

RT Box *DEMO MODEL*

CAN channel loop-back demo on a single RT Box

1台のRT BoxでCANチャネルのループバックデモ

Last updated in RT Box Target Support Package 2.0.5

1 はじめに

コントローラエリアネットワーク(Controller Area Network: CAN)は、ホストコンピュータなしでマイクロコントローラやデバイスが相互に通信できるように設計された堅牢なビークルバスです。CANコネクタは、RT Box 1、2および3のハードウェアバージョン1.2で使用できます。コネクタの図とピン配置については、RT Box User ManualのRT Boxアーキテクチャの章を参照してください。コネクタには2つのCANチャネルがあり、それぞれにHigh (CAN_H)信号とLow (CAN_L)信号のペアがあります。このデモモデルは次の内容を示します。:

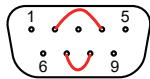
- 2つのCANチャネルを接続する単純なループバックシナリオ
- CANデータベース(.dbc)ファイルを使用してCAN PackおよびCAN Unpackブロックを構成する方法
- Can Receiveブロックの有効なポートを使用して、新しいデータの到着時に計算をトリガする方法

1.1 要求仕様

このデモモデルを実行するには、次の製品が必要です(www.plexim.comから入手可能):

- [PLECS RT Box](#)(最小ハードウェアバージョン1.2)を1台と[PLECS](#)および[PLECS Coder](#)ライセンス1つずつ
- [RT Box Target Support Package](#)
- [RT Box User Manual](#)のクイックスタートガイドに記載されている、PLECSとRT Boxの設定手順に従います
- チャンネル1と2のCAN_Hピンと、チャンネル1と2のCAN_Lピンを接続するワイヤ([図1](#)を参照してください。)

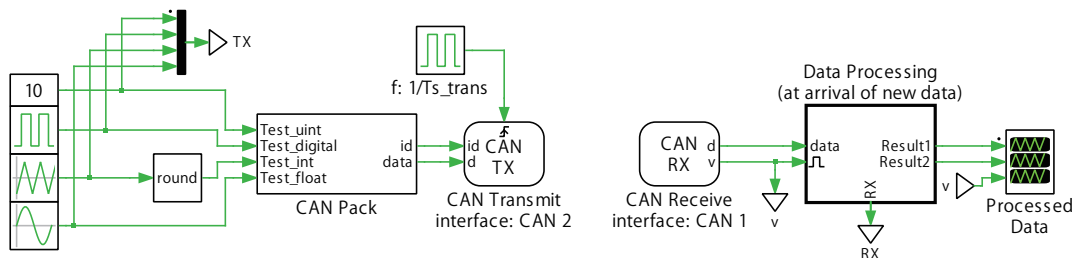
図1: RT Boxの背面パネルにあるCANチャネル1と2のループバック配線



2 モデル

このデモ モデルは、同じサブシステム内の2つの異なるチャネルでのCAN送信と受信機能を示します。[図1](#)のような配線設定により、2つのチャネル間でCANメッセージのループバックが形成されます。回路図を[図2](#)に示します。CAN Packブロックは、PLECS信号を入力としてCANメッセージを生成します。次に、CAN Transmitブロックと通信して CANメッセージを送信します。受信側では、CAN Receiveブロックの後に、CAN Unpackブロックを使用して、CANメッセージをPLECS信号にアンパックし、PLECSスコープで表示します。

図2: CANループバックデモのサブシステム回路図

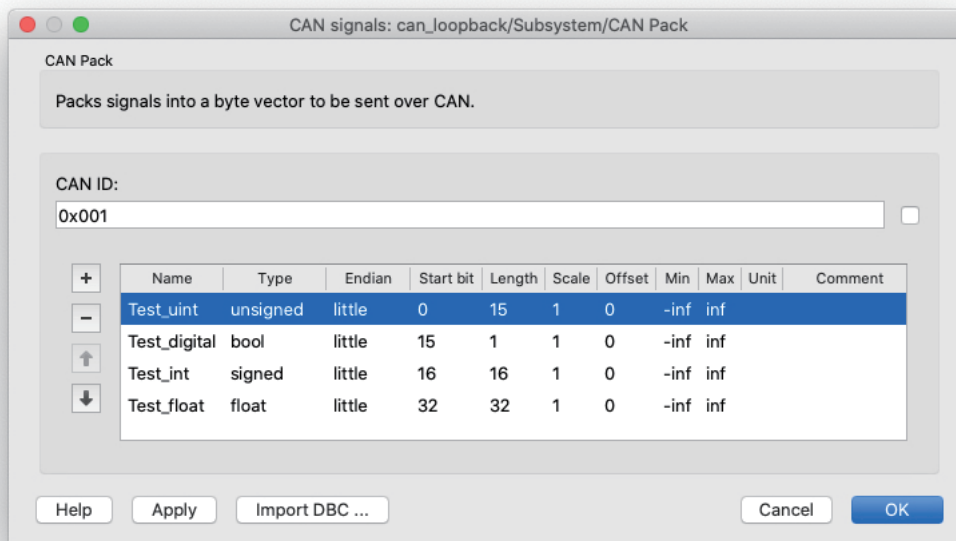


注意 CAN Packブロックは、入力される信号に対してデータのタイプキャストを強制します。したがって、供給信号の"出力データ型"とCAN Packブロック内で定義された信号データ型の間のデータ型の不一致を回避する必要があります。

