

【おすすめツール】

R20TS0577JJ0100

Rev.1.00

2020.05.16 号

GHS MULTI で簡単に実行履歴を取得して問題を早期解決

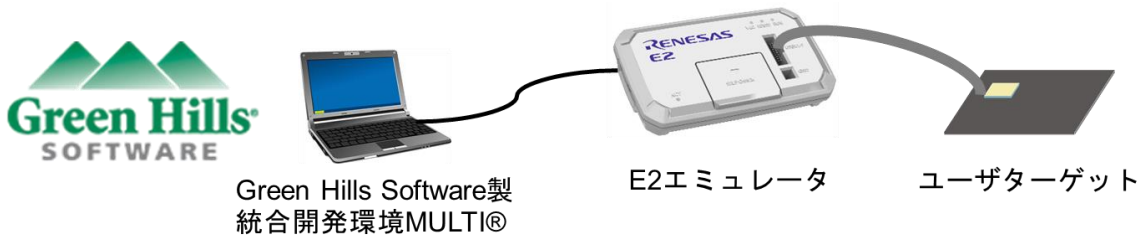
E2 エミュレータのソフトウェアトレース機能

概要

プログラムの問題の早期解決に役立つ E2 エミュレータの「ソフトウェアトレース機能」についてご紹介します。

ソフトウェアトレース機能は、プログラムのチェックしたい箇所にデバッグ命令を挿入しておくだけで、プログラムカウンタ(PC)値、レジスタ値などの実行履歴をトレースデータとして参照できる機能です。

本機能は、Green Hills Software 製統合開発環境 MULTI®(GHS MULTI)との組み合わせにより実現します。



GHS MULTI の詳細については、Green Hills Software 社ホームページをご参照ください。

<https://www.ghs.com/>

なお、本機能は、ルネサス製統合開発環境 CS+との組み合わせでも、ご使用いただけます。

1. 特長

GHS MULTI がサポートしている E2 エミュレータのソフトウェアトレース機能は、RH850 ファミリマイコンのデバッグ命令と連携し、プログラムカウンタ(PC)の値やレジスタ値などの実行履歴が確認できる機能です。

ソフトウェアトレース機能は以下の特長から簡単に実行履歴を取得できます。

- プログラムにデバッグ命令を挿入するだけで実行履歴の取得が可能
- トレースメモリを搭載していない RH850 ファミリマイコンでも実行履歴の取得が可能

デバッグ命令の一覧は以下の通りです。このデバッグ命令は CPU 内部の動作に影響するようなコンパイラの標準出力関数などを使用せずに実行履歴を確認いただけます。

デバッグ命令の詳細につきましては、RH850 各コアのデバッグ命令編をご参照ください。

表 1 デバッグ命令一覧

デバッグ命令	命令形式	ソフトウェアトレース出力内容
DBCP	DBCP	プログラムカウンタ(PC)値
DBTAG	DBTAG imm10	プログラムカウンタ(PC)値、 10ビット・イミーディエト(imm10)値
DBPUSH	DBPUSH rh-rt	プログラムカウンタ(PC)値、 汎用レジスタ rh から rt までのレジスタ番号とレジスタ値

お客様がプログラムのチェックしたい箇所にデバッグ命令を挿入するだけで、プログラムカウンタ(PC)値、レジスタ(reg)、値(val)などの実行履歴をトレースリストで参照可能です。

The top screenshot shows the GHS MULTI Editor with the following code snippet:

```

void test_debug_inst(int val1, int val2)
{
    __asm("dbtag 0x12");
    __asm("dbpush r6-r7");
}

int main()
{
    int f;
    test_posfttrace(0x11223344, 0x55667788);
    __asm("dbcp");
    test_debug_inst(0x12345678, 0x9abcdef0);
    return 0;
}
    
```

Blue dashed boxes highlight the inserted debug commands: `__asm("dbtag 0x12");`, `__asm("dbpush r6-r7");`, `__asm("dbcp");`, and `test_debug_inst(0x12345678, 0x9abcdef0);`. A callout box points to these with the text: "プログラムにデバッグ命令を挿入".

