

電子制御ユニットの検証ソリューション：EPSテストシステムの構成例 CANバス通信とアナログ信号の同期の重要性

CAN (Controller Area Network) デバイス用のデザイン検証・製造・メンテナンスシステムを構築する際には、CANバスインタフェース (単一・複数) とCANデバイス間での通信について考慮するのはもとより、CANおよび周辺機器とのデータ集録I/O (アナログ・デジタル・カウンタ/タイマ) についても検討する必要があります。CANバス通信のデータ転送とアナログ・デジタルデータの入出力のタイミングが異なれば (時間軸に対して一致していなかったり、ズレやシフトが生じるような場合)、データの信頼性は損なわれ、システム自体の確度も落ちてしまいます。ナショナルインスツルメンツ (以下、NI) が提供する PXI/PCIプラットフォームは、複数のバスインタフェースとデータ集録デバイス (アナログ/デジタル・タイミング計測デバイスなど) 間でCANメッセージの同期をとることのできる特殊な機能を提供し、テスト・計測を行う上で大いに役立ちます。また、CANバス通信とデータ集録を完全に統合させたソリューションとして機能します。この点がPXI/PCIが他のテストソリューションと大きく異なる点です。

本アプリケーションノートではEPSユニットのテストシステム構成を例に挙げ、テストシステム検証アプリケーションの構築案を紹介します。このアプリケーションは同時計測テストシステムで、NI-CANのチャンネルAPI (アプリケーションプログラミングインタフェース) を使用したCANメッセージの通信とデータ集録 (DAQ) 計測とを同期させています。ここで集録されたデータを相互比較し、EPSユニットの電子制御装置 (ECU) が適切なCANメッセージを生成しているかどうか、DAQ計測を基に検証します。

目次

- 背景
- EPSとは？
- テストシステムにおけるI/Oハードウェアの同期の必要性
- EPSテストシステムのタイミング構成
- EPSテストシステム検証アプリケーション、LabVIEWサンプルコード
- CANチャンネルAPI、書き込み / 読み取りの種類
- リアルタイムシステムでの実装
- 結果
- まとめ

背景

自動車業界では、ここ数十年の間で環境、安全、ドライバーの利便性の追求など、様々な面での技術的発展により、多くの課題を解決できるテクノロジーが生まれました。排気ガスの低減、燃料効率の向上、アンチロックブレーキ制御システムなどの技術は、このような課題への取り組みによる成果ですが、これらの技術が普及するに伴い、自動車のデザイン用に様々な電子デバイスが開発され、組み入れられています。

従来、車内で使用される電子デバイスは単独で動作するものでしたが、搭載されるデバイスの数が増えるにつれ、接続しなければならない配線の数や取り付けるセンサの数も増大してきました。さらに、高性能化の追求から、複数のデバイスをシステムレベルで統合し、デ

バス間で複雑な通信が行われるようになりました。このような状況に対応すべく、1980年代半ばにRobert Bosch GmbH社がCANを開発し、その後ISOで国際標準となりました。現在、CANは自動車に使用されるデバイスを始め、航空電子工学を用いた航空機内デバイス、船舶・海洋アプリケーションで使用される電子機器、医療機器、エレベータなどの機械機器など、幅広く活用されています。

CANは2線式のマルチドロップ式シリアル通信バスです。バス型接続形態で、接続されたデバイスが相互に通信を行います。CANはいくつかの規格に分かれており、高速CAN（ISO11898）、低速CAN（ISO11519-2、フォールトトレラント式）、単線式などの種類があります。CANバスとNI-CANインタフェース製品に関する詳細はni.com/jp/canまたは、NI Developer Zoneをご覧ください（zone.ni.com（英語）にて検索キーワードに「NI-CAN」と入力するとご覧いただけます）。