2.1 CAN送信

CAN Packブロックは、入力信号をバイト列にパックしてCANメッセージを生成します(図3を参照)。**CAN ID**フィールドは、CANメッセージのIDを指定します。このデモでは、0x1に設定しています。CAN IDは、11ビット値(CAN 2.0A準拠の場合)または29ビット値(CAN 2.0B準拠の場合)のいずれかで提供します。生成されたCANメッセージでこのパラメータを使用するには、ブロックの**id**端子をCAN Transmitブロックの**id**端子に接続する必要があります。CANメッセージ内の信号は、そのデータ型、バイト順序(リトルエンディアン/ビッグエンディアン)、および64ビットCANメッセージ内の開始ビットと長さによって定義されます。信号に傾きと切片を適用して、浮動小数点信号を整数として効率的に送信できます。CANメッセージ内のビットには、0(最初のバイトの最下位ビット)から63(最後のバイトの最上位ビット)までの番号が付けられます。オプションで、**DBCファイルのインポート...**ボタンをクリックすると、信号定義をDBC(CANデータベース)ファイルからインポートできます。

図3: CAN Packブロックのマスクの内容



次に、このデモモデルに付属するcan_loopback.dbcファイルを参照して、**開く**をクリックします。CAN Packブロックに最初にリストされたものと同じCANメッセージ信号定義が表示されるはずです。**インポート**をクリックすると、DBCのインポートプロセスが完了します。

ほとんどのCANネットワークは、データのデコードに必要なDBCファイルをOEM(Original Equipment Manufacturer)のみが所有しているという意味で独自のものです。これは、たとえば、ほとんどの自動車、バイク、EV、生産機械などの生のCANデータが該当します。

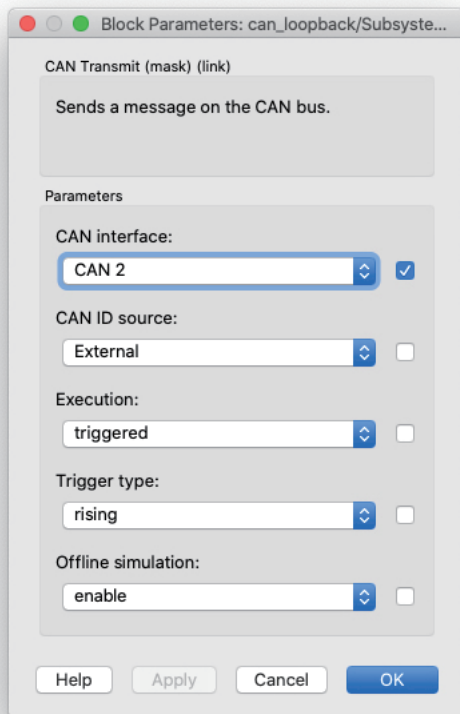
このデモモデルでは、次のものを送信します:

- 1 定数ブロックによって提供されるビット0からビット14までのunsigned整数型(15ビット長)
- 2 周波数1Hz、デューティー比0.5のパルス発生器からの15ビットのbool型デジタル信号(値は0と1の間で交互に変化)

- 3 三角波発生器から提供される-5から+5までの5Hzをビット16からビット31までのsigned整数型(16ビット長)を表現
- 4 振幅1の5Hz正弦波発生器(32ビット長)、ビット32からビット63までのfloat型

この情報は合計64ビットで保存され、これは1つのパッケージ内の最大データ量に相当します。最初のテスト信号の定数ブロックが、**Coder オプション...**ウィンドウの**パラメーターのインライン化**タブの**例外**フィールドに追加されていることに注意してください。つまり、モデルがリアルタイムで実行されると、外部モードでのリアルタイムシミュレーション中に値を調整できることになります。

