

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2005-293372

(P2005-293372A)

(43) 公開日 平成17年10月20日(2005.10.20)

(51) Int. Cl.⁷

G06F 17/50

G06F 13/42

F I

G06F 17/50

658K

G06F 13/42

350A

テーマコード(参考)

5B046

5B077

審査請求 未請求 請求項の数 4 O L (全 15 頁)

(21) 出願番号 特願2004-109259 (P2004-109259)
 (22) 出願日 平成16年4月1日(2004.4.1)

(71) 出願人 000006013
 三菱電機株式会社
 東京都千代田区丸の内二丁目2番3号
 (74) 代理人 100089118
 弁理士 酒井 宏明
 (72) 発明者 山元 浩幸
 東京都千代田区丸の内二丁目2番3号 三
 菱電機株式会社内
 Fターム(参考) 5B046 AA08 BA03
 5B077 GG07 GG34 MM01 MM02

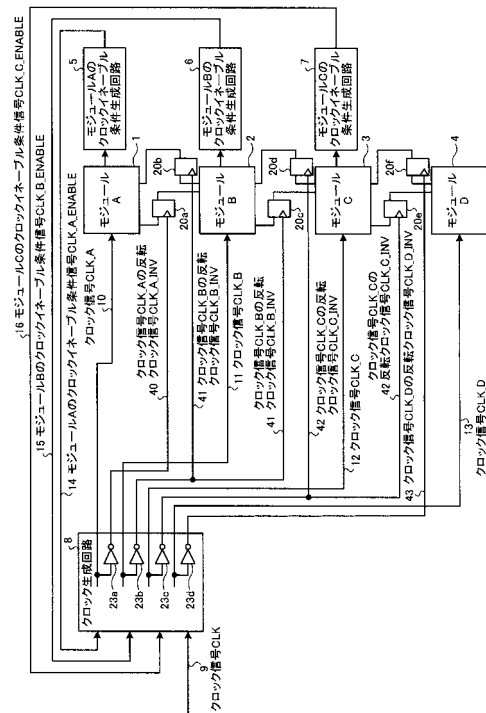
(54) 【発明の名称】 低消費電力回路の設計方法および低消費電力回路の設計支援装置

(57) 【要約】

【課題】 クロックスキューを考慮する必要が無く、かつ半導体デバイスの種類に関係なく論理回路の低消費電力回路化設計が行えるようにすること。

【解決手段】 半導体集積回路の論理回路を、データを取り込むためのイネーブル条件が付いているフリップフロップが含まれるモジュールA、B、Cと、データを取り込むためのイネーブル条件が付かないフリップフロップが含まれるモジュールDとに分ける。そしてモジュールA、B、Cには前記イネーブル条件に従ってデータを取り込むタイミングでのみクロック供給となる供給・停止付きのクロック信号が供給されるようにし、モジュールDには常にクロック供給となるクロック信号が供給されるようにする。その後、前記分割したモジュール間のデータ転送経路に、データ送信側のモジュールに供給する前記クロック信号を反転したクロック信号でデータを取り込むフリップフロップを挿入する。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

半導体集積回路の論理回路におけるフリップフロップを、データを取り込むためのイネーブル条件が付いている第 1 フリップフロップのグループと前記データを取り込むためのイネーブル条件が付かない第 2 フリップフロップのグループとに分ける工程と、

前記第 1 フリップフロップのグループを複数の第 1 モジュールに分割し、前記第 2 フリップフロップのグループを 1 以上の第 2 モジュールに分割する工程と、

分割したモジュールを単位として、前記第 1 モジュールでは、そのモジュールに含まれる第 1 フリップフロップに対し、前記イネーブル条件に従ってデータを取り込むタイミングではクロック供給となりその他のタイミングではクロック停止となる供給・停止付きのクロック信号が供給されるようにし、前記第 2 モジュールでは、そのモジュールに含まれる第 2 フリップフロップに対し、常にクロック供給となりデータの取り込みを連続して行わせるクロック信号が供給されるようにする工程と、

前記分割したモジュール間のデータ転送経路に、データ送信側のモジュールに供給する前記クロック信号を反転したクロック信号でデータを取り込むフリップフロップを挿入する工程と、

を含むことを特徴とする低消費電力回路の設計方法。

【請求項 2】

前記モジュールに分割する工程では、少なくとも、前記第 1 フリップフロップのグループを複数の第 1 モジュールに分割する場合に、当該第 1 モジュール間でのデータ転送に関わる信号数が少なくなるように前記第 1 フリップフロップのグループを分割する工程を含むことを特徴とする請求項 1 に記載の低消費電力回路の設計方法。

【請求項 3】

半導体集積回路の論理回路を表現した論理記述データと前記半導体集積回路に含まれるクロック信号の総数が含まれる制約条件とに基づき、前記クロック信号に接続されるフリップフロップが、データを取り込むためのイネーブル条件付きであるか否かを判別する手段と、

前記判別手段の判別結果に基づき、イネーブル条件付きのフリップフロップをグループ化してモジュールに分割し、イネーブル条件付きでないフリップフロップをグループ化して別のモジュールに分割する手段と、

前記制約条件として与えた半導体集積回路に含まれるクロック信号数を満足するように前記分割したイネーブル条件付きのフリップフロップが含まれるモジュールを細かく分割する手段と、

システムの異なるクロック信号が供給されるモジュール間のデータ転送経路に、送信側のモジュールに供給するクロック信号を反転したクロック信号で駆動されるフリップフロップを挿入する手段と、

前記イネーブル条件付きのフリップフロップが含まれる各モジュールについてそのモジュール内のフリップフロップがデータを取り込むタイミングのみでクロック信号が供給されるようにするクロックイネーブル条件を生成する回路を生成する手段と、

前記イネーブル条件付きのフリップフロップが含まれる各モジュールに対しては各モジュールに対する前記クロックイネーブル条件に基づき、そのモジュール内のフリップフロップがデータを取り込むタイミングではクロック供給となりその他のタイミングではクロック供給停止となる供給・停止付きのクロック信号を生成する一方、前記イネーブル条件付きでないフリップフロップが含まれる各モジュールに対してはそのモジュール内のフリップフロップがデータの取り込みを連続して行うように常にクロック供給となるクロック信号を供給する回路を生成する手段と、

前記の各手段によって生成された論理記述データを設計結果として出力する手段と、

を備えたことを特徴とする低消費電力回路の設計支援装置。

【請求項 4】

半導体集積回路の論理回路を表現した論理記述データと前記半導体集積回路に含まれる

10

20

30

40

50

クロック信号の総数が含まれる制約条件とに基づき、前記クロック信号に接続されるフリップフロップが、データを取り込むためのイネーブル条件付きであるか否かを判別する手段と、

