

ホワイトペーパー

Simulinkとモデルベースデザインによる、 バッテリーマネジメントシステムの開発

様々な業界で、バッテリーパックによるエネルギー貯蔵への依存度の上昇により、多様な充放電および環境条件下で、最大のパフォーマンス、安全なオペレーション、および最適な寿命を保証できるバッテリーマネジメントシステム (BMS) の重要性が高まっています。これらの目的を満たすBMSを設計するために、次のようなフィードバック制御と監視制御アルゴリズムを開発します。

- セル電圧と温度を監視
- 充電状態 (SOC) および劣化状態 (SOH) の推定
- 熱および過充電保護のための電力入出力の制限
- 充電プロファイルの制御
- セルの充電状態のバランスング
- 必要に応じてバッテリーパックと負荷の切り離し

本ホワイトペーパーでは、Simulink® を用いてシステムレベルのシミュレーションを実行することにより、BMSのアルゴリズムとソフトウェアを開発する方法について説明します。Simulinkを使用したモデルベースデザインでは、バッテリーパックの振る舞いを理解したり、ソフトウェアアーキテクチャを調整したり、様々なオペレーションケースをテストしたり、ハードウェアテストを早期に実行することで、設計のエラーを削減することができます。モデルベースデザインにおいて、BMSモデルは、設計の機能面のデスクトップシミュレーションや、業界標準に対応した形式検証、リアルタイムシミュレーションやハードウェア実装を目的としたコード生成など、すべての設計、開発作業の基礎となります。(図1)

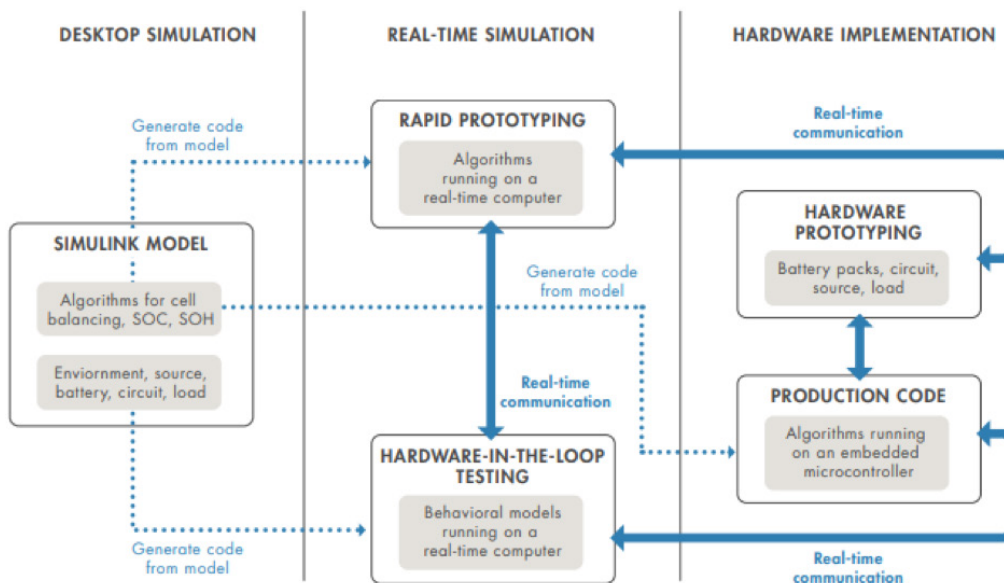


図1 Simulinkとモデルベースデザインによる、バッテリーマネジメントシステム開発ワークフロー。

デスクトップシミュレーション。 Simulinkのデスクトップシミュレーションでは、充放電の挙動 (シングルセルバッテリー等価回路を使用)、電子回路設計、フィードバック制御および監視制御アルゴリズムなど、BMS設計の機能面を検証できます。ビヘイビアモデルを使用して、バッテリーシステム、環境、およびアルゴリズムをシミュレーションできます。たとえば、セルバランスのアクティブ方式とパッシブ方式の設定とアルゴリズムを比較し、特定のアプリケーションに対する適合性を評価できます。また、ハードウェアプロトタイプを作成する前に、新しい設計アイデアを検討したり、複数のシステムアーキテクチャをテストすることもでき、故障分離検出時に、コンタクターが開閉されないようにするなど、要件テストも実行できます。