図4: CAN Transmitブロックのマスクの内容



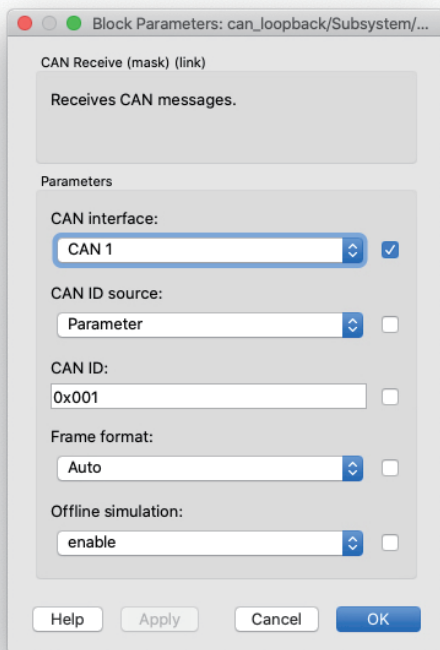
CAN Transmitブロックは、CANバス上でメッセージを送信します。図4はCAN Transmitブロックのマスク内容を示しています。

- CANチャネル2は送信チャネルとして使用するため、**CAN interface**フィールドから選択します。
- **CAN ID source**フィールドは、CAN Packブロックから入力信号として提供されるため、External(外部)を選択します。Parameter(パラメータ)を選択し、ここでCAN IDを指定することもできます。
- **Execution**フィールドは、regular(通常)またはtriggered(トリガ)を選択します。regularを選択した場合、次のフィールド**Sample time**で、定期的なパケット送信の期間を指定します。このデモでは、トリガ感度が上昇するtriggeredを使用します。CAN TransmitブロックのマスクレベルにTrigger Portが表示され、そこにはCAN送信サンプル時間(0.01秒)に等しい周波数のパルス発生器を接続しています。
- **Offline simulation**フィールドでは、オフラインシミュレーションでCANメッセージのシミュレーションを可能にするルート回路図上でTarget Inportsをenable(有効)またはdisable(無効)にします。

2.2 CAN受信

CAN ReceiveブロックはCANメッセージを受信します。CANメッセージを受信すると、データは8バイトのベクトル化された信号としてブロック出力`d`から出力されます。出力`v`は、新しい(有効な)データを受信する各シミュレーションステップでは1になり、それ以外の場合は0になります。[図5](#)はCAN Receiveブロックのマスク内容を示しています。

図5: CAN Receiveブロックのマスクの内容



- CANチャンネル1は受信チャンネルであり、CAN Receiveブロックの**CAN interface**パラメータフィールドから選択します。
- データのループバックを実現するには、CAN TransmitブロックとCAN Receiveブロックで同じ**CAN ID**を使用する必要があります。ご注意ください。
- **Frame format**をAuto(自動)に設定すると、指定したCAN IDが2047より小さい場合にBaseフォーマットを使用します。それ以外の場合、Extendedを使用します。
- **Offline simulation**フィールドでは、オフラインシミュレーションでCANメッセージのシミュレーションを可能にするルート回路図上でTarget Importsをenable(有効)またはdisable(無効)にします。

CAN Unpackブロックは、CAN経由で受信したバイト列からの信号を元のメッセージにデコードします。CAN IDと信号の定義は、[図3](#)に示すCAN Packブロックとまったく同じ方法で設定する必要があります。これは、CAN Unpackブロックの内容を手動で編集するか、または単に**DBCファイルのインポート...**を選択して、CAN Pack側で使用しているのと同じDBCファイルを指定することで実行できます。

3 シミュレーション

選択したCANインタフェースは、**Coderオプション**ダイアログで有効化および設定する必要があります。**システム**リストから、"Subsystem"を選択し、**ターゲット**タブに移動します。サブタブ**CAN**で、**Enable CAN1**と**Enable CAN2**の両方がチェックされていることを確認してください。