前記判別手段の判別結果に基づき、イネーブル条件付きのフリップフロップをグループ化してモジュールに分割し、イネーブル条件付きでないフリップフロップをグループ化して別のモジュールに分割する手段と、

少なくとも、前記イネーブル条件付きのフリップフロップ間でのデータ転送経路を解析し、その経路総数を調べる手段と、

前記経路総数に基づき、モジュール分割後のモジュール間でのデータ転送を行う信号数が少なくなるように、前記分割したイネーブル条件付きのフリップフロップが含まれるモジュールを再分割する手段と、 10

系統の異なるクロック信号が供給されるモジュール間のデータ転送経路に、送信側のモジュールに供給するクロック信号を反転したクロック信号で駆動されるフリップフロップを挿入する手段と、

前記イネーブル条件付きのフリップフロップが含まれる各モジュールについてそのモジュール内のフリップフロップがデータを取り込むタイミングのみでクロック信号が供給されるようにするクロックイネーブル条件を生成する回路を生成する手段と、

前記イネーブル条件付きのフリップフロップが含まれる各モジュールに対しては各モジュールに対する前記クロックイネーブル条件に基づき、そのモジュール内のフリップフロップがデータを取り込むタイミングではクロック供給となりその他のタイミングではクロック供給停止となる供給・停止付きのクロック信号を生成する一方、前記イネーブル条件付きでないフリップフロップが含まれる各モジュールに対してはそのモジュール内のフリップフロップがデータの取り込みを連続して行うように常にクロック供給となるクロック信号を供給する回路を生成する手段と、 20

前記の各手段によって生成された論理記述データを設計結果として出力する手段と、
を備えたことを特徴とする低消費電力回路の設計支援装置。

【発明の詳細な説明】

【技術分野】

【0001】

この発明は、半導体集積回路の設計において所定のタイミングで動作する論理回路を低消費電力回路化する低消費電力回路の設計方法および低消費電力回路化の設計を支援する低消費電力回路の設計支援装置に関するものである。 30

【背景技術】

【0002】

携帯機器等、省電力化の要求の厳しい分野の半導体集積回路においては、ゲートドクロックを用いた設計が行われている。しかしながら、ゲートドクロック回路を用いることにより、クロックスキューが大きくなり回路が誤動作する可能性がある。

【0003】

この課題に対して従来から種々の提案がなされている（例えば、特許文献1～3）。すなわち、特許文献1では、同一のクロックドメインに含まれるフリップフロップの数をほぼ同一にし、できるだけフリップフロップ転送条件が同一のフリップフロップの数が多くなるようにクロックツリーを形成する技術が開示されている。この特許文献1に記載の技術では、同一のクロックドメインに含まれるフリップフロップの転送条件の和集合をクロックのイネーブル条件とする。このようなクロックツリーを形成することにより、クロックドメイン間のクロックスキューを小さくすることができ、消費電力を低減することができる。 40

【0004】

また、特許文献2では、半導体集積回路の階層ブロック毎にクロックツリー合成や配置配線を行った後の論理合成、タイミング解析、およびクロックツリー合成の効率および精 50

度を向上させることを目的に、クロックで動作する第1クロック同期セルを有する階層ブロックと、上記クロックと同じクロックで動作し、階層ブロックに含まれない複数の第2クロック同期セルにおいて、複数の第2クロック同期セルのクロック入力端子と、このクロック入力端子にクロック信号を供給するクロック配線との間にこのクロック配線を入力とする第1のクロックバッファを挿入し、これによりクロック配線を、階層ブロックに接続する第1クロック配線と、第2クロック同期セルに接続する第2クロック配線とに分割する技術が開示されている。

【0005】

また、特許文献3では、複数のクロックモードを具備する半導体集積回路の設計において、複数のクロックモードにおけるクロックスキューの最適化を効率的に行うことを目的に、半導体集積回路のレイアウトデータから、各クロックモードにおける複数のクロックパスを検出し、各クロックモードにおける複数のクロックパスを検出し、検出したクロックパス上に存在するすべての素子のディレイ量を収集し、検出した各クロックパス上におけるディレイ調整位置を設定し、設定したディレイ調整位置におけるディレイ量を非負数の変数とし、この変数と収集した素子のディレイ量とからクロックパスごとに線形式を作成して演算することにより、各パス上のディレイ調整位置における最適なディレイ値を算出し、算出したディレイ値に基づき、各ディレイ調整位置に、該当するディレイ量を有する遅延素子を配置して、レイアウトデータによる回路構成を自動的に修正する技術が開示されている。

10

【0006】

【特許文献1】特開2001-84287号公報

【特許文献2】特開2003-92354号公報

【特許文献3】特開2003-271689号公報

【発明の開示】

【発明が解決しようとする課題】

【0007】

しかしながら、特許文献1に記載の技術では、各クロックドメインに接続するフリップフロップ数を均等にするにより、各クロックドメインに接続するフリップフロップ間のクロックスキューが低減するという特徴はあるが、半導体集積回路の配置配線が考慮されていない。つまり、クロックスキュー値がチップ内でどの程度になるかは、配置配線後のクロック配線の引き回し経路やデバイスの製造バラツキ等があるので、正確に見積もることは困難であり、クロックスキューが原因でデータ転送が正しく行われなことが有り得る。

20

30

【0008】

また、特許文献2に記載の技術では、複数のクロックを有する回路に対して、階層的なレイアウトでは有効なクロックツリー合成手法であるが、クロックドメインの異なるフリップフロップ間のクロックスキュー調整のためにクロックツリーを合成する他に、実際には、クロックドメインの異なる複数のフリップフロップの間に、クロックスキュー調整用のバッファを挿入することも考えられるので、レイアウト後の正確なタイミング解析ができない場合には、回路が誤動作する可能性がある。また、FPGA(Field Programmable Gate Array)のようにクロック配線が予めなされているデバイスでは、クロックツリーを合成する本手法を適用することができないという問題もある。

40

【0009】

また、特許文献3に記載の技術では、複数のクロックモードにおいて、レイアウトデータから各クロックパスのディレイ量を算出し、各クロックパスの最適な位置に最適なディレイを挿入するので、レイアウト後の対策のために精度の高いディレイをクロックパスに挿入することができる。しかし、特許文献2に記載の技術と同様に、予めクロック配線がなされているFPGAにおいては、上記の手法を適用することができないという問題がある。

【0010】

50

今日の半導体集積回路、特に、セルベースLSI (Large Scale Integrated Circuit) 等の設計においては、半導体集積回路の機能検証のために、FPGA等の再構成可能なデバイスを用いて高速に機能を検証することが行われているので、様々なデバイスに適用できる低消費電力設計方式の開発が望まれている。

