

非同期式制御回路における等時分岐の制約の緩和

中神 賢士[†] 桑子 雅史[‡] 持木 幸一[‡]
 武蔵工業大学 大学院工学研究科 電気工学専攻[†] 武蔵工業大学 知識工学部 情報科学科[‡]

1 はじめに

現在、主流な同期式回路において生じている消費電力増大などの問題を解決する一手法として非同期式設計がある [1].

非同期式回路では、論理設計において遅延仮定を設定することが重要であるが、その一つに QDI(Quasi Delay Insensitive) モデルがある. QDI モデルは、「素子及び配線の遅延時間の上限値は有限ではあるが未知である」「配線に分岐がある場合、その分岐した先の伝搬遅延時間はすべて等しい (等時分岐の仮定)」とするモデルである. このモデルは等時分岐の仮定を設けているため LSI への実装が困難である.

しかし、QDI モデルに基づく回路であっても、実際の回路ではすべての分岐が厳密に等時である必要があるとは限らない. そこで本研究では、STG(Signal Transition Graph) を元に設計された QDI モデルに基づく非同期式制御回路の中で必ずしも等時でなくてもよい分岐を発見する手法を提案する.

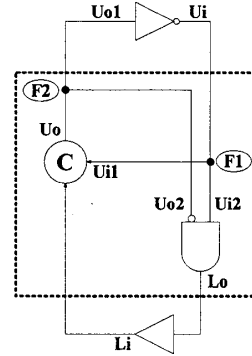


図 1: ASM(Auto Sweeping Module)

2 等時分岐制約を緩和する従来手法

STG から設計された QDI モデルに基づく制御回路の等時分岐の制約を緩和する従来手法として Nattha らの手法 [2] がある.

STG とは、制御信号の因果関係を記述した有向グラフであり、ある信号 x の 0 から 1 への遷移は x^+ と表記し、1 から 0 への遷移は x^- と表記する. そして回路が現在どの状態であるかを表すにはトークン '●' を用いる.

Nattha らの手法では、まず STG から E-STG(Extended-STG) を作成する. E-STG とは、回路を構成する論理素子の入出力すべての遷移に着目した STG である. そして、E-STG を用いて回路にハザードをもたらす可能性のある後継を持たない遷移を発見し、後継を持たない遷移と他の遷移間のレースが静的ハザードまたは動的ハザードを起こすかどうかの検証を行う. 検証を行う際には、環境側 (制御回路外) への分岐配線の遅延は、制御回路内のそれに比べて相対的に大きいと定める. 検証の結果ハザードを起こすならば、その分岐配線を含む分岐は等時分岐の制約を満たす必要があると判断し、ハザードを起こさなければ等時分岐が緩和できると判定している.

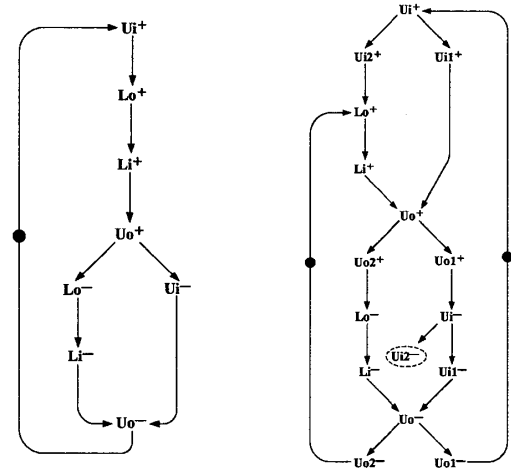


図 2: 図 1 の設計仕様の STG

図 3: 図 1 の E-STG

3 従来手法の問題点

Nattha らの手法では、実際には等時である必要のある分岐を、等時でなくてもよいと稀に判断する可能性がある. その一例を ASM(Auto Sweeping Module)[3](図 1) とその動作仕様である図 2 を用いて説明する. ここでは、図 1 に示した回路全体が制御回路内であると考えられるものとする. 図 1 中の◎は C 素子と呼ばれる. これは、入力信号すべてが 0 になった時に出力が 0, 入力信号すべてが 1 になった時に出力が 1, その他の入力値の組み合わせにおいては前の値を保持するという素子である.

