

(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2012-243205

(P2012-243205A)

(43) 公開日 平成24年12月10日(2012.12.10)

(51) Int.Cl.	F I	テーマコード (参考)
G06F 11/14 (2006.01)	G06F 11/14 310D	5B027
G06F 11/22 (2006.01)	G06F 11/22 310E	5B048

審査請求 未請求 請求項の数 11 O L (全 14 頁)

(21) 出願番号	特願2011-114833 (P2011-114833)	(71) 出願人	302062931 ルネサスエレクトロニクス株式会社 神奈川県川崎市中原区下沼部1753番地
(22) 出願日	平成23年5月23日 (2011.5.23)	(74) 代理人	100103894 弁理士 冢入 健
		(72) 発明者	高橋 大悟 神奈川県横浜市神奈川区金港町3番地1 ルネサスマイクロシステム株式会社内
		(72) 発明者	伊藤 健一 神奈川県横浜市神奈川区金港町3番地1 ルネサスマイクロシステム株式会社内
		(72) 発明者	遠藤 誠 神奈川県横浜市神奈川区金港町3番地1 ルネサスマイクロシステム株式会社内
		Fターム(参考)	5B027 BB01 CC04 5B048 AA20 CC01

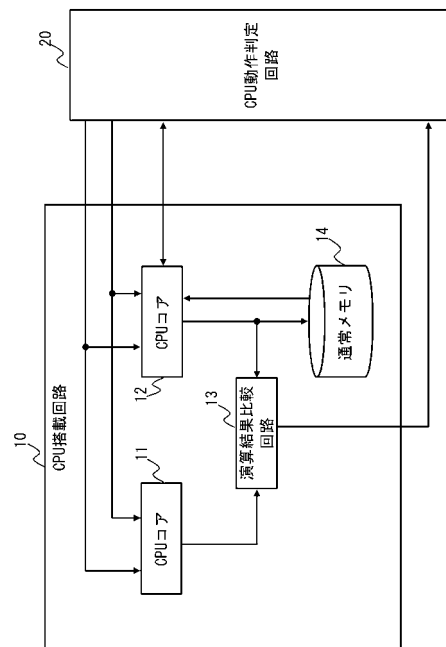
(54) 【発明の名称】 半導体集積回路及びデータ退避方法

(57) 【要約】

【課題】 正常性が確保された経路を介してデータの退避を行うことができる半導体集積回路を提供すること

【解決手段】 本発明にかかる半導体集積回路は、CPUコア12と、CPUコア12の演算結果と、第1のサンプルデータとを保持する通常メモリ14と、第1のサンプルデータと同一の第2のサンプルデータを保持するサンプルデータ格納メモリ24と、通常メモリ14から出力された第1のサンプルデータと、第1のサンプルデータが出力された経路と重複しない経路を介してサンプルデータ格納メモリ24から出力された第2のサンプルデータとを比較するデータ比較判定器26と、を備えCPUコア12は、第1のサンプルデータと第2のサンプルデータとが一致する場合に、演算結果を、第1のサンプルデータがデータ比較判定器26に出力された経路と同じ経路を介してCPUコア12から退避させるものである。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

プロセッサと、

前記プロセッサの演算結果と、第 1 のサンプルデータとを保持する第 1 のメモリと、
前記第 1 のサンプルデータと同一の第 2 のサンプルデータを保持する第 2 のメモリと、
前記第 1 のメモリから出力された前記第 1 のサンプルデータと、前記第 1 のサンプルデータが出力された経路と重複しない経路を介して前記第 2 のメモリから出力された前記第 2 のサンプルデータとを比較するデータ比較判定部と、を備え

前記プロセッサは、

前記第 1 のサンプルデータと前記第 2 のサンプルデータとが一致する場合に、前記演算結果を、前記第 1 のサンプルデータが前記データ比較判定部に出力された経路と同じ経路を介して前記第 1 のメモリから退避させる、半導体集積回路。

10

【請求項 2】

前記プロセッサと異なる比較用プロセッサと、

前記プロセッサにおける演算結果と、前記比較用プロセッサにおける演算結果とを比較する演算結果比較部と、をさらに備え、

前記データ比較判定部は、

前記演算結果比較部において、前記プロセッサにおける演算結果と、前記比較用プロセッサにおける演算結果とが異なると判定された場合に、前記第 1 のサンプルデータと、前記第 2 のサンプルデータとを比較する、請求項 1 記載の半導体集積回路。

20

【請求項 3】

前記演算結果比較部において、前記プロセッサが通常動作している場合に、前記プロセッサにおける演算結果と、前記比較用プロセッサにおける演算結果とが異なると判定された場合に、

前記プロセッサは、

前記第 1 のメモリから前記データ比較判定部に対して、前記第 1 のサンプルデータを出力させる、請求項 2 記載の半導体集積回路。

【請求項 4】

前記プロセッサの動作を切り替える動作制御部をさらに備え、

前記動作制御部は、

前記演算結果比較部において、前記プロセッサが通常動作している場合に、前記プロセッサにおける演算結果と、前記比較用プロセッサにおける演算結果とが異なると判定されると、通常動作をしている前記プロセッサを、前記第 1 のメモリから前記データ比較判定部に対して、前記第 1 のサンプルデータを出力させる経路故障判定動作に切り替える、請求項 3 記載の半導体集積回路。

30

【請求項 5】

前記動作制御部は、

前記データ比較判定部において、前記第 1 のサンプルデータと前記第 2 のサンプルデータとが一致すると判定されると、前記経路故障判定動作をしているプロセッサを、前記演算結果を前記第 1 のメモリから退避メモリへ退避させる退避動作へ切り替える、請求項 4 記載の半導体集積回路。

40

【請求項 6】

前記退避メモリは、不揮発性メモリである請求項 5 に記載の半導体集積回路。

【請求項 7】

プロセッサと、前記プロセッサの演算結果と、第 1 のサンプルデータとを保持する第 1 のメモリと、前記第 1 のサンプルデータと同一の第 2 のサンプルデータを保持する第 2 のメモリと、を備える半導体集積回路のデータ退避方法であって、

