機能安全に対応した車載アプリケーション向けFPGA設計手法の適用

シノプシス　FPGAベース・シンセシス・ソフトウェア・ツール担当プロダクト・マーケティング・マネージャー　Joe Mallett

現在、車載アプリケーションでは機能安全要件の順守が求められています。Synplify® Premierを使用することでどのようにこれらの要件に対応できるかについて、シノプシスのFPGAベース・シンセシス・ソフトウェア・ツール担当プロダクト・マーケティング・マネージャーJoe Mallettがご説明します。

車載アプリケーションの設計にFPGAを使うと、機能安全に対応したプラットフォームを車両に実装できるため、差別化した機能を迅速に実現できるというメリットがあります。しかしながら、車載向けに作られる一般のチップと同様に、FPGAもISO 26262などの規格で定義された機能安全要件を順守しなければなりません。

シノプシスのSynplify Premier FPGAシンセシス・ツールを使用すると、エラー検出およびミティゲーション・ロジックの合成を自動化できるため、機能的に安全な設計を迅速かつ確実に提供できます。

車載向け設計のニーズに応えるFPGA

車載アプリケーションにおける電子機器の使用は広く普及しています。インフォテインメント、ドライバー情報、車載ネットワーク、運転支援システムなど、運転者や同乗者が待ち望んできた高度な機能はどれも、ますます進化を遂げる高度な電子回路によって実現されています。

FPGAは、アプリケーションに特化した専用プロセッサの搭載により、高い柔軟性を提供し、高速インターフェイスに加えて、マルチ・スレッディングなどの複数機能を並列処理できる高い演算能力を有しているため、さまざまな車載アプリケーションに最適です。解析、画像ワーピング、高ダイナミック・レンジ、対象物の分類を含む複数の画像処理機能をサポートする必要のあるアプリケーションにも対応しており、サラウンドビュー・カメラや暗視装置ではFPGAの使用が一般的になりつつあります。

FPGAはシステム内での実装で使用される以外に、ASIC車載システムの高性能プロトタイプ作成でも使用できます。シノプシスのHAPSなどの市販のプロトタイピング・ボードを使用すると、プロジェクト・サイクルの早い段階で迅速に車載システムを検証できるようになります。

このように、FPGAが自動車のサブシステムで使用されるケースが増えてきたため、特定の設計手法を迅速かつ効率的に適用して、FPGA設計内で発生するエラーによるリスクを軽減することが求められています。

エラーの発生原因

FPGAメーカーは、コア電圧を抑え、スイッチングを高速化した先進のプロセス・テクノロジの開発に焦点を置いているため、FPGAそのものは、放射線が引き起こすシングル・イベント・アップセット（SEU）エラーの影響を受けやすくなっています。また、SEUは標高が上がるとさらに発生しやすくなり、わずか1,500mほどの標高（コロラド州デンバーの標高に相当）でも、FPGA回路が受ける影響は海面レベルの時の約4倍に上ります。

エラーが引き起こされる原因には、放射線や電圧のドロップアウトのほかにもさまざまなものがありますが、回路設計において重要なのは、原因を問わずあらゆる種類のエラーやビット反転を検出して修正できるようにすることで、システム全体にエラーが広がって重大な結果を招くことのないように防止することです。

Synplify Premierを使用した車載向けFPGAシステムの設計

車載システムの設計では、ロジック回路、ブロックRAM、I/Oに対して高い信頼性と機能的安全性を確保することが重要になりますが、このほかに、性能と面積を最適化する必要もあります。先進運転支援システム（ADAS）に対応したアプリケーションを含む多数の車載アプリケーションで、高い演算処理能力が必要とされており、可能な限り消費電力とコストを抑えたFPGAパッケージに対応する高スループット・アーキテクチャが求められています。

コストおよび消費電力の削減と機能安全要件の順守を両立する設計には、多数のトレードオフが伴います。たとえば、冗長性をもたせるために回路を三重化するとデザイン面積も3倍になるため、開発者は一部のみを選択的に三重化することで、面積コストの最小化を図ろうとします。このようなマニュアル作業を部分的に行うので、設計チームはデザインをある程度コントロールできますが、非常に時間がかかるうえ、ヒューマンエラーの原因にもなります。