CANデバイスを用いたテストアプリケーションが進化する中、テストシステムの構成要素（電圧・センサ測定、アクチュエータ制御、分散された様々な計測・制御ノードの統括・コントロールなど）の統合は必至です。そのためには、CANバス通信とこれらの様々なI/Oを同期させる技術が重要となってきます。

EPSとは？

従来の油圧式パワーステアリングシステムでは、エンジンベルトを使用し、ロータリーポンプ（ベーン・ピストン）を駆動していました。このポンプは、パワーステアリングギアやアクチュエータ内にあるピストンを動作させるためのオイルを加圧させて配給し、車輪の向きを変える際にドライバーを補助します。このユニットではステアリングシステムが使用されていない場合でも、絶えず小さな負荷がエンジンにかかっています。この技術を改良した結果誕生したのが、電動パワーステアリング（EPS）です。

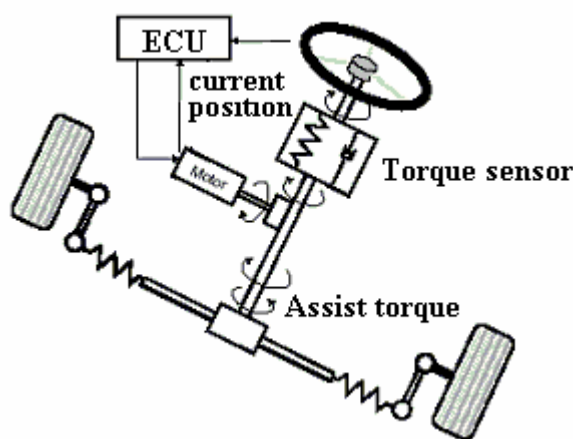


図1 電動パワーステアリング

EPS（図1）は、ギア機構によりステアリングコラムまたはラックに取り付けられた電子モータを使用しています。このプロセスでは、従来のパワーステアリングシステムで使用されていたポンプ、ホース、オイル、駆動ベルト、プーリーは不要になります。モータを駆動することで、ドライバーがステアリングに加える力を補助します。さらに、ECUではステアリン

グトルク、車両速度、モータの位置などのデータを集め、制御アルゴリズムを適用して、モータがステアリングに加える力を制御します。電動パワーステアリングシステムは従来の油圧ユニットに比べると軽量であることから、エンジン速度に関係なく、ステアリング補助機能が動作する場合にのみ動力が引き出されるため、燃料効率が向上します。

テストシステムにおけるI/Oハードウェアの同期の必要性

EPSユニットのテストを行うためには、ECUからのCANメッセージとセンサからの入力を同期させる必要があります。NIの提供するこれらのインタフェースには、特定のイベントが発生した場合のタイムスタンプ機能があるため、ループを使用して様々なハードウェア製品を「ソフトウェアで同期させる」プログラムを設計することができます。しかし、ソフトウェアによる同期を行う場合には、次の2点を考慮しなければなりません。

- 1) クロックのずれ
- 2) ソフトウェア遅延

これらは、ソフトウェアで同期を行う上で両インタフェースから得られるデータの相互関係を保つ、重要なポイントとなります。

クロックのずれ

クロック発振器には通常、ppm（百万分率）の単位で表される固有のジッタがあります。このため、時間の経過に伴い、同期が取れていない2つのクロックはずれていき、誤差が生じます。20 MHz \pm 100 ppmと定められている2台の発振器を使用した場合、2分以内に最大1ミリ秒の相対的なずれが生じることになります。また時間が経つにつれ、クロックのずれも大きくなります。

しかし、多くの製品検証アプリケーションでは長時間にわたってデバイスのテストを行うため、このクロックのずれが許されない場合があります。DAQとCANボード上で2つのクロックがずれてしまった場合、各クロックから集録されたサンプルが相互に関連していることを保証することはできません。この問題は、RTSIバスを用いて同期をとることにより、DAQボードから再同期パルスを発生させて100ミリ秒ごとに2つのクロックの再同期を行うことで解決できます。この方法を利用すれば、2つのボードのタイムベースに数 μ 秒以上（上記のスペックでは最大10 μ 秒）の差が生じることはなく、テストシステムでのクロックのずれのを最小限に抑えることができます。