前記第 1 のメモリから出力された前記第 1 のサンプルデータと、前記第 1 のサンプルデータが出力された経路と重複しない経路を介して前記第 2 のメモリから出力された前記第 2 のサンプルデータとを比較し、

50

前記第 1 のサンプルデータと前記第 2 のサンプルデータとが一致する場合に、前記演算結果を、前記第 1 のサンプルデータが出力された経路と同じ経路を介して前記第 1 のメモリから退避させる、データ退避方法。

【請求項 8】

前記プロセッサにおける演算結果と、前記プロセッサと同一処理を実行する比較用プロセッサにおける演算結果とを比較し、

前記プロセッサにおける演算結果と、前記比較用プロセッサにおける演算結果とが異なると判定された場合に、前記第 1 のサンプルデータと、前記第 2 のサンプルデータとを比較する、請求項 7 記載のデータ退避方法。

【請求項 9】

前記プロセッサが通常動作している場合に、前記プロセッサにおける演算結果と、前記比較用プロセッサにおける演算結果とが異なると判定された場合に、

前記第 1 のメモリから前記第 1 のサンプルデータを出力させる、請求項 8 記載のデータ退避方法。

【請求項 10】

前記プロセッサが通常動作している場合に、前記プロセッサにおける演算結果と、前記比較用プロセッサにおける演算結果とが異なると判定されると、通常動作をしている前記プロセッサを、前記第 1 のメモリから前記第 1 のサンプルデータを出力させる経路故障判定動作に切り替える、請求項 9 記載のデータ退避方法。

【請求項 11】

前記第 1 のサンプルデータと前記第 2 のサンプルデータとが一致すると判定されると、前記経路故障判定動作をしているプロセッサを、前記演算結果を前記第 1 のメモリから退避メモリへ退避させる退避動作へ切り替える、請求項 10 記載のデータ退避方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は半導体集積回路及びデータ退避方法に関し、特にデータの退避経路の正常性を判定する半導体集積回路及びデータ退避方法に関する。

【背景技術】

【0002】

近年、車載製品等の半導体集積回路には高い安全性を確保することが必要なため、信頼性向上の必要性が高まっている。これにより、自己診断機能を搭載した半導体集積回路が普及してきた。2000年に機能安全規格IEC 61508（電気・電子・プログラマブル電子安全関連システムの機能安全）が発行され、欧州は、自動車に搭載するECUレベルの機能安全認証の取得を義務付けている。そのため、MCU（Micro Control Unit）は、IEC 61508 SIL3のサポートを要求されており、CPU（Central Processing Unit）の冗長化や自己診断機能の採用によるシステムの信頼性向上が必要とされている。

【0003】

上記規格を満足するための自己診断方法として、次のような方法が普及している。2つのCPUを搭載している半導体集積回路において、一方のCPUをマスタCPU、もう一方のCPUをチェック用CPUとする。このような半導体集積回路において、2つの演算結果を比較して半導体集積回路の故障有無等を診断する。

【0004】

前記の自己診断方法を搭載した半導体集積回路において、物理的な故障ではない外的要因、例えばノイズや宇宙線の影響で発生したエラーにより、それぞれのCPUの演算結果に不一致が発生した場合、半導体集積回路は、リセットされエラー発生直前からの動作を再開する故障延命手段を用いることが検討されている。

【0005】

車載製品等の半導体集積回路は、半導体集積回路を搭載した自動車の動作中に自己診断

10

20

30

40

50

による再起動が発生する場合がある。この場合、動作中のリセットから半導体集積回路の動作を再開するまでの時間を短くする必要がある。そのため、それぞれのCPUの演算結果に不一致が発生した後に、CPUは、不一致が発生する直前のデータを不揮発性メモリに退避し、動作再開時に不揮発性メモリに退避していたデータを読み出し、読み出したデータを動作状態の設定に利用する方法が検討されている。CPUの演算結果の不一致発生後にデータを不揮発性メモリに退避するためには、半導体集積回路のデータ退避に関わる部分に故障がないことを判定する必要がある。さらに、2つのCPUにおいて、故障延命手段を確実に実行できる技術が求められている。

【0006】

特許文献1を用いて、故障延命手段が用いられている半導体集積回路の構成について説明する。具体的に、図7及び8を用いて、特許文献1に開示されている半導体集積回路の構成について説明する。

10

【0007】

半導体集積回路は、CPU110からデータバス190、210、230を經由して主記憶170へデータを出力し、CPU120からデータバス200、220、240を經由して主記憶180へデータを出力する。

【0008】

通常処理では、CPU110から主記憶170へ出力されるデータは、データバス190、210、230を經由する。CPU120から主記憶180へ出力されるデータは、データバス200、220、240を經由する。通常処理においては、これらの経路が標準となる。CPU110、120は、データバスへデータを出力する毎にCPUに内蔵している退避レジスタ130、140にデータを退避する。

20

【0009】

比較器160は、処理ブロック320に示すように、CPU110からデータバス190及び210へ出力されたデータと、CPU120からデータバス200及び220へ出力されたデータを、比較器160に内蔵された退避レジスタ250及び260に出力し、データの比較を行う。

【0010】

比較器160において比較された結果、CPU110とCPU120から出力されたデータが等しい場合、処理ブロック325に示すように、処理を継続する。比較器160において比較された結果、CPU110とCPU120から出力されたデータが不一致である場合、処理ブロック330に示すように、比較器160は、CPU110、120に対して、不一致の発生を通知するために不一致検出信号270、280を出力する。処理ブロック340に示すように、CPU110、120は不一致検出信号270、280を検出すると同時に割込みを発生し、不一致検出割込みハンドラに処理を移す。不一致検出割込みハンドラは、処理ブロック350に示すように、一方のデータバスの接続を切断し、処理ブロック360に示すようにCPU110の退避レジスタ130のデータを出力する。