リアルタイムシミュレーション。シミュレーションでの検証後、Simulinkモデルからラピッドプロトタイピング (RP) またはHIL (Hardware-in-the-Loop) テスト用の C/HDLコードを生成し、リアルタイムでBMSアルゴリズムをさらに検証することができます。RPでは、リアルタイムテスト用のコードを手書きする必要はありません。コントローラーモデルからコードを生成し、量産用マイクロコントローラーの機能を実行するリアルタイムコンピュータに実装します。モデルからコードを自動生成できるため、モデル内のアルゴリズムを変更後、わずか数時間で、リアルタイムハードウェア上で試すことができます。さらに、Simulink内からリアルタイム制御ハードウェアを直接変更し、アルゴリズムのパラメーター変更や、テストデータのログが可能です。

ラピッドプロトタイピングと同様に、HILテストでもSimulinkモデルからコードを生成し、それをリアルタイムコンピュータに実装します。HILテストの場合、コードは制御アルゴリズムモデルではなくバッテリーシステムモデルから生成され、バッテリーパック、アクティブ方式やパッシブ方式の回路要素、負荷、充電器、その他のシステムコンポーネントを表す仮想リアルタイム環境を提供します。この仮想環境を使用し、ハードウェアプロトタイプを開発する前に、BMSコントローラーの機能をリアルタイムで検証できます。デスクトップシミュレーションで作成したテストをHILテストでも引き続き利用し、要求仕様を満たすことを確認しながら、BMSの設計を進めることができます。HILテストは主にマイクロコントローラーやFPGAで実行されるコードをテストするために行いますが、量産コントローラーハードウェアを選択する前に、*Simulink Real-Time™* と *Speedgoatターゲットマシン* のような、ラピッドプロトタイピングシステムをHILセットアップに接続して利用することもできます。

ハードウェア実装。ハードウェア実装段階では、デスクトップシミュレーション、RP、およびHILで作成・検証したSimulink制御モデルから、効率的で量産可能なBMS用のコードを生成します。量産コード生成プロセスは、自動車、航空宇宙やその他規制のある業界での正式な認証基準に準拠させることも可能です。

“MATLAB、Simulink、およびSimscape™で行うモデリング、シミュレーションは、物理的なプロトタイプを構築するよりも早く、安全で、そして低コストです。システム全体を動かさなくても、どのようなアルゴリズムや充電方法が設計に適しているかを特定し、実際のバッテリーでは困難だったり危険なシナリオをテストして、任意のアプリケーションや利用法に対して、設計を最適化させることができます。”

— Romeo Power, Cecilia Wang

» [ユーザー事例を読む](#)

デスクトップシミュレーション：BMSソフトウェアのモデリング

BMS制御ソフトウェア開発の核となる、現実的なシミュレーションを実行するには、バッテリーパックの正確なモデルが重要です。多くのバッテリーは、電池の物理的構成と電気熱化学的性質を表現する有限要素解析（FEA）モデルを使用して設計されています。こうしたモデルは、バッテリーパックの化学特性および配列の設計や最適化の点では優れていますが、制御エンジニアがシステムモデル設計やソフトウェア開発に必要とするモデルとは異なります。

“以前のバッテリーモデルは、理想的な電圧源と固定インピーダンスで設定した、経験則に基づくものでした。現在は、はるかに洗練された第一原理モデルを使っており、その結果として、電気化学デバイスとしての電池について多くを理解できています。Simulinkで構築した高度な等価回路モデルのおかげで、様々な充電状態、放電速度、温度、劣化を予測できるようになりました。

同様に、セーフティクリティカルな場面にもシミュレーションを行い、バッテリーパックがオーバーヒートしないよう、バッテリー内の冷却性能を予測しています。物理、化学、熱伝達などのマルチドメインに渡る影響を把握するには、通常は有限要素解析ツールと膨大な労力が必要ですが、MathWorksのツールを使用して分析を行うことで、バッテリー技術を劇的に進歩させる洞察を得ました。”

— Tesla Motors, Dr. Chris Gadda, Dr. Andrew Simpson

» [記事を読む](#)

