

(注1)本資料は英語版を翻訳した参考資料です。内容に相違がある場合には英語版を優先します。資料によっては英語版のバージョンが更新され、内容が変わっている場合があります。日本語版は、参考用としてご使用のうえ、最新および正式な内容については英語版のドキュメントを参照ください。

(注2)本資料の第6章まで(要旨除く)の日本語訳は、「[Synergy™ Software Package \(SSP\) v1.5.0 ユーザーズマニュアル モジュール概要編\(参考資料\)](#)」の第4章「モジュールの概要」に掲載されていますのでそちらを参照ください。

要旨 (Introduction)

本モジュールガイドは、ユーザがモジュールを効果的に使用してシステムが開発できるようになることを目的としています。このモジュールガイドを習得することで、開発システムへのモジュールの追加とターゲットアプリケーション向けの正確な設定 (configuration) ができ、さらに付属のアプリケーションプロジェクトコードを参照して、効率的なコード記述が行えるようになります。

より詳細な API や、より高度なモジュール使用法を記述した他のアプリケーションプロジェクト例もルネサス WEB サイト(本書末尾の「参考文献」の項を参照)から入手でき、より複雑な設計に役立ちます。

CAN (Controller Area Network、コントローラ領域ネットワーク) HAL モジュールは、CAN ネットワーク向けの汎用 API (generic API) です。CAN HAL モジュールは、r_can 内で実装されており、Renesas Synergy マイクロコントローラハードウェア (microcontroller hardware) で使用できる CAN 周辺回路 (peripherals) をサポートしています。ユーザコールバック関数を定義する必要があり、送信、受信、エラー割り込み受信の際に、ドライバはこの関数を実行します。コールバックは、チャンネル (channel)、メールボックス (mailbox)、イベント (event) を表すパラメータを返します。

目次

1. CAN HAL モジュールの特徴 (CAN HAL Module Features)	3
2. CAN HAL モジュールの API の概要 (CAN HAL Module APIs Overview)	3
3. CAN HAL モジュールの動作の概要 (CAN HAL Module Operational Overview)	3
4. アプリケーションへの CAN HAL モジュールの組み込み (Including the CAN HAL Module in an Application)	3
5. ADC HAL モジュールの構成 (Configuring the CAN HAL Module)	3
6. アプリケーションでの CAN HAL モジュールの使用 (Using the CAN HAL Module in an Application)	3
7. CAN HAL モジュールのアプリケーションプロジェクト (The CAN HAL Module Application Project)	3
8. ターゲットアプリケーションに対応する CAN HAL モジュールのカスタマイズ (Customizing the CAN HAL Module for a Target Application)	5

9. CAN HAL モジュールのアプリケーションプロジェクトの実行 (Running the CAN HAL Module Application Project)	5
10. CAN HAL モジュールのまとめ (CAN HAL Module Conclusion)	7
11. CAN HAL モジュールの次の手順 (CAN HAL Module Next Steps)	7
12. CAN HAL モジュールの参考情報 (CAN HAL Module Reference Information)	7

1. CAN HAL モジュールの特徴 (CAN HAL Module Features)
2. CAN HAL モジュールの API の概要 (CAN HAL Module APIs Overview)
3. CAN HAL モジュールの動作の概要 (CAN HAL Module Operational Overview)
4. アプリケーションへの CAN HAL モジュールの組み込み (Including the CAN HAL Module in an Application)
5. ADC HAL モジュールの構成 (Configuring the CAN HAL Module)
6. アプリケーションでの CAN HAL モジュールの使用 (Using the CAN HAL Module in an Application)
7. CAN HAL モジュールのアプリケーションプロジェクト (The CAN HAL Module Application Project)

このモジュールガイドに関連するアプリケーションプロジェクトは、設計全体の手順を示します。ISDE でアプリケーションプロジェクトをインポートして開き、CAN HAL モジュールに対応する設定項目を表示することができます。また、完成した設計 (complete design) で、CAN API を実現するために使用している `can_tx_rx.c` 内のコードを確認することもできます。

このアプリケーションプロジェクトは、CAN API の一般的な使用方法を示します。このアプリケーションプロジェクトは CAN HAL モジュールを初期化し、ユーザが SK-S7G2 ボード上のスイッチ S4 を押したときにメッセージを送信します。ユーザがスイッチを押していない場合、このプログラムは連続的な受信モード (continuous-reception mode) になっています。その結果は、共通のセミホスト機能 (semi-hosting function) を使用してデバッグコンソールに出力されます。以下の表は、このアプリケーションプロジェクトが使用する関連ソフトウェアとハードウェアの対象バージョンを示します。