30

【0011】

処理ブロック370に示すように、比較器160は、比較器160の退避レジスタ250とCPU110から出力された退避レジスタ130の値を比較する。

40

【0012】

続けて、接続していたデータバスを切断し、切断していたデータバスを接続し、処理ブロック370に示すように、比較器160は、比較器160の退避レジスタ250とCPU110から出力された退避レジスタ130の値を比較する。比較した結果、二つのデータバスの処理結果がともに一致した場合、処理ブロック380に示すようにもう一方のCPUに対して、同様の処理を行う。比較した結果、両方とも不一致な場合、処理ブロック400に示すようにCPUを停止し、もう一方のCPUに対して、同様の処理を行う。比較した結果、どちらか一方が不一致だった場合、処理ブロック390に示すように不一致を検出したデータバスを切断し、もう一方のCPUに対して、同様の処理を行う。

50

【 0 0 1 3 】

つまり、特許文献 1 に開示されている半導体集積回路は、2 個の CPU で構成されるシステムにおいて、比較器内の退避レジスタの値とデータバスを介して CPU から出力された退避レジスタの値を比較することによって、データバスのエラーを検出し、エラーの発生したデータバスを切断し、エラーが発生していないデータバスを使用して処理を継続する。

【 先行技術文献 】

【 特許文献 】

【 0 0 1 4 】

【 特許文献 1 】 特開平 9 - 2 7 4 6 0 9 号公報

10

【 発明の概要 】

【 発明が解決しようとする課題 】

【 0 0 1 5 】

しかし、特許文献 1 に開示されている半導体集積回路においては、データバス 1 9 0 または、データバス 2 0 0 にエラーが発生した場合、エラー箇所が断定されず継続処理となる。そのため、CPU からデータバス 1 9 0 またはデータバス 2 0 0 を経由してメモリへ退避されるデータは、破損し、誤ったデータがメモリに格納される。その結果、不一致発生後に半導体集積回路がメモリに格納されたデータを用いて不一致発生直前からの動作を再開できず、信頼性を確保した上での故障延命が実現できないという問題点がある。

【 0 0 1 6 】

20

具体的には、図 7 において、例えばデータバス 1 9 0 にエラーがあった場合、比較器 1 6 0 は、CPU 1 1 0 からデータバス 1 9 0、データバス 2 1 0 を経由した退避レジスタ 1 3 0 の値と、エラー発生時に CPU 1 1 0 からデータバス 1 9 0、データバス 2 1 0 を経由した退避レジスタ 1 3 0 の値を格納した退避レジスタ 2 5 0 の値を比較する。しかし比較結果は一致し、比較器 1 6 0 は、データバス 1 9 0 にエラーがないと誤判定する。理由は、退避レジスタ 2 5 0 の値は常に書き換わるため、データバス 1 9 0 がエラーとなった時点で退避レジスタ 2 5 0 に誤った値が格納され、データバス 1 9 0、データバス 2 1 0 を経由した退避レジスタ 1 3 0 と同じ値となるためである。データバス 1 9 0 にエラーがあるにも関わらず、エラーがないと誤判定して動作を継続し、故障したデータバス 1 9 0 経由で主記憶装置 1 7 0 にデータを退避するため、主記憶装置 1 7 0 には、誤ったデータが格納される。

30

【 課題を解決するための手段 】

【 0 0 1 7 】

本発明の第 1 の態様にかかる半導体集積回路は、プロセッサと、前記プロセッサの演算結果と、第 1 のサンプルデータとを保持する第 1 のメモリと、前記第 1 のサンプルデータと同一の第 2 のサンプルデータを保持する第 2 のメモリと、前記第 1 のメモリから出力された前記第 1 のサンプルデータと、前記第 1 のサンプルデータが出力された経路と重複しない経路を介して前記第 2 のメモリから出力された前記第 2 のサンプルデータとを比較するデータ比較判定部と、を備え前記プロセッサは、前記第 1 のサンプルデータと前記第 2 のサンプルデータとが一致する場合に、前記演算結果を、前記第 1 のサンプルデータが前記データ比較判定部に出力された経路と同じ経路を介して前記第 1 のメモリから退避させるものである。

40

【 0 0 1 8 】

このような半導体集積回路を用いることにより、第 1 のメモリに格納されている演算結果は、第 1 のサンプルデータが出力された経路と同じ経路を介して退避される。そのため、データ比較判定部によって、第 1 のサンプルデータが出力された経路の正常性が確認された場合、演算結果も正常性が確認された経路を介して退避されるため、退避された演算結果の信頼性が向上する。

【 0 0 1 9 】

本発明の第 2 の態様にかかるデータ退避方法は、プロセッサと、前記プロセッサの演算

50

結果と、第 1 のサンプルデータとを保持する第 1 のメモリと、前記第 1 のサンプルデータと同一の第 2 のサンプルデータを保持する第 2 のメモリと、を備える半導体集積回路のデータ退避方法であって、前記第 1 のメモリから出力された前記第 1 のサンプルデータと、前記第 1 のサンプルデータが出力された経路と重複しない経路を介して前記第 2 のメモリから出力された前記第 2 のサンプルデータとを比較し、前記第 1 のサンプルデータと前記第 2 のサンプルデータとが一致する場合に、前記演算結果を、前記第 1 のサンプルデータが出力された経路と同じ経路を介して前記第 1 のメモリから退避させるものである。

【0020】

このようなデータ退避方法を用いることにより、第 1 のメモリに格納されている演算結果は、第 1 のサンプルデータが出力された経路と同じ経路を介して退避される。そのため、第 1 及び第 2 のサンプルデータの比較結果により、第 1 のサンプルデータが出力された経路の正常性が確認された場合、演算結果も正常性が確認された経路を介して退避されるため、退避された演算結果の信頼性が向上する。

