

# Embedded Code Generation *DEMO MODEL*

## Simple CAN Model

シンプルなCANモデル

- TI C2000 TSPを使用したCAN通信を診断するシンプルなモデル -

Last updated in C2000 TSP 1.9.1

# 1 はじめに

このデモでは、PLECS Coderと TI C2000 Target Support Packageを使用して、Texas Instruments (TI) C2000 マイクロコントローラ(Microcontroller: MCU)上のController Area Network(CAN)ブロックを使用するシンプルなモデルを紹介します。

CANバスは、ホストコンピュータなしでマイクロコントローラとデバイスが相互に通信できるように設計された堅牢な自動車用の規格です。

モデルは、[図1](#)のように、7つの異なるMCUターゲットに対応する14個のサブシステムに分かれています。各サブシステムは、対応するTI C2000 LaunchPadハードウェアに個別にデプロイできます。次章以降に、モデルの簡単な説明と、それをシミュレートする方法について説明します。

## 要件

各LaunchPadターゲットには、トランシーバが統合されたCANインターフェースが1つあります。したがって、CAN通信を診断するには、[第2章](#)で説明しているように、このモデルでは任意の2つのLaunchPadターゲットが必要です。

# 2 モデル

最上位レベルの回路図には、[図1](#)に示すように、7つの異なるC2000 ターゲットに対応する14個の独立したサブシステムが含まれています。拡張子が"tx"のサブシステムはCANメッセージを送信し、"rx"のサブシステムはCANメッセージを受信するように構成されています。"28069"というラベルのサブシステムはTI 28069 LaunchPad [\[3\]](#)用、"280049"はTI 280049C LaunchPad [\[2\]](#)用、"28377S"はTI 28377S LaunchPad [\[4\]](#)用、"28379D"はTI 28379D LaunchPad [\[5\]](#)用、"280039"はTI 280039C LaunchPad [\[1\]](#)用、"28P550SJ9"はTI 28P550SJ9 LaunchPad [\[6\]](#)用、"28P650DK9"はTI 28P650DK9 LaunchPad [\[7\]](#)用に構成されています。

各サブシステムは、サブシステムブロックの太い外枠で示されるように、コード生成が有効になっています。この手順は、PLECS Coderを介してサブシステムのモデル コードを生成するために必要です。この設定は、サブシステムを選択し、**編集メニュー -> サブシステム -> 実行の設定...**から**コード生成機能の有効化**オプションを選択することによって構成されます。

生成されたコードは、MCUの基本サンプル時間または**離散化ステップサイズ**で実行されます。このモデルでは、各サブシステムの離散化ステップサイズは100  $\mu$ sに設定されています。

各LaunchPadターゲットは、それぞれ[図2](#)および[図3](#)に示すように、CANメッセージを送信または受信するように構成されています。CAN通信の診断には、任意の2つのターゲットを選択します。1つはCANメッセージを送信し、もう1つは受信します。

## CAN Port

[図2](#)および[図3](#)に示すCAN Portブロックは、CAN通信ポートを設定します。

入力**en**はCANポートの状態を決定します。**en**を0に設定するとCANポートは強制的に**bus-off**状態になり、1に設定するとCANポートは**bus-on**状態に移行します。Auto bus-onが有効(Enabled)になっていない場合、有効信号を0に設定してから1に戻すことで、**bus-off**状態をクリアする必要があります。CANエラーモードの詳細な説明は、このブロックの"ヘルプ"に記載されています。

## Simple CAN Model

図1: 7つのサブシステムを持つモデルのトップレベルの回路図

Simple CAN model with seven distinct subsystems  
Each subsystem can be independently deployed to the corresponding TI C2000 LaunchPad target