【0011】

この発明は、上記に鑑みてなされたものであり、クロックスキューを考慮する必要が無く、かつ半導体デバイスの種類に関係なく論理回路の低消費電力回路化に適用できる低消費電力回路の設計方法を得ることを目的とする。

【0012】

また、この発明は、クロックスキューを考慮する必要が無く、かつ半導体デバイスの種類に関係なく論理回路の低消費電力回路化に適用できる低消費電力回路をコンピュータの支援によって自動的に生成する低消費電力回路の設計支援装置を得ることを目的とする。

【課題を解決するための手段】

【0013】

上述した目的を達成するために、この発明にかかる低消費電力回路の設計方法は、半導体集積回路の論理回路におけるフリップフロップを、データを取り込むためのイネーブル条件が付いている第1フリップフロップのグループと前記データを取り込むためのイネーブル条件が付かない第2フリップフロップのグループとに分ける工程と、前記第1フリップフロップのグループを複数の第1モジュールに分割し、前記第2フリップフロップのグループを1以上の第2モジュールに分割する工程と、分割したモジュールを単位として、前記第1モジュールでは、そのモジュールに含まれる第1フリップフロップに対し、前記イネーブル条件に従ってデータを取り込むタイミングではクロック供給となりその他のタイミングではクロック停止となる供給・停止付きのクロック信号が供給されるようにし、前記第2モジュールでは、そのモジュールに含まれる第2フリップフロップに対し、常にクロック供給となりデータの取り込みを連続して行わせるクロック信号が供給されるようにする工程と、前記分割したモジュール間のデータ転送経路に、データ送信側のモジュールに供給する前記クロック信号を反転したクロック信号でデータを取り込むフリップフロップを挿入する工程とを含むことを特徴とする。

【0014】

この発明によれば、モジュール間のデータ転送は、データ送信側のモジュールに供給するクロック信号を反転したクロック信号でデータを取り込むフリップフロップを介して行うようにしたので、クロックスキューの影響無しにモジュール間のデータ転送が行えるようになる。このとき、データを取り込むためのイネーブル条件が付いている第1フリップフロップが含まれるモジュール間のデータ転送では、当該モジュール間に挿入されたフリップフロップは、供給・停止付きのクロック信号の反転クロック信号によって動作するので、低消費電力化も実現することができる。

【発明の効果】

【0015】

この発明によれば、クロックスキューの影響が無い低消費電力回路を半導体デバイスの種類に依存しないで設計することができるという効果を奏する。

【発明を実施するための最良の形態】

【0016】

以下に図面を参照して、この発明にかかる低消費電力回路の設計方法および低消費電力回路の設計支援装置の好適な実施の形態を詳細に説明する。

【0017】

実施の形態1.

図1は、この発明にかかる低消費電力回路の設計方法の一実施の形態による低消費電力回路の設計方法を説明する図である。図1において、モジュールA1, モジュールB2, モジュールC3およびモジュールD4は、それぞれ論理回路の回路要素であるフリップフロップを、供給するクロック信号によってグループ化したモジュールであり、次のように

して分割され、クロックの供給が行われる。なお、少なくともモジュール A 1 , モジュール B 2 およびモジュール C 3 は、モジュール間のデータ転送に関わる信号数が少なくなるように分割されている。

【 0 0 1 8 】

すなわち、半導体集積回路の論理回路に存するフリップフロップを、まず、データを取り込むためのイネーブル条件付きのフリップフロップ（以降「データイネーブル条件付きのフリップフロップ」と記す）と、常にクロック信号が供給されそのクロック信号に同期してデータを常に取り込み続けるフリップフロップ（以降「フリーランクロック信号が供給されるフリップフロップ」と記す）とに分ける。

【 0 0 1 9 】

次に、データイネーブル条件付きのフリップフロップを複数のモジュールに分割（図 1 ではモジュール A 1 , モジュール B 2 , モジュール C 2 の 3 つを示す）し、それぞれのモジュールにおけるフリップフロップの全てがデータを取り込むタイミングのみでクロックを供給するためのクロックイネーブル条件（クロックイネーブル条件信号 C L K _ A _ E N A B L E 1 4 , C L K _ B _ E N A B L E 1 5 , C L K _ C _ E N A B L E 1 6 ）をそれぞれ生成する回路（クロックイネーブル条件生成回路 5 , 6 , 7 ）を設計する。

10

【 0 0 2 0 】

そして、それらのクロックイネーブル条件（クロックイネーブル条件信号 C L K _ A _ E N A B L E 1 4 , C L K _ B _ E N A B L E 1 5 , C L K _ C _ E N A B L E 1 6 ）と基準クロック信号 9 とに基づき、データを取り込むタイミングのみでクロック供給となりその他のタイミングではクロック供給の停止となる供給・停止付きのクロック信号（クロック信号 C L K _ A 1 0 , C L K _ B 1 1 , C L K _ B 1 2 ）を対応するモジュール（モジュール A 1 , モジュール B 2 , モジュール C 2 ）に分配供給するクロック生成回路 8 を設計する。

20

【 0 0 2 1 】

また、フリーランクロック信号が供給されるフリップフロップも複数のモジュールに分割（図 1 ではモジュール D 4 の 1 個を示す）し、クロック生成回路 8 はフリーランクロック信号（クロック信号 C L K _ D 1 3 ）を基準クロック信号 9 に基づき供給するように設計する。

【 0 0 2 2 】

次に、互いに異なるクロック信号が入力されるモジュール間のデータ転送部分に、データ送信側モジュールのクロック信号を反転したクロック信号で動作するフリップフロップを挿入する。すなわち、図 1 に示すように、クロック生成回路 8 に、クロック信号 C L K _ A 1 0 を反転するインバータ 2 3 a と、クロック信号 C L K _ B 1 1 を反転するインバータ 2 3 b と、クロック信号 C L K _ C 1 2 を反転するインバータ 2 3 c と、クロック信号 C L K _ D 1 3 を反転するインバータ 2 3 d とをそれぞれ設ける。

30

【 0 0 2 3 】

そして、モジュール A 1 からモジュール B 2 へのデータ転送部分に、インバータ 2 3 a が出力する反転クロック信号 C L K _ A _ I N V 4 0 で駆動されるフリップフロップ 2 0 a を挿入する。モジュール B 2 からモジュール A 1 へのデータ転送部分とモジュール B 2 からモジュール C 3 へのデータ転送部分とに、インバータ 2 3 b が出力する反転クロック信号 C L K _ B _ I N V 4 1 で駆動されるフリップフロップ 2 b , 2 c をそれぞれ挿入する。同様にモジュール C 3 からモジュール B 2 へのデータ転送部分とモジュール C 3 からモジュール D 4 へのデータ転送部分とに、インバータ 2 3 c が出力する反転クロック信号 C L K _ C _ I N V 4 2 で駆動されるフリップフロップ 2 d , 2 e をそれぞれ挿入する。また、モジュール D 4 からモジュール C 3 へのデータ転送部分に、インバータ 2 3 d が出力する反転クロック信号 C L K _ D _ I N V 4 3 で駆動されるフリップフロップ 2 f に挿入する。