従来手法では、まず図 1 と図 2 から図 3 を作成する. E-STG において出力枝を持たない遷移 U_{i2}^- (図 3 において点線円で囲まれた遷移) を後継を持たない遷移であると判定する. そし

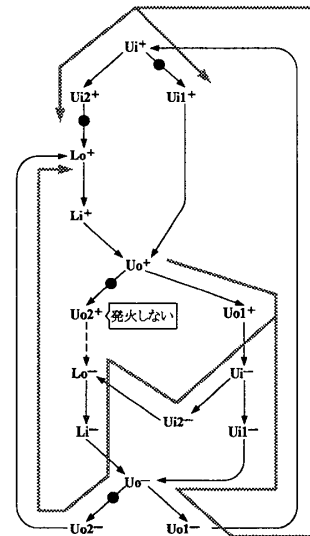


図 4: U_{o2}^+ の発生が遅れた場合の動作

Moderation of constraints of isochronic forks in asynchronous control circuits

[†] Kenji NAKAGAMI

[‡] Masashi KUWAKO

[‡] Kouichi MOCHIKI

[†] Graduate School of Research Division in Engineering, Musashi Institute of Technology

[‡] Faculty of Knowledge Engineering, Musashi Institute of Technology

て、 $Ui2^-$ と他の遷移間のレースがハザードを起こすかどうかの検証を行う。

ここで分岐配線 $Uo1$ と $Uo2$ に着目する。従来手法では $Uo1^+$, $Uo2^+$, $Uo1^-$, $Uo2^-$ のすべてが出力枝を持つので、これらは後継を持たない遷移とは見なされず、ハザードを起こすかどうかの検証対象とはならない。そして、 $Uo1$ と $Uo2$ は等時である必要はない ($Uo1$ と $Uo2$ のどちらの遅延が大きくても正しく動作する) と判定される。

しかし、分岐配線 $Uo2$ に相対的に大きな遅延が生じて、 $Uo^+ \rightarrow Uo2^+$ の枝にトークンが乗った状態のまま $Uo2^+$ は発火しない、すなわち $Uo2$ は 0 のままであるという状況を考える。動作が進行し、 $Uo1^+ \rightarrow Ui^- \rightarrow Ui2^-$ となると論理素子 Lo の入力は $Ui2=0$, $Uo2=0$ であり、 Lo の出力は 1 から 0 に遷移し、 $Lo^- \rightarrow Li^-$ と動作が進行する。一方、 $Ui^- \rightarrow Ui1^-$ と動作が進行すると、 Uo^- が発火し $Uo^- \rightarrow Uo1^-$ と $Uo^- \rightarrow Uo2^-$ の枝にトークンが乗る。 $Uo1^- \rightarrow Ui^+ \rightarrow Ui2^+$ と動作が進行し、 $Ui2^+ \rightarrow Lo^+$ の枝にトークンが乗った状態となる (図 4)。

図 4 の状態では $Uo2^-$ は発火する前の状態であるが、 $Uo2^+$ も発火する前であるため $Uo2=0$ である。 $Uo2=0$ であるため Lo^+ が発火できる。 Lo^+ の発火後に $Uo^+ \rightarrow Uo2^+$ 上のトークンによって発生が遅れていた $Uo2^+$ が発火して、更にその後に $Uo^- \rightarrow Uo2^-$ 上のトークンによって $Uo2^-$ が発火すると、 Lo が瞬時的に 0 に落ちる動的ハザードが生じる。これは当然仕様を満たさない動作であり、従来手法では等時である必要はないと判定される $Uo1$ と $Uo2$ は、実際には $Uo1$ よりも $Uo2$ の遅延が大きいと誤動作を起こす危険があることがわかる。

4 等時分岐の制約を緩和する提案手法

従来手法では、E-STG において出力枝を持たない遷移を後継を持たない遷移であるのみならず、しかし、この定義では、実際の回路動作を表現するには不十分な点がある。この定義による「後継を持たない遷移」は、実際の回路において後継を持つ場合があり、反対に「後継を持つ遷移」が実際には後継を持たない場合も考えられるためである。前節で説明したように、例えば図 1 において $Uo2^+$ の発火が遅れた場合、 $Ui2^-$ によっても Lo^- は発火することができる。すなわち図 3 においては $Uo2^+$ は後継を持つように見え、 $Ui2^-$ は後継を持たないように見えるが、実際には $Uo2^+$ は後継を持たず、 $Ui2^-$ が後継を持つような動作を行う場合が考えられる。