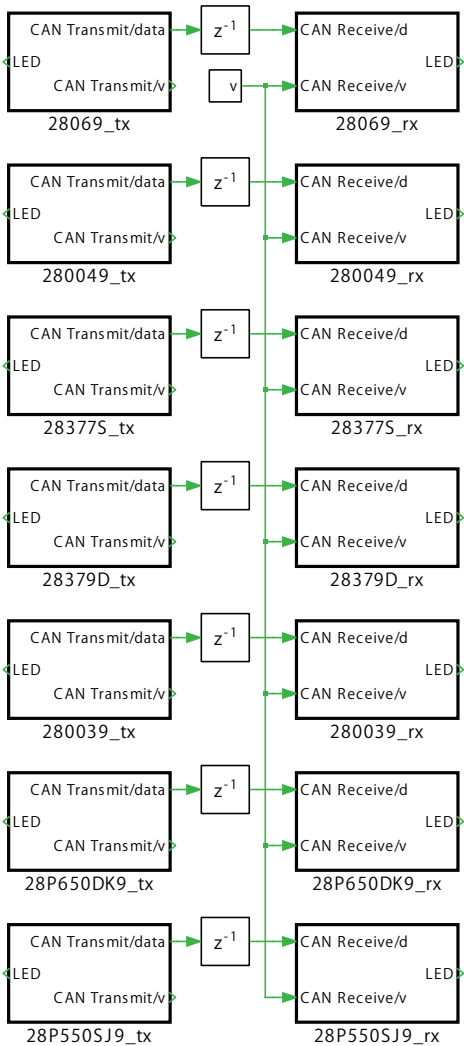
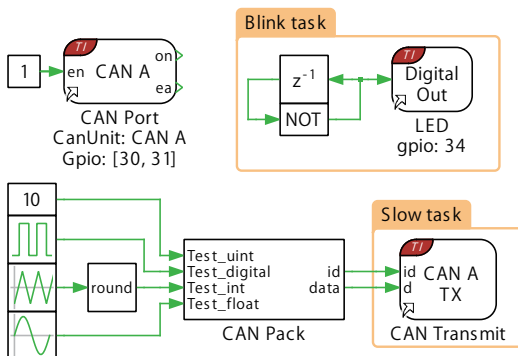


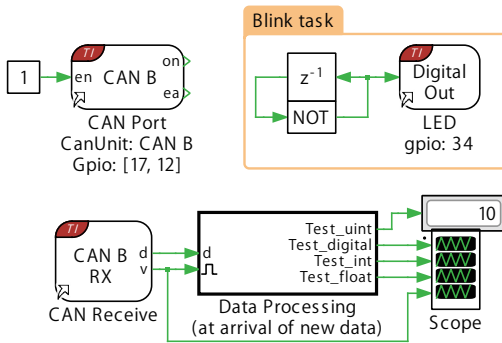
図2: CANメッセージを送信するための28069\_txサブシステムの回路図



## CAN送信

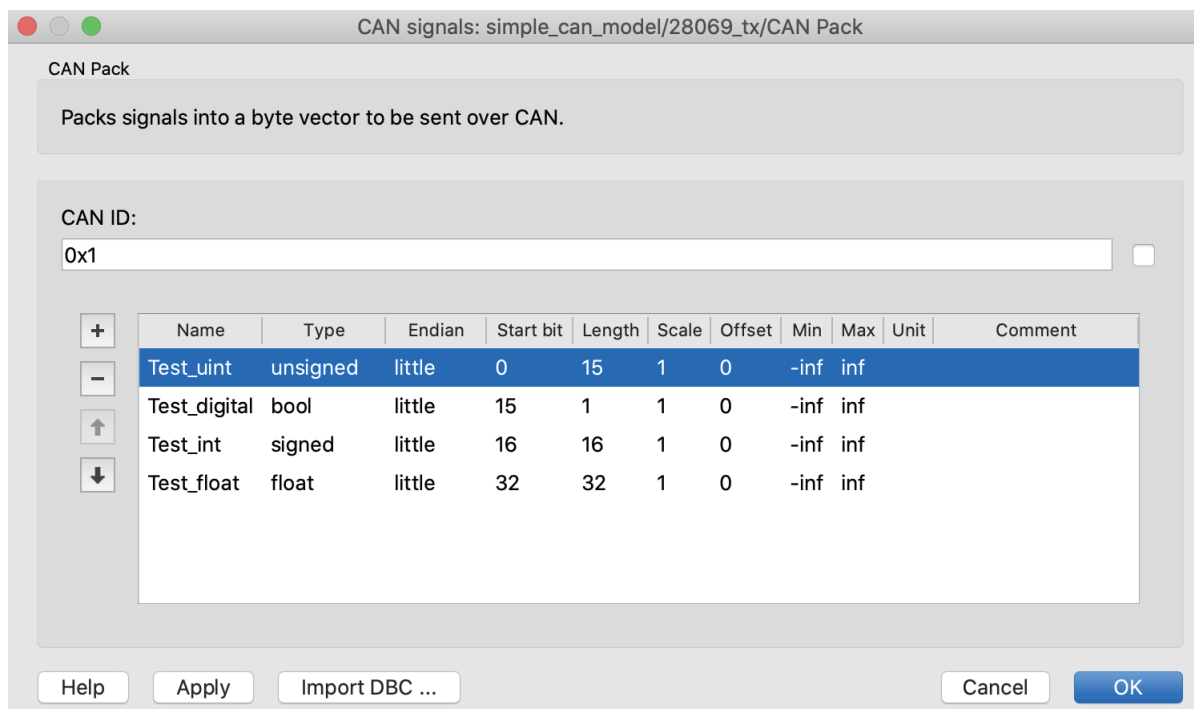
CAN Transmitブロックは、CANバス上でデータを送信します。送信するデータは、データ型**uint8**のベクトル化された信号としてブロック入力**d**に指定する必要があります。送信されるCANメッセージの長さは、入力信号の幅(1から8バイト)によって決まります。