10

【発明の効果】

【0021】

本発明により、正常性が確保された経路を介してデータの退避を行うことができる半導体集積回路及びデータ退避方法を提供することができる。

【図面の簡単な説明】

【0022】

【図 1】実施の形態 1 にかかる CPU 搭載回路の構成図である。

20

【図 2】実施の形態 1 にかかる CPU 動作判定回路の構成図である。

【図 3】実施の形態 1 にかかる通常動作停止処理のフローチャートである。

【図 4】実施の形態 1 にかかる半導体集積回路のリセット処理のフローチャートである。

【図 5】実施の形態 1 にかかる半導体集積回路のリセット後通常動作を開始する処理のフローチャートである。

【図 6】実施の形態 2 にかかる通常動作停止処理のフローチャートである。

【図 7】特許文献 1 にかかる半導体集積回路の構成図である。

【図 8】特許文献 1 にかかる半導体集積回路の正常性判定処理のフローチャートである。

【発明を実施するための形態】

【0023】

30

(実施の形態 1)

以下、図面を参照して本発明の実施の形態について説明する。はじめに、図 1 を用いて、本発明の実施の形態 1 にかかる半導体集積回路に含まれる CPU 搭載回路 10 の構成例について説明する。CPU 搭載回路 10 は、CPU コア 11 と、CPU コア 12 と、演算結果比較回路 13 と、通常メモリ 14 とを備えている。また、CPU コア 11 と、CPU コア 12 と、演算結果比較回路 13 とは、それぞれ CPU 動作判定回路 20 と接続されている。

【0024】

CPU コア 11 及び CPU コア 12 は、プロセッサに含まれ、RAM (Random Access Memory) または ROM (Read Only Memory) 等に格納されているプログラムに基づいてプログラム処理を実行する。CPU コア 11 及び CPU コア 12 は、同一の処理を実行し、冗長に構成されている。そのため、CPU コア 11 及び CPU コア 12 に故障等が発生していない場合、CPU コア 11 及び CPU コア 12 は、同一の演算結果を出力する。CPU コア 11 及び CPU コア 12 は、演算結果を演算結果比較回路 13 へ出力する。さらに、CPU コア 12 は、演算結果を通常メモリ 14 へ出力する。

40

【0025】

演算結果比較回路 13 は、CPU コア 11 から出力された演算結果と、CPU コア 12 から出力された演算結果とを比較する。演算結果比較回路 13 は、それぞれの演算結果が一致する場合、CPU コア 11 及び CPU コア 12 は正常に動作していると判定する。演算結果比較回路 13 は、それぞれの演算結果が一致しない場合、CPU コア 11 又は CP

50

Ｕコア１２のいずれかに故障又はエラーが発生していると判定し、演算結果の比較結果を演算結果比較回路１３へ出力する。

【００２６】

通常メモリ１４は、ＣＰＵコア１２からＣＰＵコア１２の通常動作時の演算結果を受け取り格納する。ＣＰＵコア１２における通常動作とは、例えば半導体集積回路が車載用であれば、車載の機能を実現するために実行される動作である。通常メモリ１４は、ＣＰＵコア１２から演算結果の退避処理を実行する旨の通知を受けた場合、格納している演算結果をＣＰＵコア１２へ出力する。ＣＰＵコア１２は、通常メモリ１４から受け取った演算結果をＣＰＵ動作判定回路２０へ出力する。通常メモリ１４は、ＣＰＵ搭載回路１０に搭載されているＲＡＭ等のメモリであってもよく、ＣＰＵ搭載回路１０の外部に接続されているハードディスク装置等の記憶装置であってもよい。

10

【００２７】

また、通常メモリ１４は、演算結果比較回路１３において、ＣＰＵコア１１及びＣＰＵコア１２のそれぞれの演算結果が一致していないと判定された場合に、ＣＰＵコア１２に対してサンプルデータを出力する。サンプルデータは、通常メモリ１４に格納されている演算結果を退避させる際に、退避経路に障害が発生しているか否かを判定する際に用いられる。サンプルデータは、ＣＰＵコア１１及びＣＰＵコア１２が通常動作を開始する際に、後述するサンプルデータ格納メモリ２４から出力され、通常メモリ１４に格納される。もしくは、サンプルデータは、ＣＰＵ搭載回路１０の製造時等に、あらかじめ通常メモリ１４に格納されてもよい。

20

【００２８】

ここで、ＣＰＵコア１１及びＣＰＵコア１２の動作モードについて説明する。ＣＰＵコア１１及びＣＰＵコア１２は、通常動作と、故障判定動作と、退避動作とを切り替えて動作する。通常動作は、上述したように、例えば半導体集積回路が車載用であれば、車載の機能を実現するために実行される動作である。故障判定動作は、通常メモリ１４に格納されている演算結果を退避する際に、退避経路に故障や、障害が発生しているか否かを判定する動作である。退避動作は、ＣＰＵコア１２が、通常メモリ１４に格納されているデータを不揮発性メモリ等へ退避させる動作である。

【００２９】

ＣＰＵコア１１及びＣＰＵコア１２は、ＣＰＵ動作判定回路２０の制御によって、通常動作と故障判定動作と退避動作とを切り替えられる。

30

【００３０】

図１においては、ＣＰＵコアが２つ搭載された回路の構成例について説明したが、ＣＰＵ搭載回路１０は、ＣＰＵコアを３つ以上搭載してもよい。

【００３１】

続いて、図２を用いて本発明の実施の形態１にかかる半導体集積回路に含まれるＣＰＵ動作判定回路２０の構成例について説明する。ＣＰＵ動作判定回路２０は、コード切替回路２１と、故障判定コード格納メモリ２２と、データ退避コード格納メモリ２３と、サンプルデータ格納メモリ２４と、動作制御回路２５と、データ比較判定器２６と、不揮発性メモリ２７と、を備えている。また、コード切替回路２１は、通常コード格納メモリ３０と接続されている。通常コード格納メモリ３０は、図２のようにＣＰＵ動作判定回路２０の外部に設けられてもよく、もしくはＣＰＵ動作判定回路２０の内部に設けられてもよい。