Synplify Premierを使用すると、設計者が三重化したい箇所をマークするだけで、三重化ロジックのインプリメンテーション・プロセスが自動化されます。これにより、設計スケジュールが大幅に短縮されるだけでなく、設計チームは三重化すべき部分の選択作業の方に十分な時間をかけられるため、信頼性の向上と面積削減の両方を同時に実現できます。また、ISO 26262などの機能安全に関する規格では、厳密なトレーサビリティを確保するドキュメントの提供が義務付けられていますが、Synplify Premierが生成するログ・ファイルとレポートには、デザインの冗長性を実現するために複製された箇所が正確に記録されています。

ADASをはじめとする複雑なアプリケーションでは、サブ・アプリケーションごとに異なる機能安全要件があるため、機能的な安全性の実装方法を厳密にコントロールすることが非常に重要になります。

Synplify Premierは、次のようなエラー検出および修正テクニックをサポートしています。

フォールト・トレラントFSM

2つのメカニズムを使用して、ステート・レジスタ内のエラーを見つけて修正する安全性の高いFSMと、ステート・マシンに適切な動作を再開させるロジックを作成します。最初に、Hamming-3符号を使用して、ステート・レジスタ内のシングルビット・エラーを自動的に検出および修正します。これにより、適切な動作が即座に再開されます。次に、設計者が安全なFSMを指定すると、シングルビット・エラーが検出された場合に強制リセットが生成されるか、またはカスタムのエラー・リカバリ・スキームが作成されます。

冗長性に基づくリスクのミティゲーション

設計者が三重化するモジュールを個別に指定すると、Synplifyによって多数決回路を含む三重化ロジックが作成されます。FPGAプログラミングで使用できるTriple Modular Redundancy（TMR）テクニックには以下の3種類があります（図1 - ローカルTMR、分散TMR、ブロックTMR）。

ローカルTMR（LTMR）を使用した場合、レジスタを三重化して保護し、そこからの出力を多数決回路に渡すことで、エラー修正済みの最終的な出力が決定されます。

分散TMR（DTMR）では、放射線が引き起こすビット反転がミティゲーション・ロジックに影響するリスクを軽減するため、各TMRブロックを物理的に分離するプロセスが自動化されます。

ブロックレベルのTMR（BTMR）では、ブロックを三重化する際の粒度を設計者が指定でき、たとえばブロック内の内部ロジックはそのままでブロック全体を複製する、といった処理を実施できます。

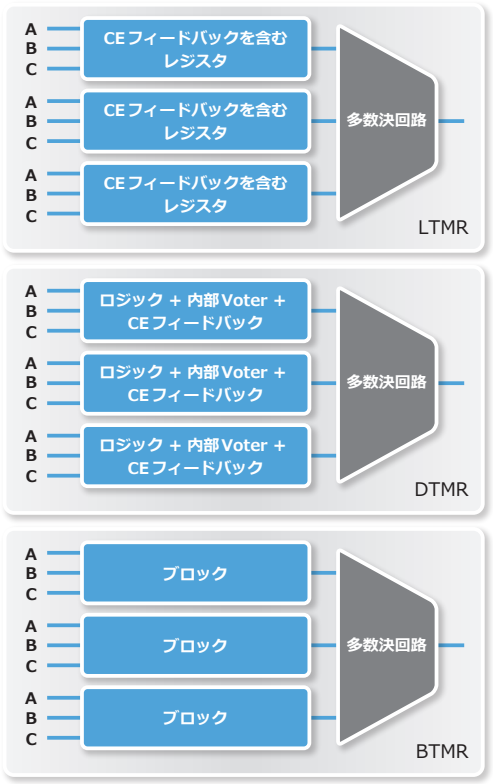

図1. Synplify PremierによるTMRの実装

メモリー・エラーの修正

3つのTMRテクニックのすべてで、ブロックRAM（BRAM）の三重化、ECC RAMの推論、エラー・フラグの作成を含むメモリー保護がサポートされており、コンフィギュレーションRAM（CRAM）のスクラビング処理の効果を高めます。

I/Oの三重化

ロジックを三重化して多数決ロジックを追加するだけでなく、I/Oの三重化がサポートされます。

デバッグ / エラー監視回路の挿入

比較付き複製（DWC）のサポートを通じてエラー・フラグを作成するとともに、force / bindを使用し、保護および観察用のデバッグ・ノードを作成することで、デバッグ回路を直接挿入します。

最終的に設計者は、システムに必要な機能安全の要件に関する知識を動員して、ミティゲーションの手法を用いる場所と方法を決めなくてはなりません。そして一度それを決めてしまえば、Synplify Premierは設計変更を自動的に実装することで設計の重要箇所を保護し、FPGA動作の安全性を確保します。

