はノード数に等しい。ただし今回の例では、入力ノード、出力ノードの処理時間が他に比べて非常に小さいため実質的な並列度は約4である。従ってPモジュール台数が4に近くなるとCモジュール側で実行可能なノードが不足してPモジュールの待ち時間が大きくなるために実行速度が飽和してしまっている。Pモジュール数が更に多くなると無駄な割り込みの回数が増加して本当に必要な割り込みを阻害するようになり、速度はむしろ低下している。

対象回路の規模を大きくし、ノード数をふやすと、実行可能なノードが十分あるため飽和する点が台数の多い方へずれていくと考えられる。またシミュレータの機能を拡張した場合は、Pモジュールの負荷が重いので相対的にオーバヘッドが小さくなって飽和の度合も小さくなると思われる。

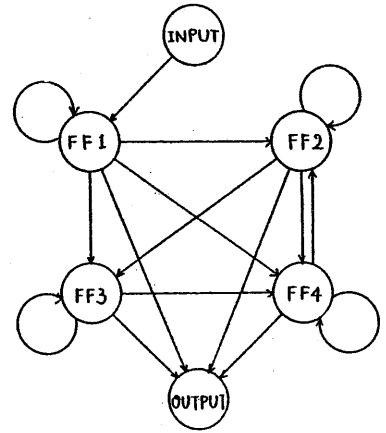


図2 対象回路のデータフローグラフ

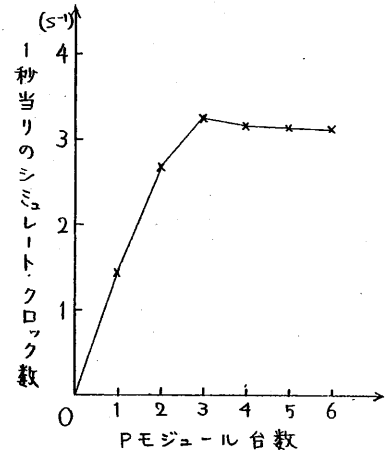


図3 Pモジュール数と実行速度

### §6 結論

今回の例では回路が小さすぎて目立った効果は得られなかったが、より実用的な回路に対しては、データフローマシンの特徴を生かし高速にコンカレント故障シミュレーションを行なえる可能性があることを示せた。

ただし、大きな回路では部分結合の影響など別の問題が現れると思われる。この点を調べることについては今後の課題としたい。

### [参考文献]

\* 元岡他「High Level Data Flow Machine (TOPSTAR) のシステム・プログラム」信学技報 EC 79-56, 1980

\*\* 元岡他「データフローマシン"TOPSTAR"による論理シミュレーション」

昭和55年度情報処理学会全国大会資料集 pp.89-90