40

【００３２】

コード切替回路２１は、通常コード格納メモリ３０から出力される通常コードと、故障判定コード格納メモリ２２から出力される故障判定コードと、データ退避コード格納メモリ２３から出力されるデータ退避コードとのいずれかを選択する。コード切替回路２１は、選択したコードをＣＰＵコア１１及びＣＰＵコア１２へ出力し、ＣＰＵコア１１及びＣＰＵコア１２の動作を制御する。

【００３３】

50

コード切替回路 21 は、演算結果比較回路 13 から出力された比較結果又はデータ比較判定器 26 から出力された比較結果に応じて、CPU コア 11 又は CPU コア 12 へ出力するコードを選択する。コード切替回路 21 は、演算結果比較回路 13 から、CPU コア 11 及び CPU コア 12 の演算結果が一致しないとの比較結果を含む信号（以下、演算比較結果信号と称する）を受け取った場合、故障判定コードを選択し、CPU コア 11 及び CPU コア 12 の動作を通常動作から動作判定動作へ切り替える。また、コード切替回路 21 は、データ比較判定器 26 から、通常メモリ 14 に格納されている演算結果を退避させる際の退避経路に障害は発生していないとの情報を受け取った場合、データ退避コードを選択し、CPU コア 11 及び CPU コア 12 の動作を、動作判定動作からデータ退避動作へ切り替える。

10

【0034】

動作制御回路 25 は、CPU コア 11 及び CPU コア 12 に対して割り込み信号を出力し、通常動作コードと、故障判定コードと、データ退避コードとを切り替える制御を行う。また、動作制御回路 25 は、CPU コア 11 及び CPU コア 12 の動作を停止させる制御も行う。動作制御回路 25 は、演算結果比較回路 13 から、CPU コア 11 及び CPU コア 12 の演算結果が一致しないとの演算比較結果信号を受け取った場合、及び、データ比較判定器 26 から、通常メモリ 14 に格納されている演算結果を退避させる際の退避経路に障害は発生していないとの情報を受け取った場合に、CPU コア 11 及び CPU コア 12 に対して割り込み信号を出力する。

20

【0035】

データ比較判定器 26 は、CPU コア 12 から出力されるサンプルデータと、サンプルデータ格納メモリ 24 に格納されているサンプルデータとを比較する。データ比較判定器 26 は、CPU コア 12 から出力されるサンプルデータと、サンプルデータ格納メモリ 24 に格納されているサンプルデータとを、それぞれ重複しない経路を介して受け取る。

【0036】

CPU コア 12 は、コード切替回路 21 から故障判定コードを受け取ると、通常メモリ 14 に格納されているサンプルデータを抽出し、データ比較判定器 26 へ出力する。データ比較判定器 26 は、CPU コア 12 から出力されるサンプルデータと、サンプルデータ格納メモリ 24 に格納されているサンプルデータとを比較し、一致しているか否かを判定する。データ比較判定器 26 は、CPU コア 12 から出力されるサンプルデータと、サンプルデータ格納メモリ 24 に格納されているサンプルデータとが一致していると判定した場合、一致していることを示す情報を含む信号（以下、データ比較結果信号と称する）をコード切替回路 21 及び動作制御回路 25 へ出力する。また、データ比較判定器 26 において CPU コア 12 から出力されるサンプルデータと、サンプルデータ格納メモリ 24 に格納されているサンプルデータとが一致していないと判定された場合、CPU 搭載回路 10 及び CPU 動作判定回路 20 を含む半導体集積回路はリセットされる。

30

【0037】

また、データ比較判定器 26 は、一致していることを示す情報を含むデータ比較結果信号をコード切替回路 21 及び動作制御回路 25 へ出力した後に、通常メモリ 14 から、CPU コア 12 を介して CPU コア 12 における演算結果を受け取る。通常メモリ 14 から出力される演算結果は、通常メモリ 14 に格納されているサンプルデータがデータ比較判定器 26 へ出力された経路と同じ経路を介して、データ比較判定器 26 へ出力される。

40

【0038】

データ比較判定器 26 が受け取る演算結果は、半導体集積回路がリセットされた後に、故障又はエラー発生直前からの動作を再開するために用いられる。データ比較判定器 26 は、受け取った演算結果を不揮発性メモリ 27 へ出力する。これにより、CPU コア 12 は、通常メモリ 14 に格納されている演算結果を不揮発性メモリ 27 へ退避することができる。不揮発性メモリ 27 へ退避されたデータは、半導体集積回路がリセットされた場合においても削除等されない。

【0039】

50

続いて、図3を用いて本発明の実施の形態1にかかる通常動作停止処理の流れについて説明する。はじめに、CPUコア11及びCPUコア12は、通常動作を行う(S1)。CPUコア11及びCPUコア12は、通常動作を行っている間、それぞれの演算結果を演算結果比較回路13へ出力する。

【0040】

次に、演算結果比較回路13は、CPUコア11及びCPUコア12の演算結果を比較する(S2)。演算結果比較回路13によってCPUコア11及びCPUコア12の演算結果が一致していると判定された場合、ステップS1に戻り、CPUコア11及びCPUコア12は、通常動作を繰り返す。演算結果比較回路13によってCPUコア11及びCPUコア12の演算結果が一致していないと判定された場合、つまり不一致と判定された場合、演算結果比較回路13は、CPUコアの演算結果が不一致であるとの情報を含む信号(以下、演算結果不一致信号と称する)をコード切替回路21及び動作制御回路25へ出力する。動作制御回路25は、演算結果不一致信号を受け取ると、CPUコア11及びCPUコア12の通常動作を停止させる(S3)。