図3: CANメッセージを受信する28379D\_rxサブシステムの回路図



CAN Packブロックは、図4に示すように、入力信号をバイトベクトルにパックしてCANメッセージを生成します。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に設定されています。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)のいずれかで指定します。CANメッセージ内の信号は、そのデータ型、バイト順序(Little Endian/Big Endian)、および 64ビットCANメッセージ内の開始ビットと長さで定義します。信号にスケールとオフセットを適用し、浮動小数点信号を整数として効率よく送信できます。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。

図4: CAN Packブロックのパラメータ画面



**DBCファイルのインポート...** ボタンをクリックすると、DBC(CANデータベース)ファイルから信号定義をインポートするオプションがあります。多くの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。これは例えば、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、以下を送信します:

- 1 定数ブロックから提供されるビット0からビット14までのinteger型(15ビット長)

- 2 1 Hz、0.5デューティー比のパルス信号から提供される、ビット15のbool型デジタル信号。値は0と1の間で切り替わります
- 3 -5から+5までの5Hz三角波をビット16からビット31までのsigned型で表したものの(16ビット長)
- 4 ビット32からビット63までのfloat型(32ビット長)で、振幅1の5Hz正弦波

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ることに注意してください。つまり、モデルが組み込みターゲット上で実行されると、外部モードを介して値をリアルタイムでオンザフライで調整できるようになります。

CAN送信の詳細については、CAN TransmitおよびCAN Packブロックの説明の"ヘルプ"ボタンをクリックしてください。

### CAN受信

CAN Receiveブロックは、指定されたCANインタフェース上で指定された識別子(ID)を持つCANメッセージの受信を開始します。CANメッセージを受信すると、データは指定されたフレーム長のベクトル化された信号としてブロック出力**d**から利用可能です。新しいデータを受信した場合、出力**v**は1シミュレーションステップに対して1、それ以外の場合は0です。

CAN Unpackブロックは、CAN経由で受信したバイトベクトル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4](#)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。

CAN受信の詳細については、CAN ReceiveおよびCAN Unpackブロックの説明の"ヘルプ"ボタンをクリックしてください。

### マルチタスクコード

PLECS CoderとTI C2000 Target Support Packageを使用すると、ユーザはTI C2000ファミリのMCU用のマルチタスクコードを生成できます。マルチタスク コードは、さまざまな時間スケールでダイナミクスで複数のシステム出力を制御するための処理能力を解放します。このモデルでは、CAN Transmitブロックは低速で実行できるため、マルチタスクコードが使用されます。

マルチタスクコード生成は、**Coder** -> **Coderオプション...**ダイアログの**タスク**タブから構成します。**タスクモード**をマルチタスクに変更し、**タスク設定**を指定することにより、各タスクのサンプル時間を設定できます。基本サンプル時間は常に**離散化ステップサイズ**と同じになります。優先度の低いタスクの**サンプリング時間**設定は、基本サンプル時間の整数倍である必要があります。異なる速度で実行する、低速で優先度の低いタスクを最大15個指定できるため、アプリケーション内で最も高速で優先度の高いタスクのプロセッサ時間を確保できます。詳細については、PLECS User Manual[\[9\]](#)の"コード生成"を参照してください。

PLECS 回路図内のブロックは、タスクライブラリコンポーネントを使用して、優先度の低いタスクに割り当てられます。このモデルでは、[図2](#)に示すように、Base taskに加えて、優先度の低い2つのタスクが定義されています。Base taskは100  $\mu$ sで実行され、Slow taskは0.01sで実行され、Blink taskは0.5sで実行されます。

## 3 シミュレーション

各サブシステムは、対応する TI LaunchPadハードウェアのターゲット固有のコードに直接変換できます。

---

**注意** 作業を進める前に、LaunchPadデバイスのDIPスイッチの位置とジャンプが正しく設定されていることを確認してください。各LaunchPadデバイスのガイドは、TI C2000 Target Support User Manual[8]の"C2000開発キットのプログラミングに関するヒント"セクションに記載されています。