ソフトウェア遅延

ソフトウェア同期を行う場合、各処理を開始したり停止したりするための制御を行うことでソフトウェア遅延が生じます。各処理はソフトウェア上で関数コールが実行され、プログラム中で一つ一つ順を追って実行されなければならないため、データ集録の開始処理とCAN集録の開始処理の間に幾分かの遅延が生じます。この遅延により、データ集録とCANのそれぞれが0番目のサンプルとみなすもの間に予測できない量の誤差が生じます。結果として、データ集録とCANデータを時間軸で相互に関連づけることができず、別途データ操作を行わない限り、両方のデータを適切に表示させることはできません。

EPSテストシステムのタイミング構成

以下に、2つのCANチャンネルと2つのアナログ入力(AI)チャンネルからの入力信号を同期させ、波形集録を行うテストシステムの例を示します。両チャンネルは特定のサンプリングレートでサンプリングするものとします。

このEPSテストシステムでは、ECUはトルク、PWM電圧、車速のような信号を集録し、制御アルゴリズムを介して、モータを制御するCANメッセージの出力を調整します（図2）。EPSの制御アルゴリズムを検証するために、自動車のテストエンジニアはCANメッセージとDAQの集録データを同時に記録して解析し、期待するモータ出力が生成されているかどうか判断します。

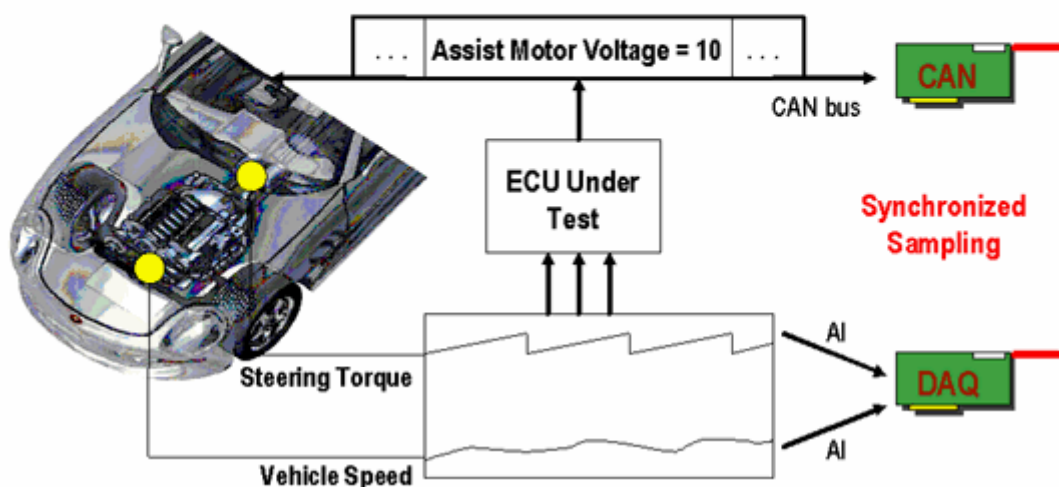


図2：上図では、EPSテストシステム内のデータ集録システムにNI PXI-6070Eを配置し、開発中のECUに送られるトルク・電圧・車速などのデータをサンプルしています。このシステムにはNI PXI-846x CANモジュールも搭載され、ECUから出力されるモータ制御信号のサンプリングを行います。

EPSテストシステム検証アプリケーション、LabVIEWサンプルコード

基本プログラミングモデル

CAN/DAQ同期アプリケーションでは、NI CANインタフェースを制御するためにNI-CAN 2.0 APIを使用しています。NI-CANは、NI-DAQ APIとよく似たAPIです。NI-DAQ APIについてはご存知の方も多いかもしれませんが、基本プログラミングモデルは以下の手順となります。