40

【 0 0 2 4 】

この設計方法によれば、モジュール間のデータ転送は、データ送信側のモジュールに供

50

給するクロック信号を反転したクロック信号でデータを取り込むフリップフロップを介して行うようにしたので、クロックスキューの影響無しにモジュール間のデータ転送が行えるようになる。しかも、データイネーブル付きフリップフロップに関わるモジュール間のデータ転送に用いる反転したクロック信号は、データ転送に必要なタイミングのみでクロック信号が供給されるので、低消費電力化も実現することができる。このような低消費電力回路は、半導体デバイスの種類を問わずに設計することができる。また、クロックスキュー等のタイミングの問題が発生しないので、確実に動作する。

【0025】

次に、図2は、図1に示すクロック生成回路8の構成例を示す回路図である。図2において、フリップフロップ18a, 18b, 18cの各データ入力端Dには、各モジュールのクロックイネーブル条件信号CLK__A__ENABLE14, CLK__B__ENABLE15, CLK__C__ENABLE16が印加されている。また、フリップフロップ18a, 18b, 18cの各クロック入力端Gには、基準クロック信号CLK9をインバータ19にて反転して印加されている。これによって、フリップフロップ18a, 18b, 18cの各データ出力端Qには、クロックマスク信号CLK__A__MASK29, CLK__B__MASK30, CLK__C__MASK31がそれぞれ出力される。

10

【0026】

AND回路22a, 22b, 22cでは、一方の入力端に基準クロック信号CLK9が共通に印加されている。AND回路22aでは、他方の入力端にクロックマスク信号CLK__A__MASK29が印加され、出力端にバッファ24aとインバータ23aとが並列に接続されている。バッファ24aからはモジュールA1に対するクロック信号CLK__A10が出力される。インバータ23aからはデータ転送用のフリップフロップ20aに対する反転クロック信号CLK__A__INV40が出力される。

20

【0027】

AND回路22bでは、他方の入力端にクロックマスク信号CLK__B__MASK30が印加され、出力端にバッファ24bとインバータ23bとが並列に接続されている。バッファ24bからはモジュールB2に対するクロック信号CLK__B11が出力される。インバータ23bからはデータ転送用のフリップフロップ20b, 20cに対する反転クロック信号CLK__B__INV41が出力される。

30

【0028】

AND回路22cでは、他方の入力端にクロックマスク信号CLK__C__MASK31が印加され、出力端にバッファ24cとインバータ23cとが並列に接続されている。バッファ24cからはモジュールC3に対するクロック信号CLK__C12が出力される。インバータ23cからはデータ転送用のフリップフロップ20d, 20eに対する反転クロック信号CLK__C__INV42が出力される。

【0029】

また、基準クロック信号CLK9が並列に入力されるバッファ24dとインバータ23dと設けられている。バッファ24dからはモジュールD4に対するフリーランクロック信号CLK__D13が出力される。インバータ23dからはデータ転送用のフリップフロップ20fに対する反転クロック信号CLK__D__INV43が出力される。

40

【0030】

なお、図2に示すように、各モジュールに対するクロックイネーブル条件信号CLK__A__ENABLE14, CLK__B__ENABLE15, CLK__C__ENABLE16を、それぞれ基準クロック信号CLK9の反転クロック信号によってフリップフロップ18a, 18b, 18cに一旦ラッチし、それと基準クロック信号CLK9とのANDを取るようにしたので、モジュール等にはグリッジが乗らないクロック信号が供給されることになる。

【0031】

次に、図3～図6を参照して、クロックスキューの影響無しにモジュール間のデータ転送を実現する方法を具体的に説明する。なお、図3は、クロック系統の異なるモジュール

50

間のデータ転送において、クロックスキューの影響によってデータ転送が正しく行われな
い可能性のある回路例である。図4は、図3に示す回路においてモジュールA1からモジ
ュールB2へのデータ転送動作を説明するタイムチャートである。図5は、図3に示す回
路において、モジュールA1とモジュールB2との間のデータ転送部分に、転送元のクロ
ック信号の反転クロック信号で駆動されるフリップフロップを挿入した回路例である。図
6は、図5に示す回路においてモジュールA1からモジュールB2へのデータ転送動作を
説明するタイムチャートである。

【0032】

図3では、説明を簡略化するために、2つのモジュールのモジュールA1とモジュール
B2には、それぞれ、1個のフリップフロップ35, 36が存在し、モジュールA1から
モジュールB2に対して1本だけデータ転送経路がある場合の構成が示されている。モジ
ュールA1からモジュールB3へのデータ転送は、複数のバッファを直列に接続したバッ
ファ37を介して行われる。したがって、クロック生成回路8は、図2に示した構成とは
異なり、モジュールA1へのクロック信号CLK_A10とモジュールB2へのクロック
信号CLK_B11のみを生成する構成となる。

10

【0033】

すなわち、図3において、モジュールA1では、2入力1出力のマルチプレクサ21a
と1つのフリップフロップ35とが示されている。マルチプレクサ21aは、一方の入力
端Aにフリップフロップ35のデータ出力端Qが接続され、他方の入力端に外部からデー
タ信号DATA_IN50が印加される。また、マルチプレクサ21aは、セット入力端
Sに外部からデータイネーブル信号DATA_A_ENABLE52が印加され、出力端
Yはフリップフロップ35のデータ入力端Dに接続されている。データイネーブル信号D
ATA_A_ENABLE52は、クロックイネーブル条件生成回路5にも入力されている。
フリップフロップ35では、クロック入力端Gにクロック信号CLK_A10が印加
される。そして、フリップフロップ35のデータ出力端Qに現れる出力データ信号DATA
_OUT51は、マルチプレクサ21aの一方の入力端Aに印加されるとともに、バッ
ファ37を介したモジュールB2への転送データとなる。

20

【0034】

モジュールB2では、2入力1出力のマルチプレクサ21bと1つのフリップフロップ
36とが示されている。マルチプレクサ21bは、一方の入力端Aにフリップフロップ3
6のデータ出力端Qが接続され、他方の入力端にバッファ37からのデータ信号DATA
_IN53が印加される。また、マルチプレクサ21bは、セット入力端Sに外部からデ
ータイネーブル信号DATA_B_ENABLE55が印加され、出力端Yはフリップフ
ロップ36のデータ入力端Dに接続されている。データイネーブル信号DATA_B_E
NABLE55は、クロックイネーブル条件生成回路6にも入力されている。フリップフ
ロップ36では、クロック入力端Gにクロック信号CLK_B11が印加される。そして
、フリップフロップ35のデータ出力端Qに現れる出力データ信号DATA_OUT54
は、マルチプレクサ21bの一方の入力端Aに印加される。