バッテリーセルのモデリングと特性評価

SimulinkでBMSアルゴリズムを開発するときは、等価回路を使用してバッテリーセルの熱、電氣的挙動をシミュレーションします。等価回路は通常、電圧源、直列抵抗、および1つ以上の抵抗とコンデンサのペアを並列に備えたものから構成されます（図2）。電圧源は開回路電圧を、他の要素はセルの内部抵抗と時間依存の挙動をモデリングします。等価回路の要素は、一般的には温度とSOC（充電状態）に依存します。依存関係はバッテリーの化学特性に固有であるため、コントローラーが設計されたものと同種のバッテリーセルで実行された測定値を用いて決定する必要があります。SimulinkとMATLAB®の最適化機能を使用して、[実験データを活用したモデル作成](#)を例に、等価回路のパラメーターを設定できます。

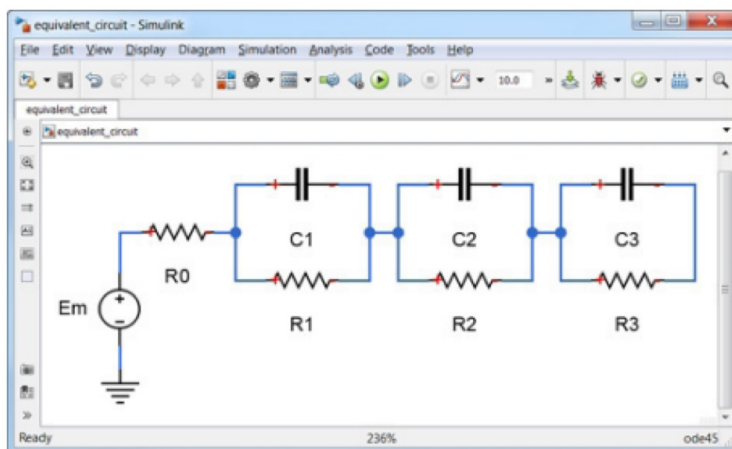


図2 3つの異なる時定数、内部抵抗、開回路電位を持つバッテリーの等価回路。

Simscape Electrical™ コンポーネントを使うと、シングルセルレベルからモジュールやパックレベルまで拡大し、セルを周辺回路と直感的に組み合わせることができます。パックレベルでシミュレーションを行うと、エネルギー貯蔵容量、電力供給率、および熱的動作範囲など、さまざまなパック構成の影響を評価でき、パックと電源、負荷、保護回路などの他のシステムコンポーネントとの相互作用も調べることができます。

バッテリーセルのモデリングと特性評価についてさらに詳しく

- [リチウム電池セル - 2つのRC分岐等価回路 - 例](#)
- [温度依存のリチウムイオン バッテリー モデル - 例](#)
- [バッテリーモデル - File Exchange](#)
- [二次電池モデルのパラメーター化 - 例](#)
- [バッテリーモデルパラメーター推定の自動化 \(9:55\) - ビデオ](#)
- [層状手法を使用したバッテリーモデルパラメーターの推定：リン酸鉄リチウムセルを使用した例 - テクニカルペーパー](#)

パワーエレクトロニクスと受動部品のモデリング

バッテリーパックのモデルに加えて、現実的なBMSシミュレーションには、バッテリーシステムを電源や負荷に接続する回路部品の正確なモデルが必要です。Simulinkのアドオン製品であるSimscape Electricalは、セルバランスのアナログフロントエンドなど、バッテリーシステム回路構築に必要な能動部品や受動部品の完全なライブラリを提供します。充電源は、太陽光発電 (PV) システムなどのDC供給源、または電流が整流されるAC供給源から構成されます。

Simulinkを使用したシステムレベルのシミュレーションでは、バッテリー周辺の充電源を精緻に構築し、BMSを様々な動作範囲や、故障条件下で検証することができます。バッテリーパックの負荷もモデル化してシミュレーション可能です。例えば、バッテリーがインバーターを介して電気自動車 (EV) の永久磁石同期電動機 (PMSM) に接続されている場合をシミュレーションし、走行サイクルを通じてEVの動作を変化させ、充電状況の変化に対処するBMSの有用性を評価できます。

バッテリーセルのモデリングと特性評価についてさらに詳しく

- [IPMSM のトルクベースの負荷制御 - 例](#)