1. 設定
2. 読み取り/書き込み
3. クリア

この基本的なプログラミングモデルを使用すると、6種類の基本的なLabVIEW VIを使ってCANデータとDAQデータを容易に関連づけることができます。CANチャンネルはLabVIEWで使用する前に、Measurement & Automation Explorerで作成するか、あるいはVector社のデータ

ベースファイルからインポートする必要があります。

CANチャネルのインポート

NI-CANドライバは、標準でVector社のデータベースファイルを取り込むことができます。既存のCANチャネルがある場合には、以下の手順でVector社のデータベースファイル(.dbc)からCANチャネル情報を直接インポートすることができます。チャネルをインポートするには、**Measurement & Automation Explorer**を開いて**データ設定のCAN Channels**を右クリックし、**Import from CANdb File**を選択して下さい。この時開いたダイアログボックスには、現在利用できる定義済みのCANメッセージが全て表示されます。メッセージをクリックすると、そのメッセージ内で定義されているチャネルの中から、インポートしたいものを選択することができます。

チャネルの設定

CANチャネルを設定するには、**Measurement & Automation Explorer**を開いて**データ設定のCAN Channels**を右クリックし、**Create Message**を選択します。CANメッセージには、以下のパラメータを設定することができます。

- Name (名前)
- Arbitration ID (Hex) (CAN ID – Standard (11ビット) またはExtended (29ビット))
- Data Bytes (各メッセージのデータ長)
- Interface (CANインタフェース名)

上記情報を入力しメッセージを作成すると、作成したメッセージは**Measurement & Automation Explorer**の**CAN Channels**の下に表示されます。各CANメッセージには、いくつかのCANチャネルを設定することができます。CANチャネルを作成するには、作成したCANメッセージを右クリックして**Create Channel**を選択して下さい。設定を行うダイアログボックスは図3にあるとおりです。ここでは、スタートビットの位置やデータのビット長など、CANチャネルの情報を入力することができます。情報が入力されると、ウィンドウ右に表記されているマトリックスがグレーのボックスで埋まり、それらのビットがこのチャネルで使用されていることを示します。青いボックスは現在定義中のビットを示します。また、CANチャネルテストパネルを使用すれば各CANチャネルのI/Oをテストすることもできます。