30

【0035】

ここで、モジュールA1に供給するクロック信号CLK_A10とモジュールB2に供
給するクロック信号CLK_A11とには、図4に示すように、クロックスキュー60が
存在する。図4では、基準クロックCLKのクロックサイクルT1, T2, T3, T4,
T5における、モジュールA1内のフリップフロップ35の入出力関係(クロック信号C
LK_A、データイネーブル信号DATA_A_ENABLE、入力データDATA_A
_IN、出力データDATA_A_OUT)と、モジュールB2内のフリップフロップ3
6の入出力関係(クロック信号CLK_B、データイネーブル信号DATA_B_E
NABLE、入力データDATA_B_IN、出力データDATA_B_OUT)とが示され
ている。

40

【0036】

図4において、モジュールA1内のフリップフロップ35に供給するクロック信号CL

50

K__Aは、クロックタイミングT1, T2とクロックタイミングT5以降では、低レベル(クロック供給停止)であり、クロックタイミングT3, T4での各前半サイクルにおいて高レベル(クロック供給)である。これに対し、モジュールB2内のフリップフロップ36に供給するクロック信号CLK__Bは、クロックの供給・停止の関係は、クロック信号CLK__Aと同じであるが、クロック信号CLK__Bの先頭供給クロックの位置は、クロックタイミングT4の前半サイクルの期間内においてクロックタイミングT4の開始時点からクロックスキュー60の分だけ位相が遅れた位置にシフトされている。

【0037】

モジュールA1内のフリップフロップ35では、データネーブル信号DATA__A__ENABLEが、クロックタイミングT2, T3の期間内高レベルとなっている。入力データDATA__A__INは、クロックタイミングT2の終了時点まで高レベルであり、クロックタイミングT3の開始時点で低レベルとなり、以降それを維持している。出力データDATA__A__OUTは、クロック信号CLK__Aに忠答して生成されるので、クロックタイミングT3の期間内では高レベルとなり、クロックタイミングT4の期間内では低レベルとなる。この出力データDATA__A__OUTが、バッファ37を介してモジュールB2に転送される。

10

【0038】

モジュールB2内のフリップフロップ36では、データネーブル信号DATA__B__ENABLEは、クロックスキュー60を考慮して生成されるので、クロックタイミングT3の開始時点からクロックスキュー60の分だけ遅れた位置にて高レベルとなり、クロックタイミングT5の開始時点からクロックスキュー60の分だけ遅れた位置にて低レベルとなっている。これに対し、入力データDATA__B__INは、モジュールA1が転送する出力データDATA__A__OUTであるので、クロックタイミングT3の期間内では高レベルとなり、クロックタイミングT4の期間内では低レベルとなる。

20

【0039】

そうすると、この入力データDATA__B__INをサンプリングするクロック信号CLK__Bは、入力データDATA__B__INが低レベルを示すクロックタイミングT4の期間以降に存在するので、出力データDATA__B__OUTは、常に、低レベルのままとなる。

【0040】

この問題を解決するために、図5に示すように、バッファ37に代えて、フリップフロップ20aを挿入する。そして、クロック生成回路8から、モジュールA1へのクロック信号CLK__A10の反転クロック信号CLK__A__INV40をフリップフロップ20aに供給する構成と採る。

30

【0041】

このようにすると、図6に示すように、モジュールA1からモジュールB2への転送経路に挿入したフリップフロップ20aでは、反転クロック信号CLK__A__INV40がクロック信号CLK__Aの後半サイクルにシフトされた関係で入力するので、モジュールA1からの転送データである入力データDATA__T__INは、半サイクルだけシフトされてデータ出力端に現れるモジュールB2への出力データDATA__T__OUTとなる。

40

【0042】

その結果、モジュールB2内のフリップフロップ36では、フリップフロップ20aからの転送データである入力データDATA__A__INは、クロックタイミングT3の後半サイクルの開始時点からクロックタイミングT4の前半サイクルの終了時点までの期間内の高レベルとなる。クロック信号CLK__Bの先頭供給クロックの位置は、クロックタイミングT4の前半サイクルの期間内においてクロックタイミングT4の開始時点からクロックスキュー60の分だけ位相が遅れた位置に存するので、入力データDATA__A__INの高レベルをサンプリングすることができ、クロックタイミングT4の前半サイクルの途中からクロックタイミングT5の前半サイクルの途中まで高レベルで、クロックタイミングT5の前半サイクルの途中から図示してないがクロックタイミングT6の前半サイク

50

ルの途中まで低レベルとなるデータ出力DATA__B__OUTが得られる。

【0043】

このように、モジュールA1の出力データDATA__A__OUTを、モジュールA1に供給するクロック信号CLK__A10の反転クロック信号CLK__A__INV40によって一旦、フリップフロップ20aにラッチし、モジュールB2に転送するようにすれば、モジュールA1とモジュールB2との間にクロックスキューがあっても、確実に、データを転送することができる。しかも、データ転送部分に追加したフリップフロップ20aには、データ転送時に必要なクロックのみしか供給されないので、消費電力を抑えることができる。

【0044】

実施の形態2 .

図7は、この発明にかかる低消費電力回路の設計支援装置の一実施の形態による低消費電力回路の設計支援装置の構成を示すフローチャートである。この実施の形態2では、実施の形態1に示したクロックシステムの異なるモジュール間のデータ転送を実現する低消費電力回路の設計において、モジュール間のデータ転送数を考慮しないでモジュールの分割を行う設計支援装置の構成例が示されている。

【0045】

図7において、記憶装置100には、半導体集積回路の論理回路を、例えばHDL (Hardware Description Language) にて表現した論理記述データが格納されている。以降、記憶装置100に代えて、論理記述データ100と記す。また、記憶装置102には、当該設計対象の半導体集積回路に含まれるクロック信号の総数が含まれる制約条件が格納されている。以降、記憶装置102に代えて、制約条件102と記す。

【0046】

当該低消費電力回路の設計支援装置を制御するコンピュータ上の支援プログラムでは、まず、論理記述データ100と制約条件102を読み込み(ステップS11)、論理回路に含まれるフリップフロップの種別の解析を行う(ステップS12)。つまり、ステップS12では、論理回路に含まれるフリップフロップが、フリーランクロック信号の供給されるフリップフロップであるか、或いは、データイネーブル付きのフリップフロップであるかの判別を行う。