監視制御アルゴリズムの開発

BMS監視制御アルゴリズムの開発には、ステートマシンを利用して故障の検出や管理、充放電の電力制限、温度制御、セルバランスを行う監視ロジックをモデル化します。**Stateflow®** は、ステートマシンとフローチャートに基づいて、組み合わせとシーケンシャルの判定ロジックをモデリング、シミュレーションするための環境です。BMSの監視制御アルゴリズム開発にStateflowを利用すると、バッテリーシステムがイベントや時間ベースの条件や外部入力信号にどのように反応するかをモデル化することができます。例えば、定電流定電圧 (CCCV) 充電における、セルの電流充電モードから電圧充電モードへの移行を制御するステートロジックを開発、テストすることができます。

“モデルベースデザインでは、アイデアから量産コードの生成まで、統合された開発プロセスがあります。MathWorksのツールを使用することで、我々の専門知識を活かして、設計を早期に、かつ繰り返し検証できる環境で、重要なバッテリーマネジメント技術を開発することができました。”

— 東風電動車両、Xiaokang Liu博士

» ユーザー事例を読む

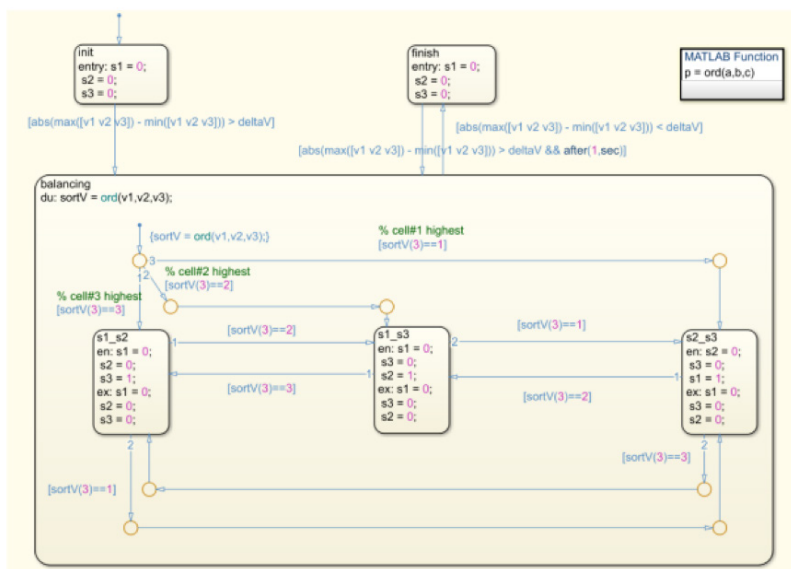


図3 Stateflowダイアグラムに実装されたセルバランスロジック。

監視制御アルゴリズムの開発についてさらに詳しく

- [Simulinkでのバッテリーマネジメントシステム開発 \(7:18\) - ビデオ](#)

充電状態 (SOC) の推定

SOC推定アルゴリズム開発には正確なバッテリーモデルが肝心です。開回路電圧 (OCV) 測定や電流積分 (クーロンカウント) などの伝統的なアプローチが有効な場合もありますが、フラットなOCV-SOC特性を持つ最新のバッテリーの化学的性質のSOC推定には、他のアプローチが必要です。正確な計算結果の実績を持つアプローチのひとつが拡張カルマンフィルタ (EKF) であり、Simulinkには、SOC推定のためのオブザーバ開発ができるEKFブロックが提供されています。こうしたオブザーバには、通常、セルから測定された電流と電圧を入力として使用する、対象となる非線形システム (バッテリー) のモデルと、二段階の予測/更新プロセス (図4) でシステムの内部状態 (SOCなど) を計算する再帰アルゴリズムが含まれます。