表 1 このアプリケーションプロジェクトが使用するソフトウェアとハードウェアのリソース

リソース	リビジョン	説明
e ² studio	5.3.1 またはそれ以降	統合ソリューション開発環境 (ISDE)
SSP	1.2.0 またはそれ以降	Synergy ソフトウェアプラットフォーム
IAR EW for Synergy	7.71.2 またはそれ以降	IAR Embedded Workbench® for Renesas Synergy™
SSC	5.3.1 またはそれ以降	Synergy Standalone Configurator
SK-S7G2	v3.0, v3.1 またはそれ以降	スタータキット (2)

以下の図は、このアプリケーションプロジェクトの簡単なフローを示します。

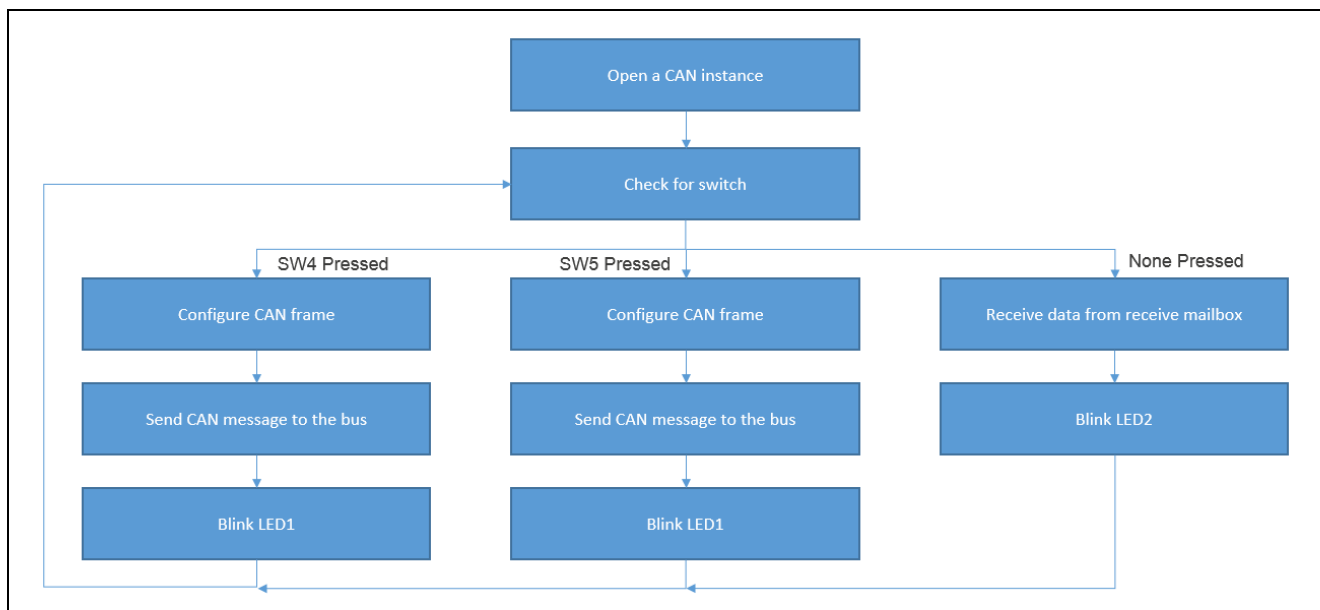


図 1 CAN HAL モジュールのアプリケーションプロジェクトのフロー図

hal_entry.c ファイルは、このプロジェクトを ISDE にインポートした後、プロジェクト内に置かれます。ISDE でこのファイルを開き、API の主な使い方を確認するのに有効な、以下の説明を参照することができます。HAL エントリから、can_tx_rx() 関数を呼び出します。can_tx_rx.c ファイルは、printf() を使用して結果を表示するセミホスト機能を許可するコードセクションを含んでいます。