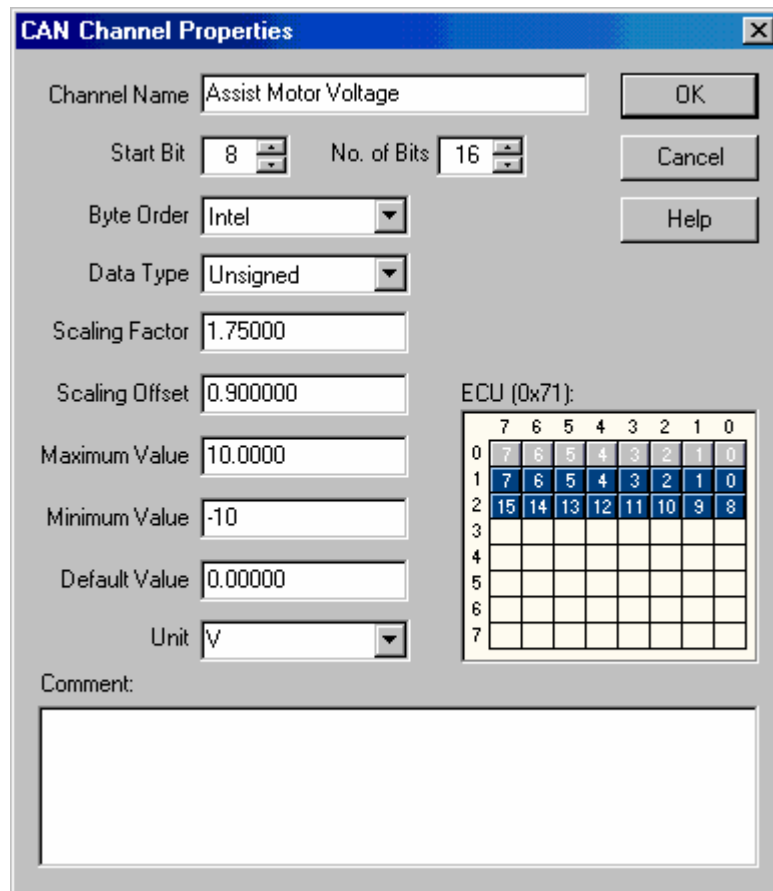


図3 : CANチャンネル設定ウィンドウ

CANチャンネルの設定またはインポートが終了すれば、LabVIEWでこれら設定したデバイスに容易にアクセスすることができます。図4に示されるCAN構成コードは、チャンネルAPIを使用してLabVIEWでCAN/DAQチャンネルを初期化するための基本的なプログラミングです。

このアプリケーションプログラムは3つの要素に大別できます。各処理について、もう少し詳しく解説します。図4はプログラムの最初の要素を示したものです。ここでは以下の構成要素について解説していきます。

- NI-CAN Init – 指定されたチャンネルリストのタスクを初期化します
- NI-DAQ Config – DAQチャンネルを初期化します
- NI-CAN Sync Start with NI-DAQ – CAN/DAQ同期用のトリガ信号やその他のトリガ信号をPXIトリガバスにルーティングを定義し、同期集録を開始します