適切なTMRの選定

図2に示すのは、主なFPGAビルディング・ブロック・タイプの例であり、これらを保護する一般的な方法を併せて紹介します。

FPGAビルディング・ブロックには次のような要素が含まれます。組み合わせロジック・ブロック（CLB）は、ルックアップ・テーブル、レジスタ、乗算器またはDSP要素によって実装されます。BRAMは専用メモリー・プリミティブです。デジタル・クロック・ジェネレータ（DCM）はグローバル・クロック配線を介してFPGAのクロック・ネットワークを駆動します。最後に、FPGA入力 / 出力バッファ（IOB）は、チップ上またはチップ外のデータをクロック制御します。

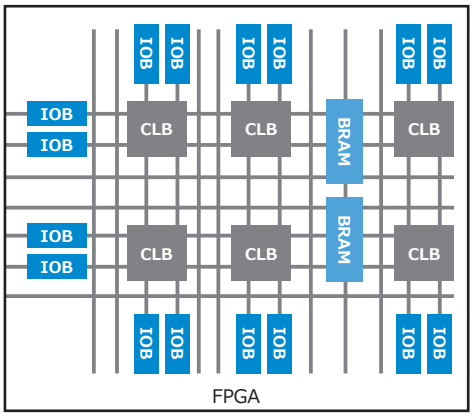

図2. TMRがダイ上のIOB、BRAM、CLBプリミティブを保護します。LTMRはレジスタを保護し、DTMRはモジュール全体とコンフィギュレーション・ビットを保護し、BTMRは内部でアクセスできないブロックおよびIPと、クロック・ネットワークならびに配線を保護します

はじめに、設計チームがデザイン内でエラーの影響を受けやすいブロックを決定する必要があります。次に、そのTMR手法が各FPGAテクノロジ（radhard、antifuse、flash、SRAMなど）に対して有効な手法とされているかどうかも確認する必要があります。

ブロック・タイプごとに効果のある手法はある程度異なります（図2）。例を挙げると、通常、LTMRはレジスタの保護に使用され、DTMRはCLB、コンフィギュレーション・ビット、BRAM、IOBで構成されたモジュール全体の保護に使用され、BTMRは内部でアクセスできないブロックおよびIPと、クロック・ネットワークならびに配線の保護に使用されます。

最後に開発者は、各エラー・ミティゲーション・アプローチにおける面積と性能のトレードオフについて検討します。

まとめ

FPGAは、車載アプリケーションにとってメリットの大きいインプリメンテーション・プラットフォームです。なぜなら、1個のFPGAに複数のサブシステムとコントローラを統合することで、部品コストが削減できるからです。また、FPGAの高いプログラマビリティにより、新機能の追加やアップグレードも容易になります。さらにFPGAアーキテクチャは高性能でレイテンシも小さいため、ADASなどの高い演算処理性能を必要とする最新機能に最適と言えます。

Synplify Premierを使用して信頼性の高いテクニックを自動化することで、設計者はFPGAベースの設計に機能安全を素早く統合できます。設計の複雑度によって異なりますが、エラー検出とミティゲーションを手動で実装する場合と比べると、数時間から数週間単位で実装期間を短縮できます。

自動車は平均使用年数が10年を超えるため、自動車メーカーは、採用するコンポーネントが数十年に渡って高い信頼性を維持できるものであることを確認しなければなりません。Synplify Premierは、回路の三重化、安全なFSM、メモリー、I/Oなどのさまざまな設計テクニックをサポートしています。またFPGAデバイスの種類にかかわらず、単一の設計フローでデザインをさまざまなFPGAデバイスに実装できます。

詳細情報

- ウェブページ：FPGAベースのデザイン・ソリューション
  http://www.synopsys.com/JP2/tools/implementation/fpgaimplementation

著者紹介

Joe Mallett : シノプシスのFPGAベース・シンセシス・ソフトウェア・ツール担当プロダクト・マーケティング・マネージャー。半導体およびEDA業界における設計およびインプリメンテーション分野で20年以上の経験を持つ。シノプシス入社前はXilinx®社にてシニア・プロダクト・マーケティング・マネージャーとしてFPGA製品の定義と発売に従事。これまでに、SoC設計 / プロトタイピング、組み込みソフトウェア、HDL合成、IP、製品 / セグメント・マーケティングなど幅広い分野を手がける。ポートランド州立大学にて電気工学理学士の学位を取得。