can_tx_rx.h ファイルはヘッダファイルを持っており、CAN インスタンス構造体(instance-structure)、CAN 通信に対応するフラグ定義、CAN フレーム宣言(CAN frame declaration)を参照します。can_tx_rx.c ファイルは、メインプログラム制御セクション(main program-control section)に対応するエントリ関数(entry function)です。open API を使用して CAN HAL モジュールを初期化します。「無限」while ループ('forever' while loop)内で、スイッチ S4 とスイッチ S5 のステータスを監視しています。スイッチが押されると、関連する CAN transmit (送信) 関数を呼び出します。どちらのスイッチも押されていない場合、このモジュールは CAN 受信フラグ(CAN receive flag)が CAN 受信を表す状態にセットされるまで待ちます。各スイッチは、異なる CAN メッセージを設定されています。CAN の transmit (送信) 関数は write API を呼び出し、CAN バスにフレームを書き込みます。CAN の receive (受信) 関数は read API を呼び出し、CAN バスからフレームを読み取ります。printf() 関数を使用して、送受信したデータをコンソールに表示します。

最後のセクションはコールバック関数です。この関数は、実施した CAN イベントに従って、それぞれのフラグをセットします。

注記: この説明は、Synergy ソフトウェアパッケージ内のデバッグコンソールで printf() を使用方法をユーザが理解していることを想定しています。このような経験がない場合は、下記 WEB サイトの FAQ 2000008 「Synergy ソフトウェアパッケージのデバッグコンソールで Printf_使用方法」という記事を参照してください。デバッグモードで変数ウォッチ機能を使用して結果を表示することもできます。

<https://ja-support.renesas.com/knowledgeBase/17792531>

あるいは、デバッグモードで変数ウォッチ機能(watch variables)を使用して結果を表示することもできます。

ターゲットボードや MCU が必要とする動作や物理プロパティ(physical properties)をサポートするために、このアプリケーションプロジェクトではいくつかの重要なプロパティを設定しています。次の表に、それらのプロパティと、このプロジェクトにおける設定値を示します。このアプリケーションプロジェクトを開き、[Properties] ウィンドウでこれらの設定を表示することもできます。

表 2 アプリケーションプロジェクトに対応する CAN HAL モジュールの設定項目

ISDE のプロパティ	設定値
Parameter Checking	Disabled
Error Interrupt Priority	Priority 4
Receive Mailbox Interrupt Priority	Priority 4
Transmit Mailbox Interrupt Priority	Priority 4
Name	g_can0
Channel	0
Callback	can_callback
Mailbox 1 ID	0x7FF
Mailbox 0-3 Group Mask	0x7FF

メッセージのフィルタリングにより、CAN ノードが、コンフィギュレータ (configurator) で設定したメッセージ以外のメッセージを受信しません。

8. ターゲットアプリケーションに対応する CAN HAL モジュールのカスタマイズ (Customizing the CAN HAL Module for a Target Application)

いくつかの設定項目は通常、アプリケーションプロジェクトで示している値に対し、ユーザが変更を加えます。たとえば、コード内で ID とデータ長を更新する方法で、さまざまな形式のメッセージを設定することができます。1 個の CAN メッセージで取り扱える最大データサイズが 8 バイトであることに注意してください。ISDE の [Clock] タブでクロック設定を更新する方法で、CAN のビットレート (bitrate) を変更できます。

9. CAN HAL モジュールのアプリケーションプロジェクトの実行 (Running the CAN HAL Module Application Project)

CAN HAL モジュールのアプリケーションプロジェクトを実行し、さらにターゲットキットでその動作を確認するために ISDE にこのプロジェクトをインポートすることで、コンパイル、デバッグを実行することができます。このデモでは、2 枚のボードとし、同じプロジェクトをインポートした e² studio が必要です。CAN の端子は J7 ポートに配置してあります。1 枚のボードからもう 1 枚のボードに CANH を接続します。次に、CANL を GND に接続します。これら 2 つのインスタンスは、異なるワークスペース (workspaces) を持ちます。e² studio の両方のインスタンスにおけるデバッグ設定で、GDB と ADM に対して異なるポートを使用します。次の表を参照して、端子を接続します。

表 3 2 枚のボード間で CAN 通信を行うための端子接続

ボード 1	ボード 2
CANH	CANH
CANL	CANL
GND	GND

IAR EW for Synergy を使用する場合、以下の手順に従ってデバッガの設定を更新する必要があります。

1. プロジェクトのプロパティを開きます。
2. デバッガを選択し、ドライバを [J-Link/J-Trace] に変更します。
3. 次に [J-Link/J-Trace] を選択し、[Connection] タブをクリックします。
4. [USB] で、もう 1 台の USB デバイスを選択します。