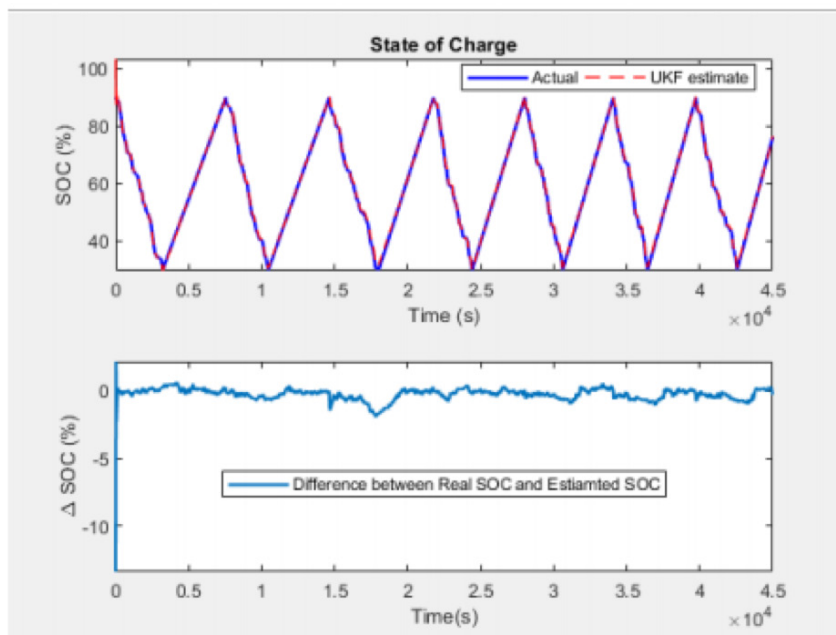


図4 Simulinkのカルマンフィルターを使用したバッテリーの充電状態の推定。

充電状態 (SOC) の推定についてさらに詳しく

- [拡張カルマンフィルターモデルに基づく充電状態 \(SOC\) の推定 - 記事](#)
- [バッテリーマネジメントシステムのリファレンスデザイン - インテルのドキュメント](#)
- [劣化するバッテリーシステムの非線形状態推定 - 例](#)
- [拡張カルマンフィルター - ドキュメンテーション](#)

劣化状態 (SOH) の推定

製造時には性能仕様を満たしていた電池も、時間の経過や使用サイクルを経て劣化することで、充電量が減り、内部抵抗が増えていきます。内部抵抗の増加は、短時間の測定を基に容易に推定することができますが、充電量の減少を正確に計算するには満充電と放電を行う必要があります。そのため、劣化状態 (SOH) 推定だけでなく、バッテリーの状態に加えパラメーターも含むように増強されたEKFに関心が高まっています。瞬時内部抵抗の正確な推定は、BMSが電力制限を設定するのに非常に役立ちます。

SOH測定方法には、客観的な共通の定義がないため、各社が独自にSOH測定方法を設定しており、SOH推定の汎用的な方法がありません。Simulinkを使用すれば、所属組織内のバッテリー状態解釈に基づく、カスタムSOHアルゴリズムを作成することができます。

劣化状態 (SOH) の推定についてさらに詳しく

- [電気自動車アプリケーション向け正常な経年リチウムイオン電池のモデルベースパラメーター同定 - テクニカルペーパー](#)

デスクトップシミュレーションによるテスト

テストプロセスにデスクトップシミュレーションを組み込むと、可能な全てのロジックと閉ループ制御の分岐の組み合わせ、つまりハードウェアテストより圧倒的広範囲のカバレッジでテストケースを作成、実行し、BMSを動作させることができます。シミュレーションモデルはBMSの設計とテストを推進する実行可能な仕様として機能します (図5)。Simulinkを用いたデスクトップシミュレーションによるテストには、以下のようなメリットがあります。

- 制限、許容誤差、論理的チェック、および時間的条件などの要件を、元の仕様へのトレーサビリティを持ってモデルに組み込む
- 機能テスト、ベースラインテスト、等価テスト、バックツーバックテストを実行するシミュレーションベースの複雑なテストの構築
- 決定、条件、MC/DCなどの業界標準のメトリクスと、関係演算子の境界カバレッジの追跡
- 全てのモデルカバレッジとカスタムオブジェクトを達成するためのテスト入力の生成
- 形式検証で、整数オーバーフロー、デッドロジック、ゼロ除算につながる潜在的な設計エラーを検出