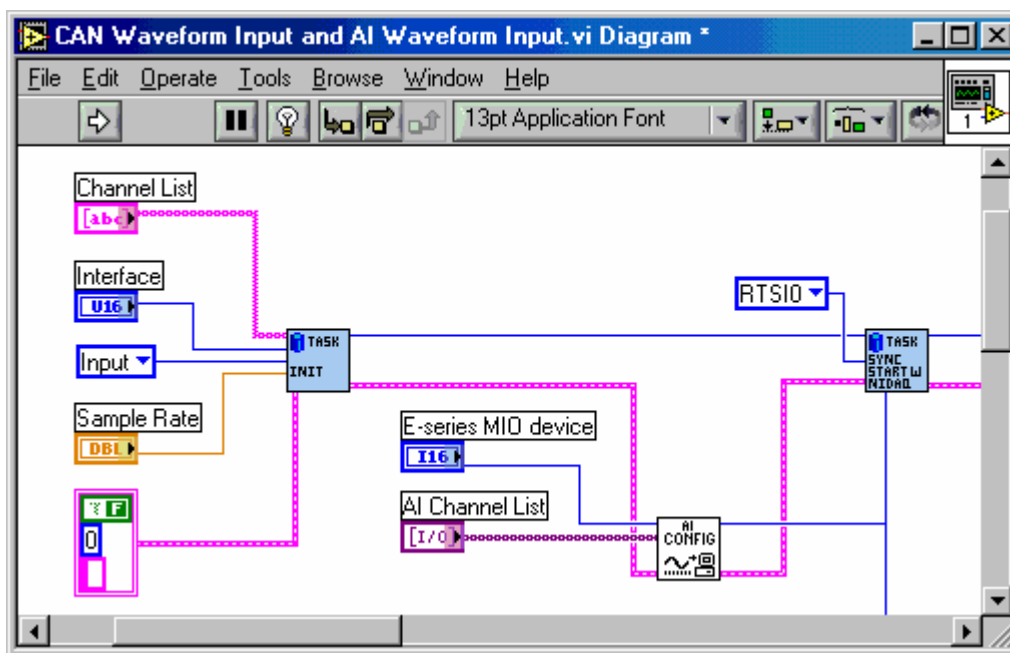


図4：CANとDAQインターフェースは初期化され、PXIトリガラインはタイミング/トリガリング信号を使用するように構成されます

- **Channel List** ではCAN通信が行われるCANチャンネルを指定します。
- **Interface** では使用するCANポートを指定します。
- **Input Mode** ではCANネットワークから読み込まれたチャンネルを指定します。この端子には、入力 (Input) 以外に他のオプションとして出力 (Output：書き込み用) とタイムスタンプ付きの入力があります。
- **Sample Rate** では1秒間に集録するサンプリング数を指定します。サンプリングレートが0の場合、シングルポイント集録が行われます。サンプリングレートが1以上の場合、有限連続集録が行われます。

NI-CAN チャンネルAPI、書き込み(write)/読み取り(read)の種類

- タイムスタンプ付 (読み込みのみ)
イベント駆動型
- Single Point(シングルポイントI/O)
最新メッセージを手早く参照
サンプリングレート端子= 0
- Continuous(連続I/O)
一定の時間内での連続したデータ列を参照
サンプリングレート端子 > 0

NI-CANのAPIには、3種類のReadモードと2種類のWriteモードがあります。Write/Read関数は、様々なアプリケーションの用途に対応できるよう用意されています。一方シングルポイントI/Oは、最新のCANメッセージにすばやくアクセスしたい場合に適しています。このシングルポイントモードでは、Write/Read関数が呼び出された時のみ、CANメッセージが送受信さ

れます。また、タイムスタンプ付きモードではCANメッセージがタイムスタンプとともに記録されるため、CAN/DAQの同期を必要とするテストアプリケーションに適しています。CANモジュールが（同期用パルスを使用して）DAQモジュールによって同期化されているため、DAQモジュールからのアナログ/デジタルデータとCANデータはタイムスタンプにて直接関連付けることができます。

DAQの連続I/OモードはDAQモジュールからのアナログ/デジタルデータとともにデータをロギングする場合に有用です。連続I/Oモードでは、新しいメッセージが受信されるまでCANの値がクロックパルスごとに再サンプリングされます。CANネットワークはイベント駆動型ですが、連続したCANデータが必要となる場合もあります。このような場合には、Write/Read関数設定時に連続I/O設定を行えば、クロックが反復するごとにCANデータが送信・受信されます。CANメッセージのI/Oよりもタイムベースの方が速いような場合には、新しいメッセージが入ってくるまでそのCANメッセージがコピー（再サンプリング）されます。

この機能は、自動車のCAN/DAQテストアプリケーションでは一般的ですが、DAQデータと関連するデータをロギングする場合にも有用です。この様なアプリケーションの場合、CANとDAQは同じタイムベースを使用します。さらに、連続I/Oモードで集録したデータはLV-Waveform-Data-Typeに変換することもできるため、データの管理も容易になります。

図5は、EPSテストシステム内のLabVIEWコードの一部で、PXIトリガバスによって同期されるCANとアナログ入力のパログラム方法を示しています。ここではサンプリングレートが1000に設定されていたため、CANメッセージは1秒間に1000回読み込まれます（連続I/Oモードの場合）。

- NI-CAN Read – CANタスクからサンプルを読み込みます
- AI Read – バッファリングされたデータ集録からデータを読み込みます
- Build Array – 複数の配列を結合します
- Align Waveform Timestamps – すべてのタイムスタンプの値（t0）を配列内の指定された指標の値と置換します
- Unbundle – エラー状態をチェックするために使用します
- NI-CAN Clear with NI-DAQ – CANとDAQのタスクをクリアします