【0047】

そして、判別したフリップフロップの種別に従って、当該半導体集積回路をモジュールに分割する(ステップS13)。つまり、ステップS13では、フリーランクロック信号の供給されるフリップフロップをグループ化してモジュールに分割する。また、データイネーブル付きのフリップフロップをグループ化して別のモジュールに分割する。

【0048】

次いで、上記のように分割したデータイネーブル条件付きのフリップフロップが含まれるモジュールを、制約条件102として与えた半導体集積回路に含まれるクロック信号数を満足するように細かく分割する(ステップS14)。そして、システムの異なるクロック信号が供給されるモジュール間のデータ転送部分に、転送元のモジュールに供給するクロック信号を反転したクロック信号で駆動されるフリップフロップを挿入する(ステップS15)。

【0049】

次いで、データイネーブル条件付きのフリップフロップが含まれる各モジュールについて、そのモジュール内のフリップフロップがデータを転送するタイミングのみでクロック信号が供給されるようにするクロックイネーブル条件を生成するクロックイネーブル条件生成回路と、前記各モジュールに対するクロックイネーブル条件によってデータを転送するタイミングのみでクロック供給となりその他のタイミングではクロック停止となる供給・停止付きのクロック信号および常にデータを取り込むフリップフロップに供給するフリーランクロック信号を生成するクロック生成回路とを生成する(ステップS16)。

【0050】

10

20

30

40

50

そして、終了処理として、上述の手順にて処理して生成されたHDLなどによる論理記述データを記憶装置101に書き出す(ステップS17)。斯くして、記憶装置101には、この実施の形態2による低消費電力化技術適用後のHDL等で表現された論理記述データが格納される。

【0051】

以上の処理によって、異なるクロック信号が供給されるモジュール間転送においてクロックスキューによる誤動作の発生しない低消費電力化を実現した論理回路を支援プログラムにより生成することができる。

【0052】

実施の形態3.

図8は、この発明にかかる低消費電力回路の設計支援装置の他の実施の形態による低消費電力回路の設計支援装置の構成を説明するフローチャートである。この実施の形態3では、実施の形態1に示したクロックシステムの異なるモジュール間のデータ転送を実現する低消費電力回路の設計において、モジュール間のデータ転送数を考慮してモジュールの分割を行う設計支援装置の構成例が示されている。

【0053】

図8において、記憶装置100には、例えば、HDL(Hardware Description Language)にて論理回路を表現した論理記述データが格納されている。以降、記憶装置100に代えて、論理記述データ100と記す。また、記憶装置102には、出力する半導体集積回路に含まれるクロック信号の総数が含まれる制約条件が格納されている。以降、記憶装置102に代えて、制約条件102と記す。

【0054】

当該低消費電力回路の設計支援装置を制御するコンピュータ上の支援プログラムでは、まず、論理記述データ100と制約条件102を読み込み(ステップS21)、論理回路に含まれるフリップフロップの種別の解析を行う(ステップS22)。つまり、ステップS22では、論理回路に含まれるフリップフロップが、フリーランクロック信号の供給されるフリップフロップであるか、或いは、データイネーブル付きのフリップフロップであるかの判別を行う。

【0055】

そして、判別したフリップフロップの種別に従って、当該半導体集積回路をモジュールに分割する(ステップS23)。つまり、ステップS23では、フリーランクロック信号の供給されるフリップフロップをグループ化してモジュールに分割する。また、データイネーブル付きのフリップフロップをループ化して別のモジュールに分割する。

【0056】

次に、実施の形態2とは異なり、論理回路間、特に、データイネーブル条件付きのフリップフロップ間でのデータ転送経路を解析しその経路総数を調べる(ステップS24)。そして、モジュール分割後のモジュール間、特に、フリップフロップ間のデータ転送を行う信号線数が少なくなるように、ステップ23にて分割したデータイネーブル付きのフリップフロップを含むモジュールを再分割する(ステップS25)。

【0057】

以降は実施の形態2と同様である。すなわち、システムの異なるクロック信号が供給されるモジュール間のデータ転送部分に、データ送信側モジュールに供給するクロック信号を反転したクロック信号で駆動されるフリップフロップを挿入する(ステップS26)。

【0058】

次いで、データイネーブル条件付きのフリップフロップが含まれる各モジュールについて、そのモジュール内のフリップフロップがデータを転送するタイミングのみでクロック信号が供給されるようにするクロックイネーブル条件を生成するクロックイネーブル条件生成回路と、前記各モジュールに対するクロックイネーブル条件によってデータを転送するタイミングのみでクロック供給となりその他のタイミングではクロック停止となる供給・停止付きのクロック信号および常にデータを取り込むフリップフロップに供給するフリ

10

20

30

40

50

ーランクロック信号を生成するクロック生成回路とを生成する(ステップS27)。

【0059】

そして、終了処理として、上述の手順にて処理して生成されたHDLなどによる論理記述データを記憶装置101に書き出す(ステップS28)。斯くして、記憶装置101には、この実施の形態3による低消費電力化技術適用後のHDL等で表現された論理記述データが格納される。

【0060】

以上の処理によって、実施の形態2と同様に、異なるクロック信号が供給されるモジュール間転送においてクロックスキューによる誤動作の発生しない低消費電力化を実現した論理回路をプログラムにより生成することができる。加えて、モジュール間のデータ転送の数が少なくなるようにモジュール分割を行っているので、モジュール間に挿入されるフリップフロップの数が少なくなり、回路規模の増加を抑えることができるという効果が得られる。

10

【0061】

以上説明したように、この発明によれば、クロックスキューの影響の無い低消費電力回路を半導体デバイスの種類に依存しないで設計することができる。これによって、半導体集積回路のレイアウトを行った後にクロックスキューの問題が発生し、タイミングが収束しなくなった最悪の場合に、今までは製造後に論理回路の誤動作が発生していたが、そのような問題を無くすることができる。

【0062】

特に、FPGAのように、予めロジックセルに対してクロック配線が行われている半導体集積回路の場合には、チップ内で使用できるクロック信号が定められているので、この発明によれば、そのクロック信号数に合うような論理回路が作成することができるようになる。

20

【0063】

また、システムの異なるクロック間のクロックスキューを考慮しなくてよいのでレイアウト以降の工数を短縮できる。さらに、ためのイネーブル条件が付いているフリップフロップが含まれるモジュールでは、各モジュール内のフリップフロップがデータを取り込むために必要なタイミングのみにクロック信号を供給することができるようになるので、論理回路の電力消費の殆どを占めるスイッチングを削減することができ、低消費電力化を実現することができる。

30

【産業上の利用可能性】

【0064】

以上のように、この発明にかかる低消費電力回路の設計方法は、半導体集積回路の論理回路の低消費電力回路化設計を、クロックスキューを考慮する必要が無く、かかる半導体デバイスの種類に依存せずに実施するのに有用である。