---

### ハードウェアの接続

各LaunchPadターゲットには、トランシーバが統合されたCANインタフェースが1つあります。CAN端子は、TI 28069、TI 28377S、およびTI 28379DはコネクタJ12を介して利用でき、TI 280049、TI 280039、TI 28P550SJ9、およびTI 28P650DK9では、コネクタJ14を介して利用できます。

必要に応じて、2つのLaunchPadターゲットを選択します。次にジャンプを使用して、2つのCANインタフェースのCAN\_H、CAN\_L、およびGNDピンを接続します。

### MCUに書き込み

サブシステムをTI MCUにアップロードするには、以下の手順に従います。

- 目的のMCUをUSBケーブルを介してホストコンピュータに接続します。
- **Coder** -> **Coderオプション...**ウィンドウの**システム**リストから、目的のMCUを選択します。
- **ターゲット**タブで、ドロップダウンメニューから適切なターゲットを選択します。次に、**General**サブタブで、目的の**Build type**を選択します。
- PLECSから直接MCUをビルドしてプログラムするには、**Build type**でBuild and programを選択し**Build configuration**でRun from FlashまたはRun from RAMのいずれかを選択し、**Board**でLaunchPadを選択して、**ビルド**をクリックします。

---


**注意** 正しくプログラムされていれば、LaunchPadボード上のLEDが点滅します。

---

Code Composer Studio (CCS)に精通している上級ユーザ向けには、Generate code into CCS projectオプションがあります。TI C2000 Target Supportパッケージには、projectsというタイトルのフォルダが含まれています。フォルダ内には、各MCU用に事前に構築されたCCSプロジェクトを含むZIPアーカイブがあります。目的のターゲットに対応するzipアーカイブフォルダをCCSにインポートします。CCSワークスペースに新しいプロジェクトが作成されます。CCSプロジェクトの`workspace_loc}/dev_28xx/cg/`フォルダの場所を**CCS project directory**フィールドに入力し、**Build**をクリックします。次に、通常のCCSプロジェクトとしてプロジェクトをビルドしてデバッグします。詳細な手順については、TI C2000 Target Support User Manual[8]の"クイックスタート"を参照してください。