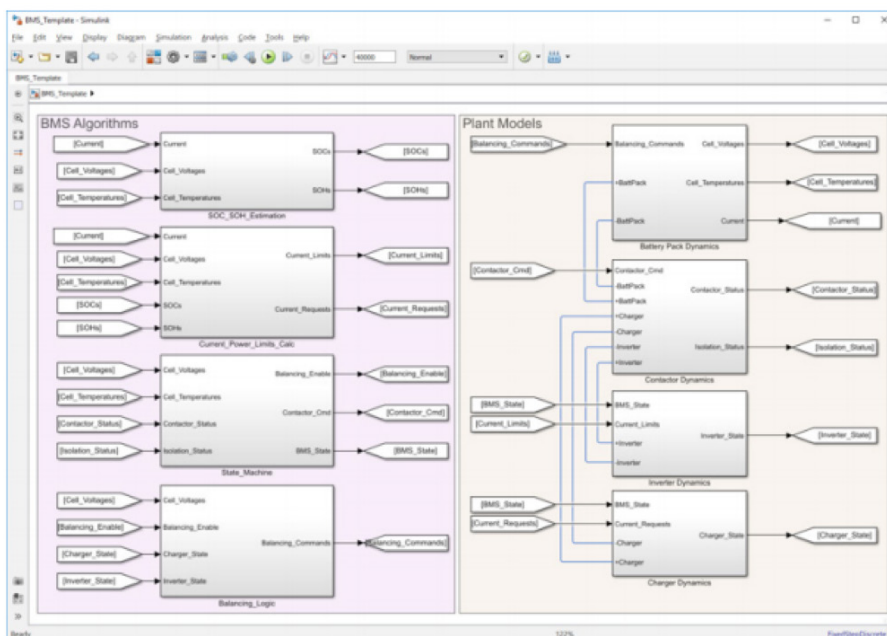


図5 バッテリーパック、コンタクター、インバーター、充電器を含むBMSアルゴリズムとプラントダイナミクスのSimulinkモデル。

IEC 61508、IEC 61851、ISO 26262などの規格に従って、ソフトウェア開発プロセスに形式検証ベースのテストを統合し、安全要件を満たすこともできます。

“モデルアドバイザーでスタイルガイドラインやモデル基準への準拠を頻繁に確認しながらSimulinkでコントローラーモデルを開発しました。また、Simulink Design Verifier™を使用して、モデル内のデッドロジック、ゼロ除算エラー、その他の設計エラーもチェックしました。”

— LG化学、Duck Young Kim、Won Tae Joe、Hojin Lee

» [記事を読む](#)

デスクトップシミュレーションによるテストをさらに詳しく

- モデルベースデザインによるハイブリッド自動車用バッテリーマネジメントシステム向けAUTOSAR、ISO26262準拠のソフトウェアの開発 - 記事
- モデルベースデザインによるTeslaロードスターの構築 - 記事

リアルタイムシミュレーション：BMSソフトウェアの検証

ラピッドプロトタイピングやHILテストなどのリアルタイムシミュレーションをすることで、BMS設計のハードウェア上での挙動を確認することができます。リアルタイムシミュレーションの目的は、システム全体の一機能をハードウェア上でエミュレートすることで、BMSコントローラーをRPで、バッテリーシステムのバランスをHILで確認などを行います。BMS設計において、非常に重要な利点を持ちます。

- 最終的なコントローラーハードウェアを選択する前にRPを実行し、アルゴリズムの検証
- リアルタイムテストシステムの柔軟性を活用し、設計の反復とテストの迅速な実施
- バッテリーシステムのプロトタイプハードウェア作成前のHILテスト
- ハードウェアでは実施が困難、高コスト、破壊的なテストケースに対し、RPとHILのテストを組み合わせたBMSアルゴリズムの実行（図6）。

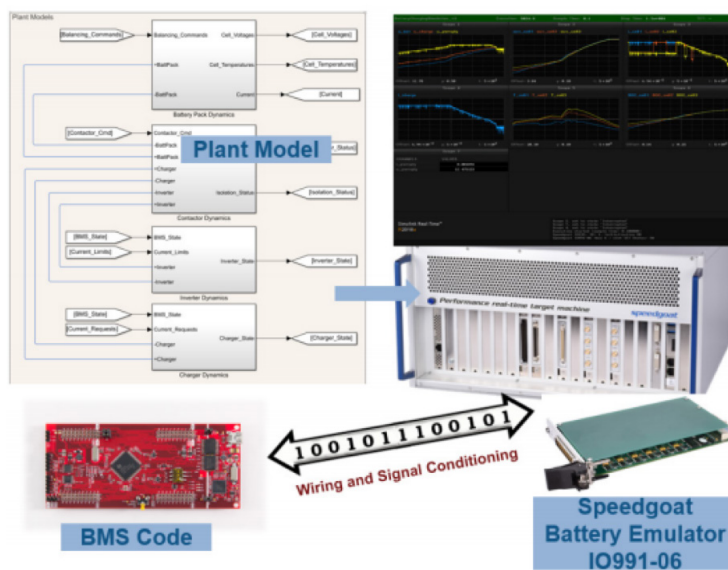


図6 バッテリーマネジメントシステムソフトウェアのHIL (Hardware-in-the-Loop) テスト。SimulinkでBMSアルゴリズムを作成し、Texas Instruments C2000マイクロコントローラーに実装。プラントモデル (バッテリーパック、コンタクター、インバーター、充電器) はSimulinkでモデリング。コード生成後、バッテリーエミュレーター付のSpeedgoatリアルタイムマシンに実装して実行。