10

【0041】

続いて、図4を用いて本発明の実施の形態1にかかる半導体集積回路のリセット処理の流れについて説明する。半導体集積回路のリセット処理は、図3における通常動作停止処理の後に行われる。はじめに、コード切替回路21は、演算結果不一致信号を受け取ると、故障判定コードを選択し、CPUコア11及びCPUコア12へ選択した故障判定コードを出力する(S11)。次に、CPUコア12は、故障判定コードを受け取ると、通常メモリ14に格納されているサンプルデータを読み出し、データ比較判定器26へ出力する(S12)。ここで、例えば、CPUコア12は、故障判定コードを受け取ると、サンプルデータの読み出し以外に、簡単なCPUコア12における演算を含ませることにより、CPUコア12の動作確認範囲を広げることできる。たとえば、サンプルデータが16ビットのAAAAHという値だった場合、サンプルデータを2倍に乗算し、1を加算することにより、5555Hという値になるか否かを確認する。これにより、CPUコア12に含まれる乗算器及び加算器(図示せず)についても故障有無を判定することができる。

20

【0042】

次に、データ比較判定器26は、CPUコア12から受け取ったサンプルデータと、サンプルデータ格納メモリ24に格納されているサンプルデータとを比較する(S13)。データ比較判定器26は、比較結果を含む信号(以下、サンプルデータ比較結果信号と称する)をCPUコア12へ出力する。CPUコア12が、比較結果が不一致であるサンプルデータ比較結果信号を受け取った場合、半導体集積回路はリセットされる(S16)。半導体集積回路のリセットは、CPUコア12が半導体集積回路のリセットを制御する機能を有し、CPUコア12の制御により行われてもよい。もしくは、CPUコア12とは異なる回路等において、リセット制御回路が設けられ、リセット制御回路の制御により、半導体集積回路のリセットが行われてもよい。

30

【0043】

CPUコア12は、比較結果が一致するとのサンプルデータ比較結果信号を受け取った場合、通常メモリ14に格納されている退避対象データを読み出し、データ比較判定器26へ出力する。データ比較判定器26は、受け取った退避対象データを不揮発性メモリ27へ出力する(S14)。これにより、通常メモリ14に格納されている退避対象データが、不揮発性メモリ27へ退避される。

40

【0044】

ここで、退避対象データとは、半導体集積回路をリセットした後に、動作再開までの時間を短縮するために、CPUコア11及びCPUコア12に設定される演算結果である。CPUコア11及びCPUコア12は、半導体集積回路がリセットされた後に、CPUコア11及びCPUコア12の演算結果の不一致が発生する前にCPUコア12から出力された演算結果(退避対象データ)を設定する。

50

【 0 0 4 5 】

次に、CPUコア11及びCPUコア12は、CPUコア11及びCPUコア12のレジスタ等に格納されている不揮発性メモリ使用フラグの値を1にセットする(S15)。次に、CPUコア12又は、リセット制御回路等は、半導体集積回路のリセットを行う(S16)。

【 0 0 4 6 】

続いて、図5を用いて本発明の実施の形態1にかかる半導体集積回路のリセット後通常動作を開始するまでの処理の流れについて説明する。はじめに、半導体集積回路がリセットされた後、CPUコア11及びCPUコア12が起動され、CPUコア11及びCPUコア12の動作が開始される(S21)。

10

【 0 0 4 7 】

次に、CPUコア12は、通常メモリ14に対して、サンプルデータ格納メモリ24に格納されているサンプルデータと同一のサンプルデータを書き込む(S22)。CPUコア12は、通常動作を行う場合に、通常コードを実行する。ここで、通常コードの最初もしくは2番目等の順番が早い命令において、サンプルデータの読み出しを規定することにより、CPUコア12は、通常コードの実行とともに、サンプルデータ格納メモリ24からサンプルデータを読み出す。CPUコア12は、読み出したサンプルデータを通常メモリ14へ格納する。もしくは、サンプルデータは、CPU搭載回路10の製造時等に、通常メモリ14に予め書込まれていてもよい。

【 0 0 4 8 】

例えば、サンプルデータは、退避対象データ、つまり、CPUコア12の演算結果と同じビット幅をもったデータとする。また、全ビットにおける0及び1両方の設定状態を保証するために、例えば退避対象データが16ビットの場合は、5555H及びAAAAH等のように、データの全ビットの0及び1それぞれの状態を確認できるデータ2つを使用する。また、通常メモリ14においてサンプルデータが書込まれるアドレスについても、退避対象データが格納されているアドレスと同じビット幅のアドレスの全ビットが確認できる番地が用いられる。たとえば、退避対象データが書込まれるアドレスが16ビットの場合は、5555H番地及びAAAAH番地等にサンプルデータが書込まれる。

20

【 0 0 4 9 】

次に、CPUコア11及びCPUコア12は、不揮発性メモリ使用フラグを確認する(S23)。不揮発性メモリ使用フラグは、不揮発性メモリ27に演算結果が退避されているか否かを示している。不揮発性メモリ使用フラグは、例えば、CPUコア11及びCPUコア12のそれぞれのレジスタ等に格納されてもよい。不揮発性メモリ使用フラグの値が1の場合、CPUコア11及びCPUコア12は、不揮発性メモリ27から退避対象データを読み込む(S24)。これにより、CPUコア11及びCPUコア12は、半導体集積回路の内部状態を、リセット前のCPUコア11及びCPUコア12の演算結果不一致直前の状態に設定する。また、CPUコア11及びCPUコア12は、不揮発性メモリ27から退避対象データの読み込みを完了した後に、不揮発性メモリ使用フラグを0に設定し、通常動作を行う(S25)。ステップS23において、不揮発性メモリ使用フラグの値が0の場合、CPUコア11及びCPUコア12は、不揮発性メモリ27へアクセスせず、通常動作を開始する(S25)。