図 2 の STG や図 3 の E-STG は回路中の各分岐が等時であることを前提に描かれている。しかし、後継を持たない遷移がどれであるかを判別するために使用する E-STG は回路中の各分岐が等時でなかった場合の動作まで表現したものでなければならぬ。そこで $Ui2^-$ から Lo^- への因果関係を追加する。ただし、この因果関係は $Uo2^+$ または $Ui2^-$ が発火すれば Lo^- が発火する性質のものなので、STG におけるブレース表現を用いて図 5 のような E-STG を作成する。そしてブレースの直前にある遷移 ($Ui2^-$, $Uo2^+$) は後継を持たない遷移であるのみならず、このように等時分岐の仮定をなくしたと仮定した場合に顕在化する因果関係を追加した E-STG を「厳密な E-STG」と呼ぶことにする。

STG から設計された制御回路中で等時分岐の制約を満たさなくてもよい箇所を発見する提案手法は以下のようになる。

まず、設計仕様の STG と回路から厳密な E-STG を作成する。簡単のために 2 入力素子を例に説明する。後継を持たない遷移 $x^{t1}(t1: \{+, -\})$ に着目する。信号 x が入力されている論理素子を G とする。 G の出力信号を z とする。 G が AND または OR 素子である場合 $t2 = t1$ である z^{t2} を考える。 G が NAND または NOR 素子である場合 $t2 = \bar{t1}$ ($\bar{t1}$ は $t1$ の逆方向の遷移) である z^{t2} を考える。 x^{t1} から z^{t2} への因果関係を追加することにより厳密な E-STG となる。

次に、後継を持たない遷移 $x^t(t: \{+, -\})$ に着目する。その遷移が遅いと仮定して動作検証を行う (後継を持つ遷移は遅くてもよいと動作検証をしなくても判定できる)。着目した後継を持

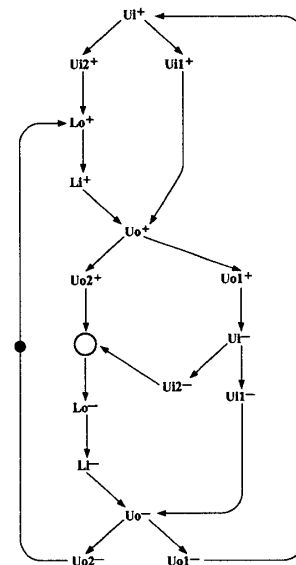


図 5: 図 1 の厳密な E-STG

たない遷移に対して並行動作を行うことができるパスを見つけ、その最上流にトークンを置く。その後、1 つでも動かすことのできるトークンがある間、すなわちデッドロックしない間、もしくは x^t を終点とする枝にトークンが置かれなかった間トークンを動かす。その結果、デッドロックした場合には遅くてもよいと判定する。 x^t を終点とする枝にトークンが置かれた場合には x^t が遅くてもよいと判定する。この処理をすべての後継を持たない遷移に対して行う。各遷移に関して得られた結果から、 x^+ と x^- のどちらも遅くてもよいと判定された場合には、分岐配線 x の遅延は大きいてもよい箇所であると決定される。それ以外の判定が得られた場合には、 x の遅延は大きいてもよい箇所であると決定される。

5 まとめ

STG を元に設計された QDI モデルに基づく非同期制御回路において、必ずしも等時でなくてもよい分岐を発見する一手法を提案した。本手法においては、従来手法において用いられる E-STG を拡張した厳密な E-STG を用いることにより、後継を持たない遷移を正確に判別して、ある分岐が等時でなかった場合にハザードを生じるかどうかの判定を誤りなく行う。

本研究では、同一信号の '+', '-' 遷移が各一つずつ含まれる STG を対象とした。同一信号の '+', '-' 遷移が複数個含まれるような STG に対応した手法の提案は今後の課題である。

参考文献

- [1] 南谷 崇 「非同期式マイクロプロセッサの動向」、情報処理, Vol.39, No.3, pp.181~186, 情報処理学会 (1998)
- [2] Nattha Sretasereekul, Takashi Nanya, "Eliminating Isochronic-Fork constraints in Quasi-Delay-Insensitive Circuits", IEICE Trans.Fundamentals, Vol.E86-A, No.4, pp.900-907 (2003)
- [3] 籠谷 裕人, 南谷 崇 「2 相式非同期回路高速化のための基本制御モジュールとその応用」、情報処理学会研究報告, Vol.1993, No.94, pp.163~170, 情報処理学会 (1993)