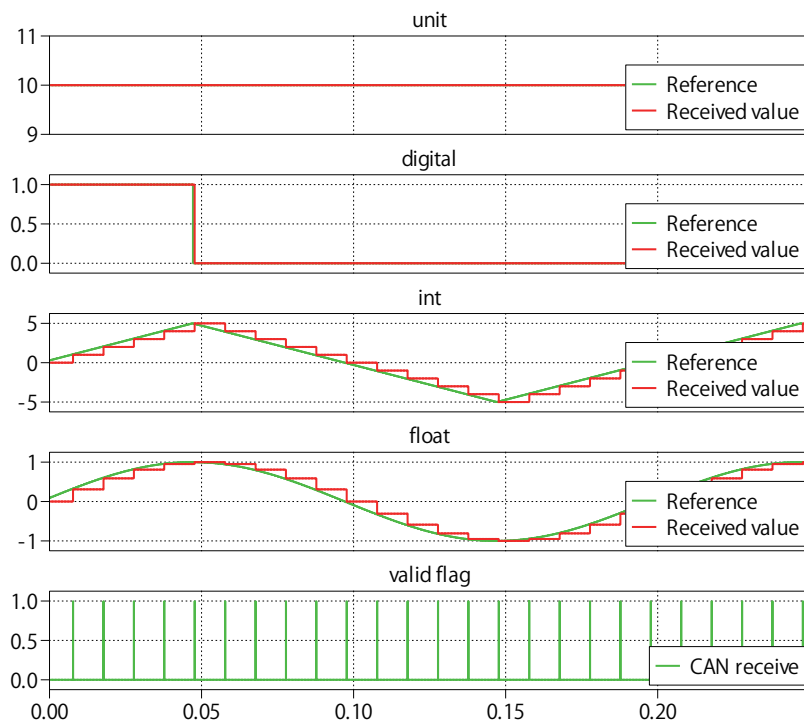
- **CAN1 baud rate** CAN送信のボーレートを指定します。CANバス上のすべてのデバイスは、同じボーレートを使用するように構成する必要があることに注意してください。このデモでは、デフォルト値の500000Hzを使用しています。
- **CAN1 tx/rx pin** CAN通信に使用するI/Oを指定します。Internal CAN interfaceオプションは、ハードウェアリビジョンが1.2以前のRT Box1を除くすべてのRT Boxバージョンで使用できます。古いリビジョンでは、デジタル入力および出力ピン26/27を使用してCANインタフェースを設定します。外部のCAN送受信チップはユーザが追加する必要があることに注意してください。
- 上記でInternal CAN interfaceを選択した場合、**Termination for CAN1**の新しいフィールドが表示されます。CANバスの両端には、CAN-HiとCAN-Low間に120Ωの終端抵抗が必要です。この例ではP2P接続について説明しているため、両方のCANインタフェースで終端抵抗を有効にする必要があります。

"RT Box"モデルを**ビルド**します。モデルがアップロードされたら、**Coder オプション...**ウィンドウの**外部モード**タブから、RT Boxに**接続し、自動トリガを有効化**します。リアルタイムシミュレーション結果を図6に示します。

- 最初の信号は値10の符号なし整数で受信します。実行中にCAN Packブロックの前の**定数**値を別の値(たとえば、10から20)に変更すると、図6のPLECSスコープの最初の受信信号に変更がすぐに反映されることがわかります。
- 2番目に受信する信号は、0と1を交互に繰り返す1Hz、0.5デューティ比のデジタルパルスです。
- 3番目の信号は、デューティ比が0.5で、-5から+5の間で変化するの5 Hzの三角波形です。
- 4番目の信号は、振幅が1の5Hz正弦波形です。
- 5番目の信号は、新しい有効なデータが到着したときにアサートされる有効フラグを示します。

最後の2つの信号では、量子化の10ミリ秒の階段効果がCAN Transmitブロックの**Sample time**設定を正しく反映していることもわかります。CAN Receiveブロックの"v"ポート信号は、新しいデータが到着した場合にのみHighになります。この信号は、triggered/enabledサブシステムに実装された、潜在的に時間のかかる計算をトリガするために使用できます。このデモでは、最初の2つの信号を加算し、最後の2つの信号を乗算する単純な設定を使用します。

図6: CANループバックテスト後の4つの受信信号を示すリアルタイム波形



4 まとめ

この RT Boxデモモデルは、RT Boxに統合されたCANインタフェースを使用したループバック テストを示します。Coder設定でCANインタフェースを設定する方法と、CAN PackおよびUnpackブロックを使用してPLECSモデルでCANメッセージが形成される方法を示します。デモモデルは、オフラインとリアルタイムシミュレーションの両方で実行できます。

改訂履歴:

RT Box Target Support Package 2.0.5 初版



Pleximへの連絡方法:

☎ +41 44 533 51 00

Phone

+41 44 533 51 01

Fax

✉ Plexim GmbH

Mail

Technoparkstrasse 1

8005 Zurich

Switzerland

@ info@plexim.com

Email

<http://www.plexim.com>

Web



アドバンオートメーションへの連絡方法:

☎ +81 3 5282 7047

Phone

+81 3 6285 0250

Fax

✉ ADVAN AUTOMATION CO.,LTD Mail

1-9-5 Uchikanda, Chiyoda-ku

Tokyo, 101-0047

Japan

@ info-advan@adv-auto.co.jp

Email

<https://adv-auto.co.jp/>

Web

RT Box Demo Model

© 2002–2025 by Plexim GmbH

このマニュアルで記載されているソフトウェアPLECSは、ライセンス契約に基づいて提供されています。ソフトウェアは、ライセンス契約の条件の下でのみ使用またはコピーできます。Plexim GmbHの事前の書面による同意なしに、このマニュアルのいかなる部分も、いかなる形式でもコピーまたは複製することはできません。

PLECSはPlexim GmbHの登録商標です。MATLAB、Simulink、およびSimulink Coderは、The MathWorks、Inc.の登録商標です。その他の製品名またはブランド名は、それぞれの所有者の商標または登録商標です。