Simulinkのデスクトップシミュレーションモデルを再利用し、リアルタイムシミュレーション用のコードを生成することで、全体的な開発期間を短縮できます。実時間での実行に最適化されたC/C++、HDLコードを生成することができ、リアルタイムシミュレーション用にSimulinkモデルから生成されたコードには、リアルタイムシミュレーションの実行中に制御パラメーターを調整できるインターフェースが含まれています。

ラピッドプロトタイピングの実行

ハードウェアテスト中に、コントローラーのコードを変更すると、遅延や追加のリスクが生じます。手作業でコードを変更し、再コンパイルし、マイクロコントローラーまたはFPGAに実装するのは時間がかかります。制御アルゴリズム開発者とソフトウェア、ハードウェア開発者が異なる場合には、さらに時間がかかる可能性が高まります。変更の程度にもよりますが、実装したコードに新しい問題が発生するリスクもあります。

Simulinkを使えば、手作業でコントローラーを更新する必要はありません。専用のコンピュータ上でリアルタイムに実行し、高速I/Oでテストハードウェアと通信するコードを生成することができます。コーディングの手作業や、それに伴う遅れを回避できるだけでなく、RPの利点は、BMSソフトウェアの変更をまずデスクトップでシミュレーションし、問題がないことを検証できることです。

HIL (ハードウェアインザループ) によるテスト

バッテリーシステムのハードウェアプロトタイプは、構築や修正にかかる労力や、修理費用が高額になることから、バッテリーパック電気システムでは使用できない場合があります。この場合、わずかな設計変更が開発スケジュールに大きく影響する可能性があるため、開発チームは大きな変更を避けがちになり、BMS開発がゆっくりと進行する傾向があります。

Simulinkを使用すると、バッテリーと、電源や負荷などの上位システムを含めたモデルからC/C++、HDLコードを生成できます。このコードをリアルタイムコンピュータに実装すれば、ハードウェアプロトタイプでテストする前に、コントローラーのコードをハードウェア上でリアルタイムシミュレーションを実行でき、高価で、交換や修理が難しいハードウェアプロトタイプに損傷を与えることなく、制御設計のエラーを発見、修正することができます。また、不適切なコンポーネントのサイズ変更など、ハードウェア設計のエラーを発見することもできます。

Speedgoatターゲットハードウェアなどの多くのHILリアルタイムシステムには、バッテリーエミュレーターが組み込まれており、ポータブルバッテリー電源や電気自動車用のバッテリースタックをエミュレートしたり、電流を下げて充電中のバッテリーをシミュレーションしたりすることができます。

“Speedgoatは、Simulink用の簡単に使えるリアルタイムプラットフォームを開発したと感じています。おかげで、制御システムのテストにより時間を費やし、HILベンチを開発する時間を短縮できるようになりました。”

— Proterra、制御エンジニア、Joaquin Reyes

» [ユーザー事例を読む](#)

HIL (Hardware-in-the-Loop) によるテストをさらに詳しく

- [バッテリーマネジメントシステム用Speedgoatリアルタイムソリューション - 概要](#)

量産用コード生成

Simulinkのコントローラーモデルから、量産マイクロコントローラー、FPGA、ASIC実装用の、読みやすく、コンパクトで効率的なC/C++、HDLコードを生成することができます。RP用コードとは異なり、量産用に生成されたコードは、リアルタイムモニタリング、パラメーターチューニング、データロギングに必要な追加のインターフェースを含みません。最適化設定を使用すると、生成された機能、ファイル、データを正確に制御ことができ、コードの効率を高め、レガシーコード、データタイプ、校正用パラメーターとの統合を促進することができます。