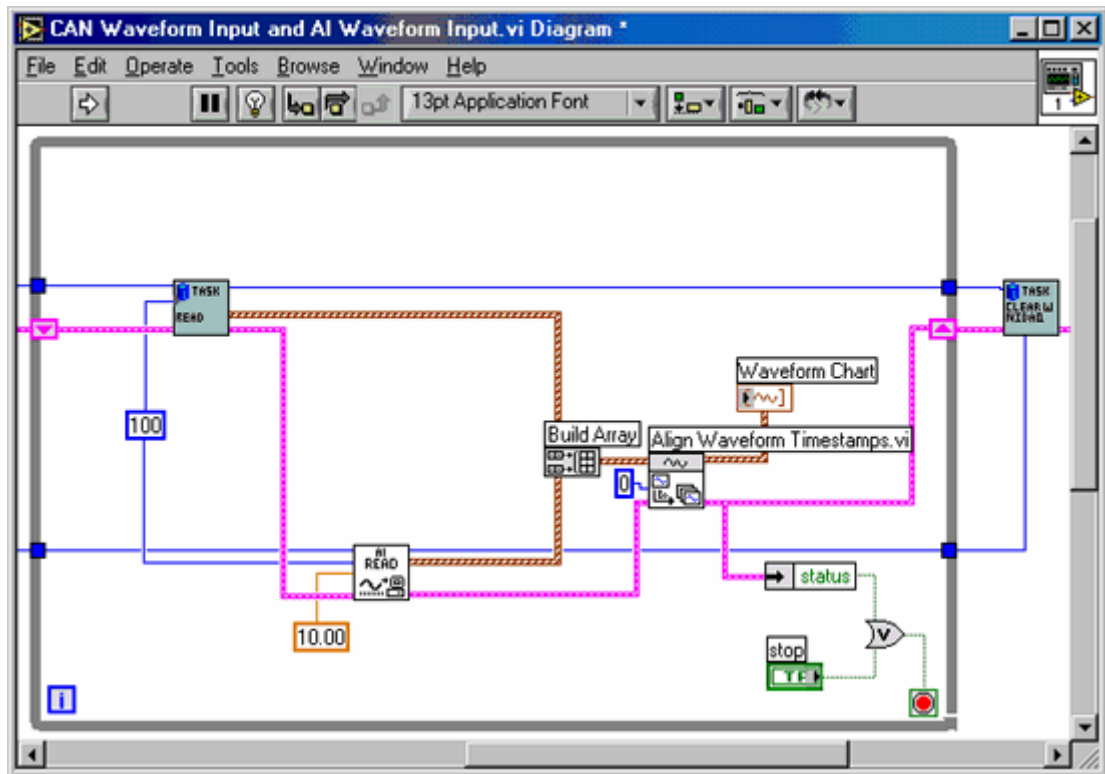


図5：CANとDAQの両信号が同期された状態で集録されます

リアルタイムシステムでの実装

上記で説明したEPSテストシステムは、テスト条件の変更によっては、より確定性の高い信頼できるテストシステムで実装する必要があります。NIのPXIシャーシにはPXIトリガバスが7本のトリガラインとともに組み込まれているため、データ集録、画像集録、モーションコントロール装置だけでなく、CANインタフェースモジュール間においても柔軟な同期関係を構築することが可能です。システム内のモジュールは、1つで使用する場合でも、複数で使用する場合でもPXIバックプレーンにタイミング信号を供給でき、システム内の各モジュールにこの信号を伝達し、I/Oの同期を行うことができます。タイミング機能はハードウェア内で実行されるため、システムが初期化されてしまえばアプリケーションソフトウェアはI/O同期に関与する必要もありません。また、PCIベースのシステムでは同一の同期機能をRTSIバスと組み合わせる利用できます。この場合、専用のリボンケーブルを使用して先端に取り付けられたコネクタによりPCIボード間を接続します。

加えて、PXIベースのシステムを構築するとLabVIEW RT(リアルタイム)に簡単に移行することができ、開発したテストシステムにリアルタイムの確定性を付加することが可能です。LabVIEW RTを始めとする、LabVIEWの製品情報はni.com/jp/labviewよりご覧いただけます。