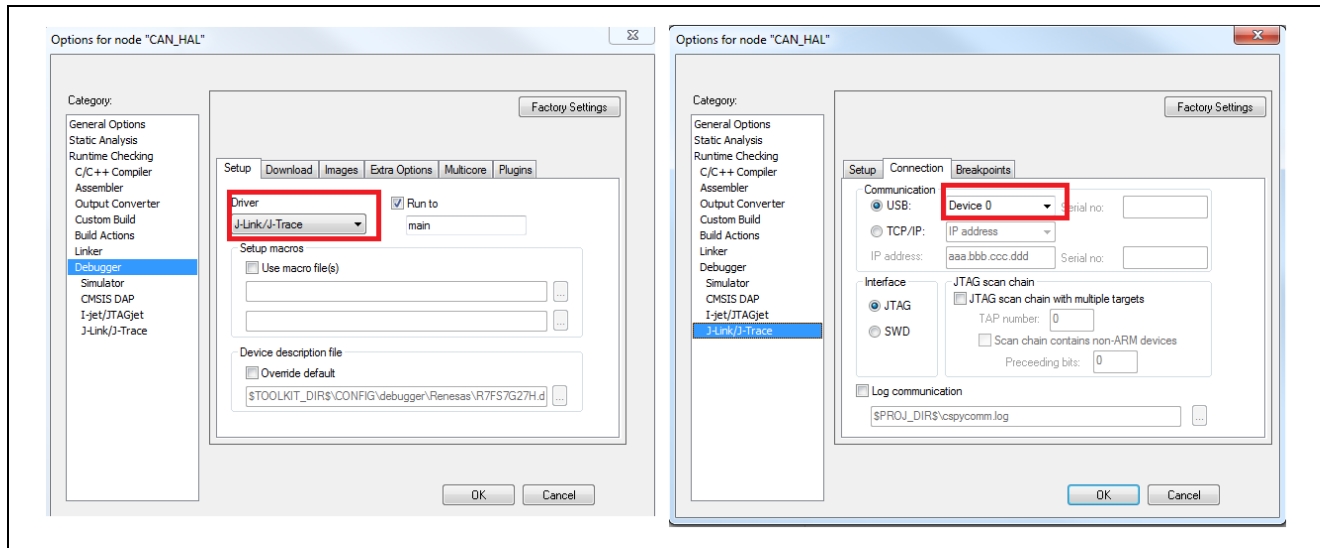


図 2 IAR EW for Synergy を使用する場合のデバッガの設定

注： 2 枚のボードを両方もとも 1 台の PC に接続する場合のみ、これらの手順を実行してください。2 台の異なるシステム (PC) でこのアプリケーションを実行する場合、デバッガの設定を変更する必要はありません。

新しいプロジェクト内で CAN HAL モジュールアプリケーションを実装するには、以下の手順のように、ターゲットキットで定義、設定、ファイルの自動生成、コードの追加、コンパイル、デバッグを行います。このガイドに示す手順により、SSP での開発プロセスをより実践的に習得するのに役立ちます。

e² studio または IAR Embedded Workbench® for Renesas Synergy™ にプロジェクトをインポートし、アプリケーションをビルドして実行する手順については、『Synergy プロジェクトインポートガイド』(下記WEB) を参照してください。

英語版:

<https://www.renesas.com/jp/ja/doc/products/renesas-synergy/apn/r11an0023eu0121-synergy-ssp-import-guide.pdf>

日本語版(参考資料):

<https://www.renesas.com/jp/ja/doc/products/renesas-synergy/apn/r11an0023ju0121-synergy-ssp-import-guide.pdf>

注： Synergy 開発プロセスの基本的な流れを経験したことのあるユーザにとって、以下の手順は十分詳細なものです。これらの手順をまだ理解していない場合、これらの手順の実施方法について、『SSP ユーザーズマニュアル』の最初のいくつかの章を参照してください。

- 2本のmicro USBケーブルを両方のSK-S7G2ボードのJ19にそれぞれつなぎ、ホストPCに接続します。
- アプリケーションのデバッグを開始します。

ISDE デバッグコンソール (ISDE Debug Console)(e2 studio または IAR EW for Synergy のターミナル I/O (terminal I/O))に、出力が表示されます。

```
Renesas Debug Virtual Console
RX flag: 1 ID: 2047 Received Data: 236
TX ID: 2047 TX Data: 236
TX ID: 2047 TX Data: 236
TX ID: 2046 TX Data: 146
TX ID: 2046 TX Data: 146
RX flag: 1 ID: 2047 Received Data: 236
```