【0065】

また、この発明にかかる低消費電力回路の設計支援装置は、クロックスキューを考慮する必要が無く、かつ半導体デバイスの種類に関係なく低消費電力回路化した論理回路をコンピュータ上の支援プログラムにて自動生成するのに有用である。

40

【図面の簡単な説明】

【0066】

【図1】この発明にかかる低消費電力回路の一実施の形態による低消費電力回路の設計方法を説明する図である。

【図2】図1に示すクロック生成回路8の構成例を示す回路図である。

【図3】クロックシステムの異なるモジュール間のデータ転送において、クロックスキューの影響によってデータ転送が正しく行われな可能性のある回路例を示す図である。

【図4】図3に示す回路においてモジュールAからモジュールBへのデータ転送動作を説明するタイムチャートである。

【図5】図3に示す回路において、モジュールAとモジュールBとの間のデータ転送部分

50

に、転送元のクロックの反転クロック信号で駆動されるフリップフロップを挿入した回路例を示す図である。

【図6】図5に示す回路においてモジュールAからモジュールBへのデータ転送動作を説明するタイムチャートである。

【図7】この発明にかかる低消費電力回路の設計支援装置の一実施の形態による低消費電力回路の設計支援装置の構成を説明するフローチャートである。

【図8】この発明にかかる低消費電力回路の設計支援装置の他の実施の形態による低消費電力回路の設計支援装置の構成を説明するフローチャートである。

【符号の説明】

【0067】

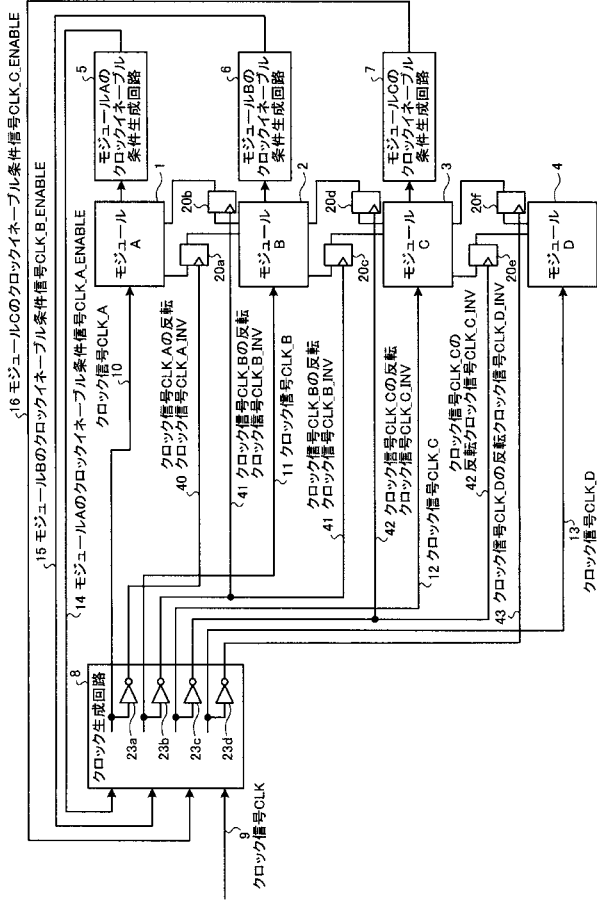
- 1 モジュールA（論理回路の回路要素を供給するクロック信号でグループ化したモジュールで、データイネーブル付きのフリップフロップを含む）、
- 2 モジュールB（論理回路の回路要素を供給するクロック信号でグループ化したモジュールで、データイネーブル付きのフリップフロップを含む）、
- 3 モジュールC（論理回路の回路要素を供給するクロック信号でグループ化したモジュール、データイネーブル付きのフリップフロップを含む）、
- 4 モジュールD（論理回路の回路要素を供給するクロック信号でグループ化したモジュールで、フリーランクロック信号によって駆動されるフリップフロップを含む）、
- 5 モジュールAのクロックイネーブル条件生成回路、
- 6 モジュールBのロックイネーブル条件生成回路、
- 7 モジュールCのクロックイネーブル条件生成回路、
- 8 クロック生成回路、
- 18a, 18b, 18c フリップフロップ、
- 20a, 20b, 20c, 20d, 20e, 20f フリップフロップ、
- 21a, 21b 2入力1出力のマルチプレクサ、
- 22a, 22b, 22c AND回路、
- 23a, 23b, 23c, 23d インバータ、
- 24a, 24b, 24c, 24d, 37 バッファ、
- 35, 36 フリップフロップ、
- 60 クロックスキュー、
- 100, 101, 102 記憶装置。