結果

上記処理で集録されたCANデータは、アナログデータと容易に比較できます。DAQとCANの集録のタイミングが同期している場合、1つのグラフに2つの波形を一緒に表示させ、同一時間での相互関係を確認することができます。この作業は、同期していない場合ではデータ

の相関関係がないため不可能です。EPSテストシステムでは、ステアリングトルクが特定のしきい値に達した際、ECUが新しいrpm値をCANバスに送ります。ステアリングトルクはアナログ入力としてDAQハードウェアで集録され、rpm値はCANメッセージによって表されます。システムの動作を検証するには、アナログ入力のしきい値を超えた時点で正しいCANメッセージが転送されたことを調べる必要があります。このプロセスではDAQとCANハードウェアの両方から得られたデータが確実に関連付けられていることが要求されます。図6は、今回のシステムを使って集録された2つの信号データを比較した波形グラフを示しています。

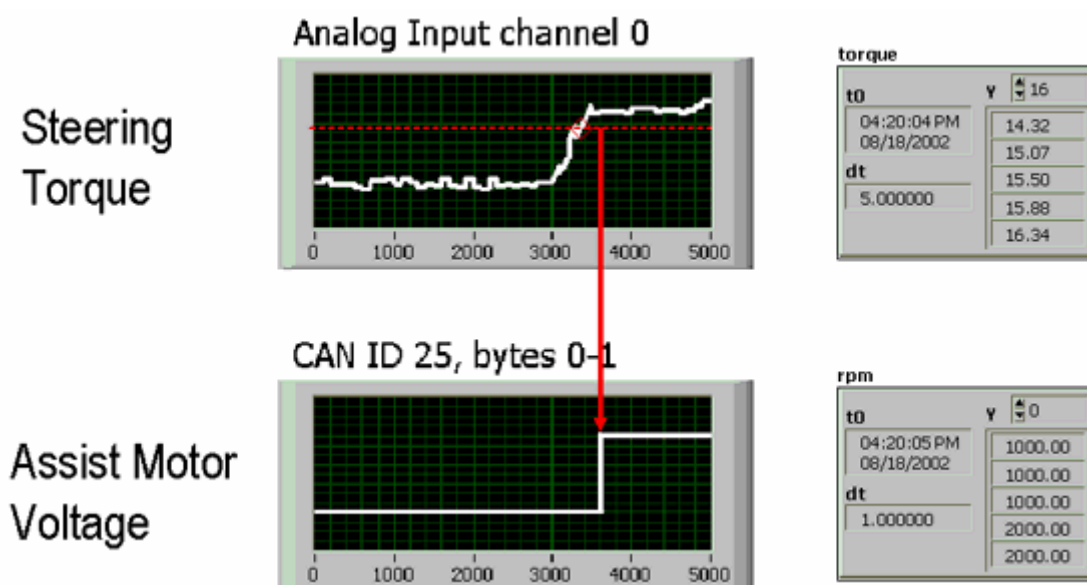


図6：同期の取れた異なるタイプのデータ

まとめ

NIの同期可能なテストシステムを使用すれば、CANデータとアナログデータを同時に集録することが可能となり、それによってPXIではPXIトリガバス、PCIではRTSIバスを使用してECUのシステム検証を実装することができます。NI-CANのチャンネルAPIをLabVIEWで使用すれば、6つのVIで同期化プログラムが完成できるというシンプルなプログラミングモデルを実現することも可能です。連続集録モードを使用すると、波形グラフでの比較を行うことができ、これにより、CANデバイスの検証を行いながら、それらデバイスが周辺、または自動車のユニット・システム全体の他のデバイスにどのような影響をもたらすかを確実に検証することができます。これもまた、自動車テストに必須のハードウェア・ソフトウェアテクノロジーとなり、これを同期可能なプラットフォームで行うことにより、刻々と変化するエンジニアのテストニーズに対応する優れたソリューションが実現します。