30

40

【 0 0 5 0 】

以上説明したように、本発明の実施の形態1にかかる半導体集積回路を用いることにより、CPUコア11及びCPUコア12の演算結果を用いた自己診断を行うことができる。CPUコア11及びCPUコア12の演算結果に不一致が発生した場合、CPUコア12は、通常メモリ14に格納されており、不一致が発生する前の演算結果をCPUコア12へ退避させる。ここで、予め通常メモリ14に格納されているサンプルデータと、サンプルデータ格納メモリ24に格納されているサンプルデータとを比較することにより、CPUコア12から不揮発性メモリ27へ退避させるデータの経路の故障の有無を判断することができる。これにより、経路に故障が無いと判断された場合、CPUコア12は、不

50

揮発性メモリ27へ正しい演算結果を退避させることができる。

【0051】

また、半導体集積回路のリセット後に、CPUコア11及びCPUコア12は、不揮発性メモリ27に退避されている演算結果を用いて、CPUコア11及びCPUコア12の演算結果不一致発生直前からの動作を再開することにより、通常動作開始までの時間を短縮することができる。

【0052】

(実施の形態2)

続いて、図6を用いて本発明の実施の形態2における通常動作停止処理の流れについて説明する。ステップS31及びS32は、図3におけるステップS1及び2と同様であるため、詳細な説明を省略する。演算結果比較回路13においてCPUコア11及びCPUコア12から出力される演算比較結果が不一致であると判定された場合(S32)、CPUコア12は、通常メモリ14に対して格納すべきデータ、つまり、退避対象データの演算比較結果に不一致があるか否かを判定する(S33)。

10

【0053】

退避対象データは、例えば、RAM又はROM等に格納されているプログラムを実行した際の演算結果や、演算結果を格納する通常メモリ14の格納先アドレス、通常メモリ14に対するread enable信号、通常メモリ14に対するwrite enable信号等である。また、リセット後の動作再開のためのデータに限らず、ユーザ固有の使用方法に合わせて最適化された設定情報や、使用環境データとして温度、電圧設定情報等も、後の解析等で参照するために退避対象データとされてもよい。

20

【0054】

また、退避対象外となるデータは、例えば、CPUコア11及びCPUコア12に対する割り込み信号や、通常メモリ14以外のメモリへアクセスする際のメモリアクセス信号等である。

【0055】

CPUコア12は、演算結果比較回路13から出力される演算比較結果に基づいて、退避対象データの比較結果に不一致が発生しているか否かを判定する。CPUコア12によって、退避対象データの比較結果が一致していると判定された場合、退避対象データは破損していないと推定される。そのため、演算結果比較回路13は、CPUコアの演算結果が不一致であるとの情報を含む演算結果不一致信号をコード切替回路21及び動作制御回路25へ出力する。動作制御回路25は、演算結果不一致信号を受け取ると、CPUコア11及びCPUコア12の通常動作を停止させる(S34)。その後、CPUコア12は、破損していない退避対象データを不揮発性メモリ27へ退避させる。

30

【0056】

CPUコア12によって、退避対象データの比較結果に不一致が発生していると判定された場合、退避対象データに破損が生じている可能性があるために、半導体集積回路はリセットされる(S35)。つまり、破損が生じている可能性がある退避対象データを、半導体集積回路をリセットした後にCPUコア11及びCPUコア12に設定した場合、CPUコア11及びCPUコア12は、故障発生前の状態に設定されず、通常動作を継続することができない。そのため、このような場合、退避対象データを不揮発性メモリ27に退避させることなく、半導体集積回路はリセットされる。