The bottom screenshot shows the Trace List window with the following data:

Address	Total Cycles	Total Time
DBCP: pc=0x000011c6	5	25.0 ns
Trace immediate data: 18		
DBPUSH: pc=0x000011a2, reg=6, val=0x12345678	17,280	86,400.0 ns
DBPUSH: pc=0x000011a2, reg=7, val=0x9abcdef0	22,464	112,320.0 ns

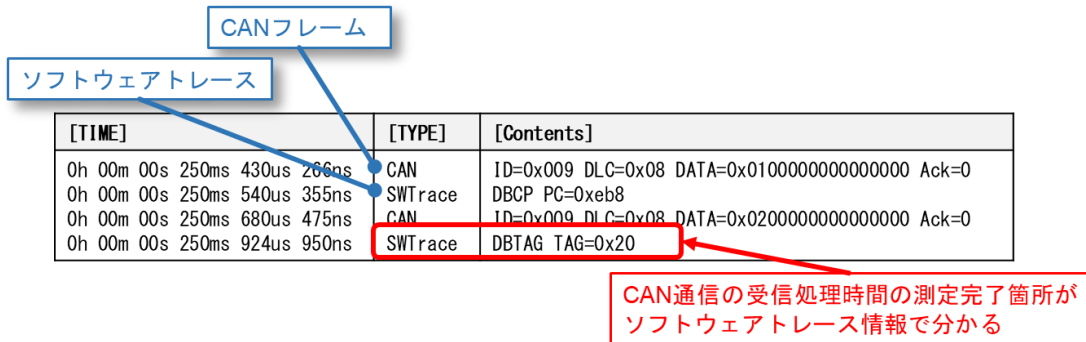
A callout box points to this table with the text: "GHS MULTIのトレースリストで実行履歴が参照可能".

ビルド時のオプションで「-gen_entry_exit_arg_history」と「-record_entry_exit_history」を指定した場合、プログラムの関数の入口と出口に自動でデバッグ命令が挿入されます。このソフトウェアトレースの結果は、パスアナライザで関数情報として参照することができます。

The screenshot shows the PathAnalyzer - swtrace window displaying a call graph. The graph shows a sequence of function calls: `main`, `function_call1`, `function_call2`, and `function_call3`. The execution time for each function call is shown in cycles. For example, `function_call1` takes 0.2 Gcycles, `function_call2` takes 0.4 Gcycles, and `function_call3` takes 0.6 Gcycles. A callout box points to the graph with the text: "GHS MULTIのパスアナライザで関数情報が参照可能".

2. CAN 通信時間計測ソリューションとの連携

E2 エミュレータの特長のひとつである「CAN 通信時間計測ソリューション」と組み合わせることで、CAN フレームとソフトウェアトレース機能のトレースデータを同時に取得できますので、プログラムの問題箇所の特定やパフォーマンスの検証に役立てることができます。



CAN 通信時間計測ソリューションについて、以下のツールニュースで紹介していますので併せてご参照ください。

<https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ts0504>

【お知らせ】

GHS MULTI で CAN 通信の受信処理時間の問題を早期解決
E2 エミュレータ CAN 通信時間計測ソリューションのご紹介

3. E2 エミュレータ

3.1 製品概要

E2 エミュレータは、「開発効率の向上」をコンセプトとした高機能オンチップデバッグエミュレータ兼フラッシュプログラマです。

E2 エミュレータの詳細については、以下の URL をご参照ください。

<https://www.renesas.com/e2>

3.2 購入方法

E2 エミュレータをご注文の際には、以下の情報を最寄りの弊社営業または特約店までご連絡ください。価格についても同様にお問い合わせください。

製品名	E2 エミュレータ
発注型名	RTE0T00020KCE00000R

4. ご参考

ソフトウェアトレース機能と活用例について、以下のツールニュースでもご紹介しています。

<https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ts0345>

【お知らせ】

プログラムの実行履歴を確認でき、お客様の各種問題を早期に解決！
E2 エミュレータ ソフトウェアトレース機能のご紹介

以上

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	May.16.20	-	新規発行

本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無いことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。

過去のニュース内容は発行当時の情報をもとにしており、現時点では変更された情報や無効な情報が含まれている場合があります。

ニュース本文中の URL を予告なしに変更または中止することがありますので、あらかじめご承知ください。

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24 (豊洲フォレシア)

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。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。