10

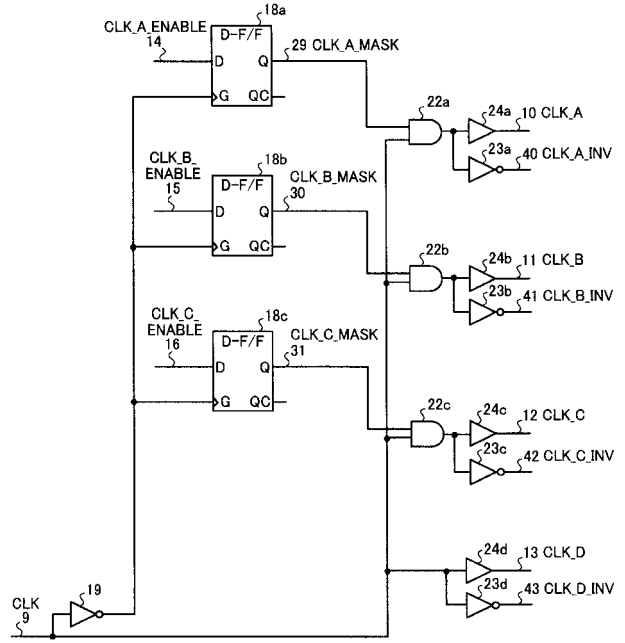
20

30

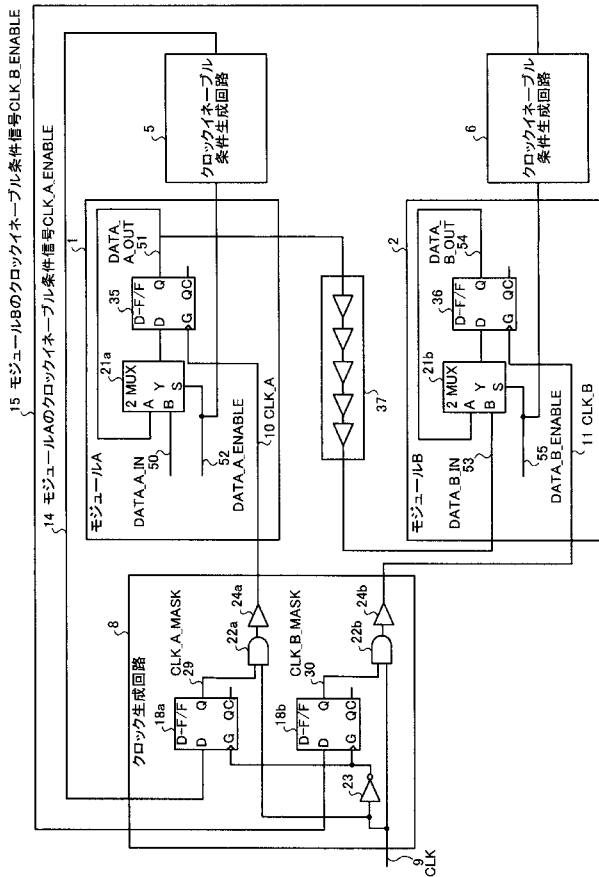
【 図 1 】



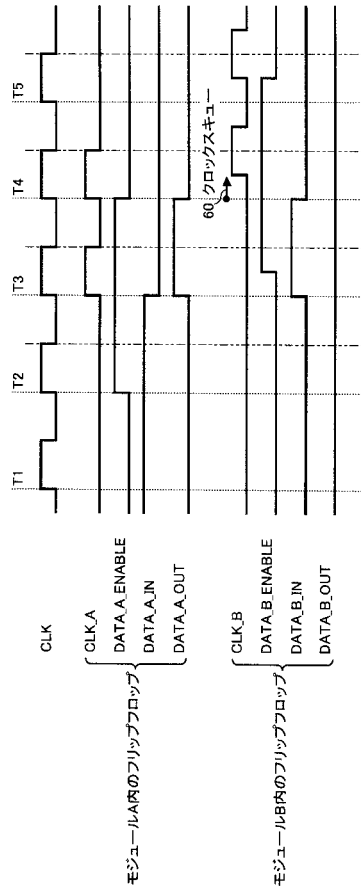
【 図 2 】



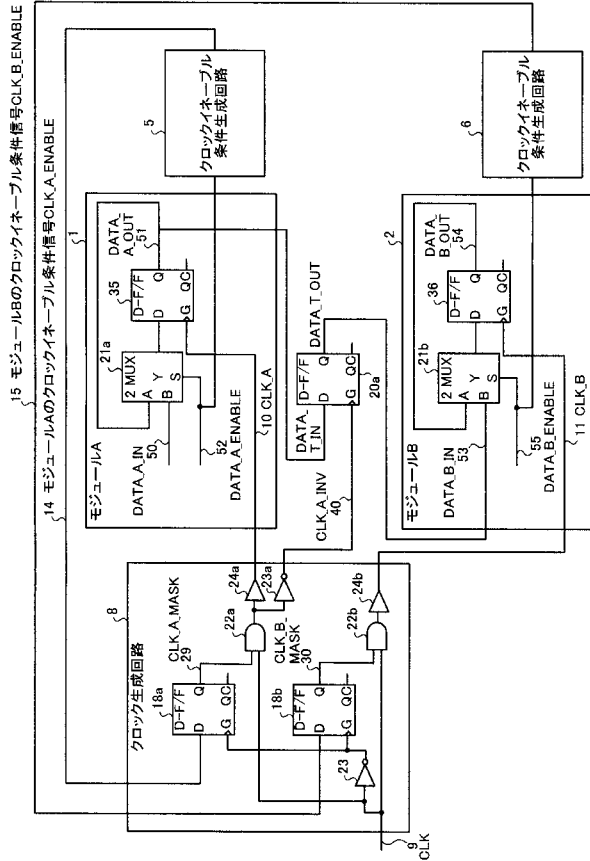
【 図 3 】



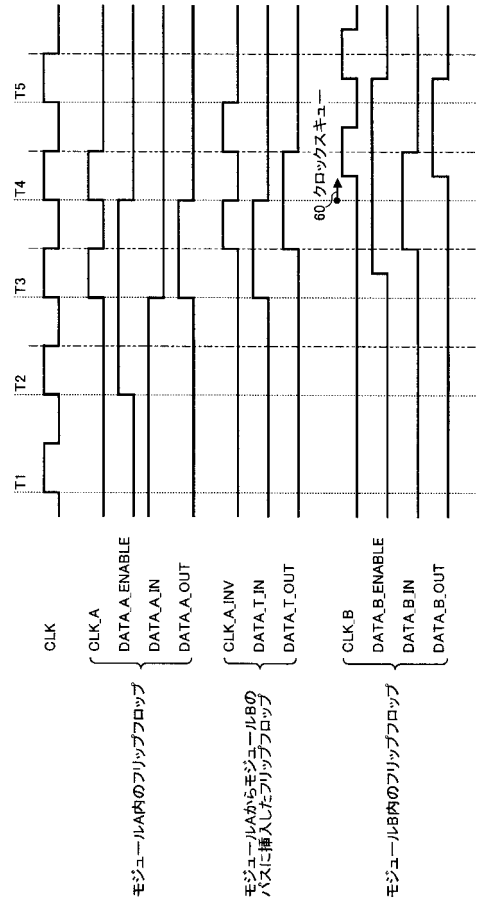
【 図 4 】



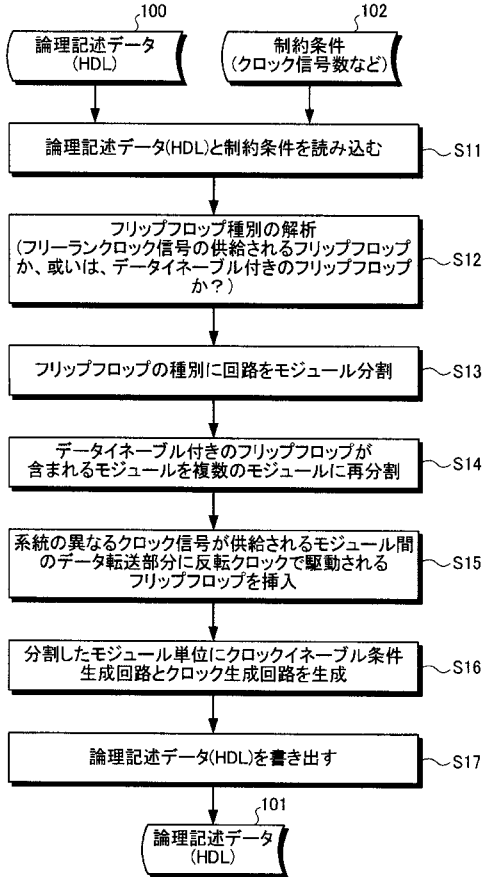
【 図 5 】



【 図 6 】



【 図 7 】



【 図 8 】