40

【0057】

以上説明したように、本発明の実施の形態2にかかる通常動作停止処理を用いることにより、不揮発性メモリ27に退避されたデータの信頼性を向上させることができる。

【0058】

なお、本発明は上記実施の形態に限られたものではなく、趣旨を逸脱しない範囲で適宜変更することが可能である。

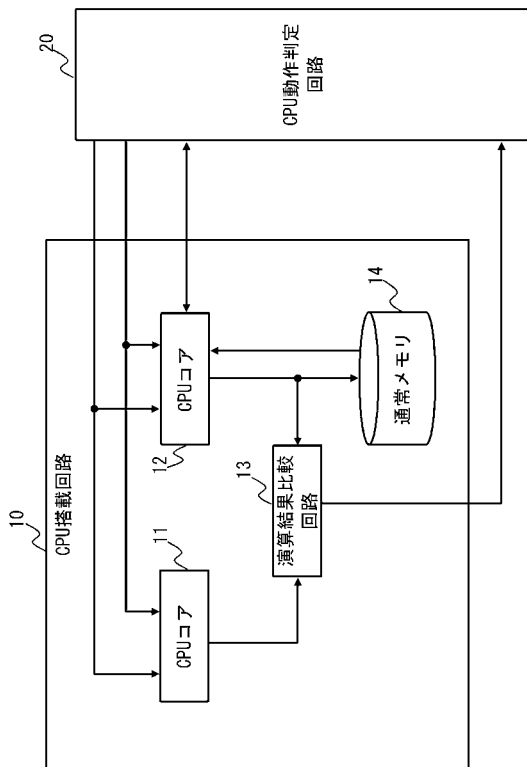
【符号の説明】

【0059】

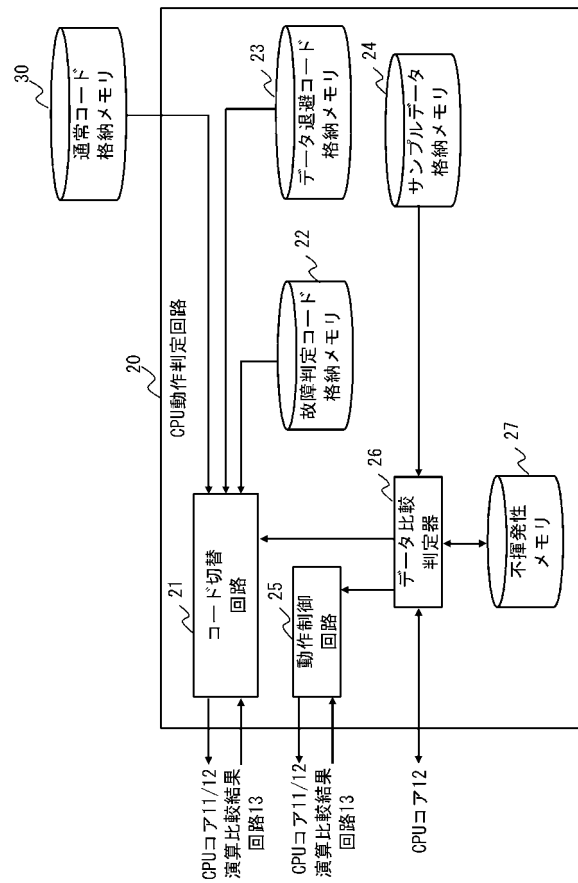
50

- 10 CPU搭載回路
- 11 CPUコア
- 12 CPUコア
- 13 演算結果比較回路
- 14 通常メモリ
- 20 CPU動作判定回路
- 21 コード切替回路
- 22 故障判定コード格納メモリ
- 23 データ退避コード格納メモリ
- 24 サンプルデータ格納メモリ
- 25 動作制御回路
- 26 データ比較判定器
- 27 不揮発性メモリ
- 30 通常コード格納メモリ

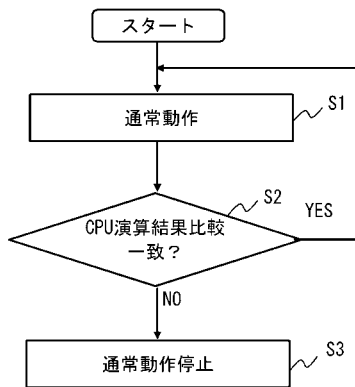
【 図 1 】



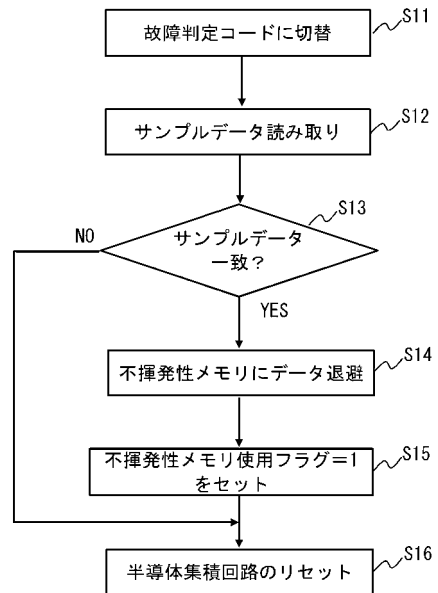
【 図 2 】



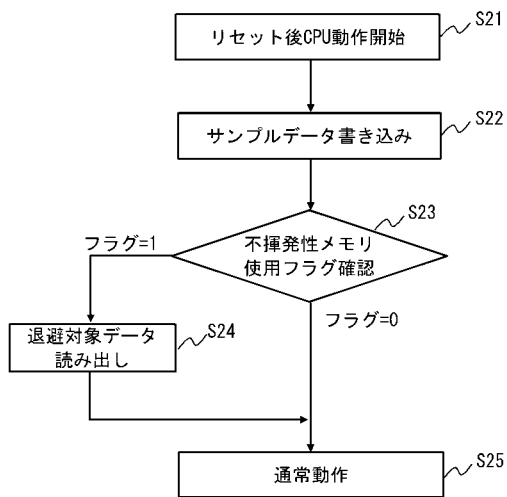
【 図 3 】



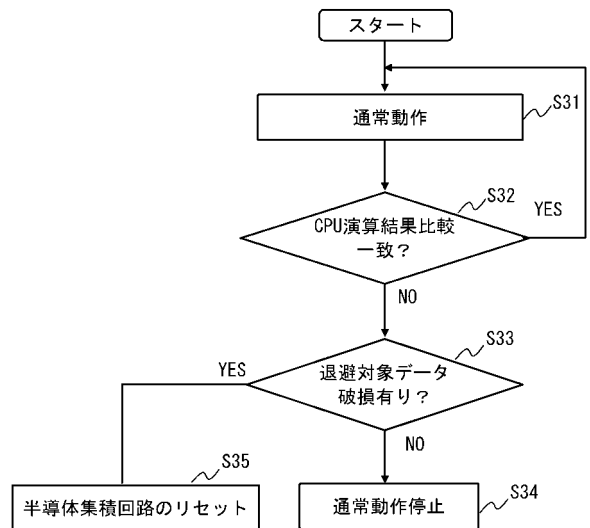
【 図 4 】



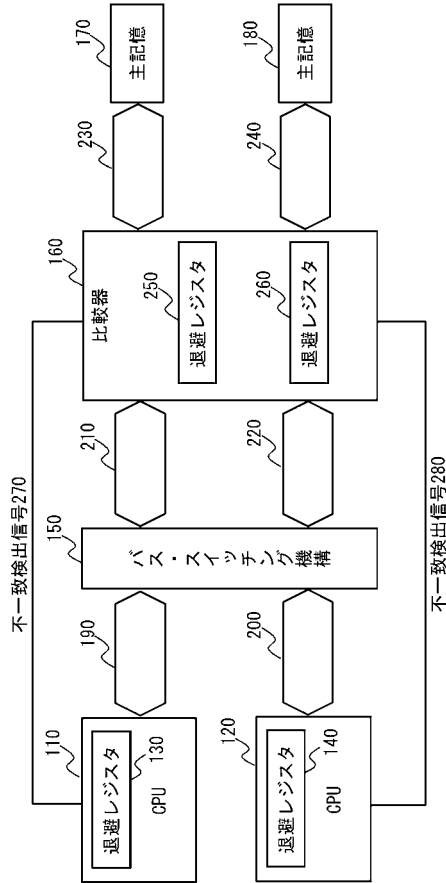
【 図 5 】



【 図 6 】



【 図 7 】



【 図 8 】