図 3 CAN アプリケーションプロジェクトからのサンプル出力 (ボード 1)

```
Terminal I/O
Output:
TX ID: 2047 TX Data: 236
TX ID: 2046 TX Data: 146
RX flag: 1 ID: 2047 Received Data: 236
RX flag: 1 ID: 2047 Received Data: 236
TX ID: 2047 TX Data: 236
```

図 4 CAN アプリケーションプロジェクトからのサンプル出力 (ボード 2)

ボード 2 上のスイッチ S4 が押されている場合、ボード 2 は ID 2047 のメッセージを受信します。ボード 1 は ID 2046 のメッセージを受信しません。コンフィギュレータは、このメッセージをフィルタリングして除外するように設定してあります。ボードが自らの監視対象メッセージを受信したときは常に、RX フラグをセットします。

10. CAN HAL モジュールのまとめ (CAN HAL Module Conclusion)

このモジュールガイドは、サンプルプロジェクトでモジュールの選択、追加、設定、使用を行うために必要な背景となる情報全般を説明しました。従来の組み込みシステムでは、これらの手順を理解することに多くに時間を必要とし、また間違いが起こりやすい操作でした。Renesas Synergy プラットフォームにより、これら手順の所要時間が短くなり、設定項目の競合や、ローレベルドライバの誤った選択など、誤りが防止できるようになりました。アプリケーションプロジェクトで示したように、ハイレベル API を使用することで高いレベルの開発からスタートし、ローレベルドライバを作成するような従来の開発環境で必要とされる時間が不要になり、開発時間を短縮できます。

11. CAN HAL モジュールの次の手順 (CAN HAL Module Next Steps)

シンプルな CAN HAL モジュールプロジェクトをマスターした後、このモジュールと、CAN バス経由で他のモジュールからのデータを通信することができる他のモジュールを組み合わせることが考えられます。たとえば、開発中のアプリケーションが ADC に接続したセンサインタフェース (sensor interface) を使用しており、このアプリケーションが CAN バス経由で他のボードに接続されている場合、1 つのプロジェクトにおいて CAN モジュールと ADC モジュールを組み合わせることができます。

12. CAN HAL モジュールの参考情報 (CAN HAL Module Reference Information)

『SSP ユーザーズマニュアル』: SSP ディストリビューションパッケージの一部として HTML 形式が入手できるほか、Renesas Synergy™ WEBサイトのSSPページ

<https://www.renesas.com/jp/ja/products/synergy/software/ssp.html>から pdf を入手することもできます。

最新版の r_can モジュールの参考資料やリソースへのリンクは、以下の Synergy WEBサイトから入手できます。

<https://www.renesas.com/jp/ja/products/synergy.html>

ホームページとサポート窓口

サポート: <https://synergygallery.renesas.com/support>

テクニカルサポート:

- アメリカ: https://renesas.zendesk.com/anonymous_requests/new
- ヨーロッパ: <https://www.renesas.com/en-eu/support/contact.html>
- 日本: <https://www.renesas.com/ja-jp/support/contact.html>

すべての商標および登録商標はそれぞれの所有者に帰属します。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.01	2019.06.17		<ul style="list-style-type: none">・初版・英文版(R11AN0065EU0101、Rev.1.01、2017.Aug.22)の巻頭と第7章以降を翻訳

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。以下同じです。）に関し、当社は、一切その責任を負いません。
 2. 当社製品、本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
 3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
 4. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
 5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA機器、通信機器、計測機器、AV機器、
家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通制御（信号）、大規模通信機器、
金融端末基幹システム、各種安全制御装置等
当社製品は、データシート等により高信頼性、Harsh environment向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
 6. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
 7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
 8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
 9. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
 10. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものといたします。
 11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
 12. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。
- 注2. 本資料において使用されている「当社製品」とは、注1において定義された当社の開発、製造製品をいいます。

(Rev.4.0-1 2017.11)



ルネサスエレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合せ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社 〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

■技術的なお問合せおよび資料のご請求は下記へどうぞ。
総合お問合せ窓口：<https://www.renesas.com/contact/>