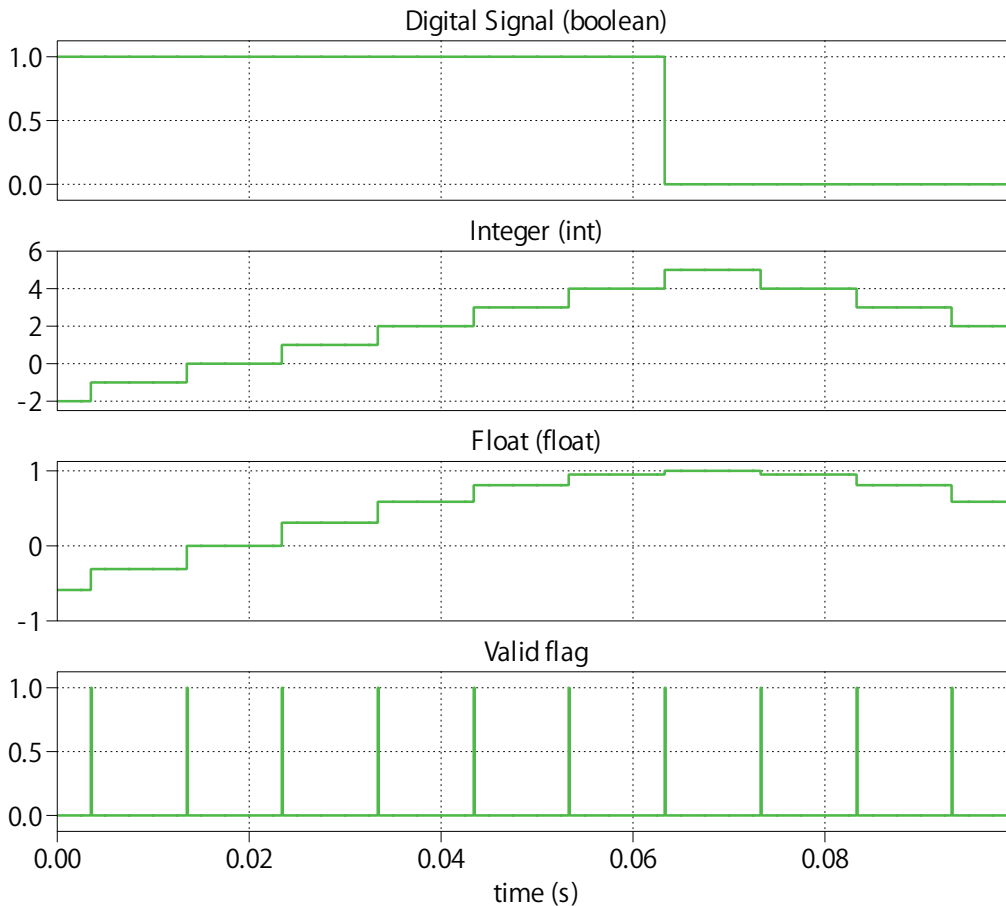
### 外部モード

生成されたコードがC2000ターゲット上で実行されると、ユーザは外部モードから、PLECSアプリケーションのPLECSスコープでリアルタイム波形を更新し、特定のシミュレーションパラメータを変更できます。以下の手順はターゲットデバイスへの接続方法の概要を示しており、追加のデバッグの詳細についてはTI C2000 Target Support User Manual[8]の"外部モードの開始"を参照してください。

- **Coder** -> **Coderオプション...**ウィンドウの左側にある**システム**リストから、目的のMCUを選択します。
- **外部モード**タブで、**ターゲットデバイス**フィールドの横にある  アイコンをクリックして、ターゲットデバイスを選択します。
- **接続**をクリックし、**自動トリガを有効化**をクリックして、サブシステムの表示とスコープで結果を確認します。

28379Dターゲットで受信したCAN波形を  に示します。

図5: 28379Dターゲットで受信したCAN波形



### パラメータのインライン化

パラメータを調整可能として構成するには、**Coder** -> **Coderオプション...**から**パラメータのインライン化**タブに移動します。回路図のコンポーネントを**例外**リストにドラッグアンドドロップすると、そのコンポーネントに関連付けられた調整可能なパラメータが実行時に調整可能になります。**例外**リストにはデフォルト設定とは逆の動作をするコンポーネントが指定されているため、この動作は**デフォルト設定**に依存することに注意してください。

この場合、外部モードに接続され、モデルが組み込みターゲットデバイス上で実行されるときに、CAN Transmitブロックへの"Test\_unit"入力をオンザフライで調整できます。パラメータの変更は、有効になるとすぐにPLECSスコープのトレースと数値表示に反映されます。

## 4 まとめ

このモデルは、シンプルなモデルを使用してTI C2000 TSPとのCAN通信を診断します。

## 5 参考文献

- [1] TI C2000 F280039C LaunchPad development kit,  
URL: <http://www.ti.com/tool/LAUNCHXL-F280039C>.
- [2] TI C2000 Piccolo MCU F280049C LaunchPad Development Kit  
URL: <http://www.ti.com/tool/LAUNCHXL-F280049C>.

- [3] TI C2000 Piccolo MCU F28069M LaunchPad Development Kit,  
URL: <http://www.ti.com/tool/LAUNCHXL-F28069M>.
- [4] TI C2000 Delfino MCUs F28377S LaunchPad Development Kit  
URL: <https://www.ti.com/lit/pdf/sprui25>
- [5] TI C2000 Delfino MCUs F28379D LaunchPad Development Kit,  
URL: <http://www.ti.com/tool/LAUNCHXL-F28379D>.
- [6] TI C2000 F28P550SJ9 LaunchPad development kit,  
URL: <https://www.ti.com/tool/LAUNCHXL-F28P55X>.
- [7] TI C2000 F28P650DK9 LaunchPad development kit,  
URL: <https://www.ti.com/tool/LAUNCHXL-F28P65X>.
- [8] PLECS TI C2000 Target Support User Manual,  
URL: <https://www.plexim.com/sites/default/files/c2000manual.pdf>.  
日本語版: <https://adv-auto.co.jp/products/plexim/manual.html>
- [9] PLECS User Manual,  
URL: <https://www.plexim.com/sites/default/files/plecsmanual.pdf>.  
日本語版: <https://adv-auto.co.jp/products/plexim/manual.html>

改訂履歴:

- C2000 TSP 1.3.1 初版
- C2000 TSP 1.4.5 Webリンクを更新
- C2000 TSP 1.6.1 280039C LaunchPadターゲットのサポートを追加
- C2000 TSP 1.9.1 28P550SJ9 LaunchPadターゲットのサポートを追加

**plexim** 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

**KESCO** 計測エンジニアリングシステムへの連絡方法:

☎ +81 3 6273 7505 Phone

+81 3 6285 0250 Fax

✉ Keisoku Engineering System CO.,LTD. Mail

1-9-5 Uchikanda, Chiyoda-ku  
Tokyo, 101-0047  
Japan

<https://kesco.co.jp> Web

*Embedded Code Generation Demo Model*

© 2002–2024 by Plexim GmbH

このマニュアルに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。