PIL (プロセッサインザループ) シミュレーションの実行

プロセッサインザループ (PIL) シミュレーションでは、デバイスがBMSハードウェアのSimulinkモデルをステップ実行している間に、C/C++、HDLコードがマイクロコントローラ、FPGA上で実行されるので、BMSコードの初期評価段階でプロトタイプが損傷するリスクを抑えることができます。PILシミュレーションはリアルタイムでは実行されませんが、ビットトゥルーであり、多様な環境下での検証を行うことができ、実システムに実装した時に正しく動作すると確信を得ることができます。

量産コードの生成

デスクトップシミュレーション、RP、HIL、PILシミュレーションすべてを通じて、BMSの制御アルゴリズムを検証します。Simulinkではこうして作成したアルゴリズムをもとに、マイクロコントローラ実装用の最適化された安定したC/C++や、FPGAプログラミングやASIC実装用の合成可能なHDLコードを生成します。自動コード生成機能により、アルゴリズムを手動で変換する際の人的エラーがなくなり、Simulinkで検証済みのアルゴリズムと数値的に等価であるC/C++、HDLコードが生成されます。可能なすべての動作条件、故障条件で制御アルゴリズムをシミュレーションできるので、実際にすべてのケースのテストをできなくても、生成コードが実システム上でどう対処するか確信を持つことができます。後工程のハードウェアテストでアルゴリズム修正が必要になった際には、モデルのアルゴリズムを変更し、テストケースを再シミュレーションし、変更点の正しさを確認し、更新されたコードを生成します。生成されたC/C++、HDLコードは完全に移植可能であり、さまざまなオプションで最適化でき、またSimulinkモデルとの双方向トラッキングが可能です (図7)。

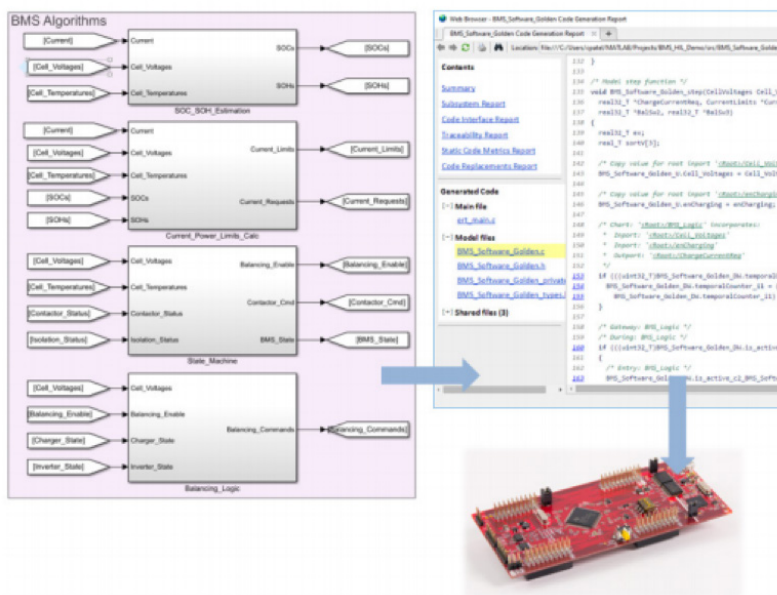


図7 Simulinkでモデル化されたBMSアルゴリズムからBMS量産コードを自動生成します。コードはTexas Instruments C2000マイクロコントローラに実装されています。

“実際の回路で検証する前に、認証に必要な要件をすべてSimulinkでシミュレーションして確認します。Embedded Coderでモデルから直接コードを生成しているので、シミュレーションと実際の組み込みソフトウェアの間に差異はありません。”

— 村田製作所、Yue Ma博士

» ユーザー事例を読む

量産コードの生成についてさらに詳しく

- [村田製作所がモデルベース デザインによりエネルギー管理システム制御ソフトの開発期間を 50%以上 短縮 - ユーザー事例](#)
- [モデルベースデザインによるハイブリッド自動車用バッテリーマネジメントシステム向けAUTOSAR、ISO 26262準拠のソフトウェアの開発 - 記事](#)

今すぐ始める

バッテリーマネジメントシステムのプロジェクトをスピードアップする方法はこちら。

[Simulinkでのバッテリーマネジメントシステム - 概要](#)

[モーターおよび電力制御用の評価版ソフトウェア - ダウンロード](#)

[バッテリーのシミュレーションと制御 - 技術コンサルティング](#)