
RTA-TRACE

入門ガイド

著作権について

本書のデータを LiveDevices Ltd. からの通知なしに変更しないでください。LiveDevices Ltd. は、本書に関してこれ以外は一切の責任を負いかねます。本書に記載されているソフトウェアは、お客様が一般ライセンス契約あるいは単一ライセンスをお持ちの場合に限り使用できます。ご利用および複写はその契約で明記されている場合に限り、認められます。

本書のいかなる部分も、LiveDevices Ltd. からの書面による許可を得ずに、複写、転載、伝送、検索システムに格納、あるいは他言語に翻訳することは禁じられています。

© **Copyright 2003/2004** LiveDevices Ltd.

本書で使用する製品名および名称は、各社の（登録）商標あるいはブランドです。

Document TD00001-004

目次

1	本書について	5
1.1	本書の対象ユーザー	5
1.2	表記上の規約	5
2	RTA-TRACE とは	6
2.1	RTA-TRACE の概要	6
2.2	トレースを有効にする方法	7
2.3	ERCOS ^{EK} のコンフィギュレーション	7
2.4	RTA-OSEK のコンフィギュレーション	7
3	インストールの手順	9
3.1	ホスト PC	9
3.2	ターゲットソフトウェア	9
3.2.1	ERCOS ^{EK}	9
3.2.2	RTA-OSEK	9
4	RTA-OSEK のサンプルアプリケーションのトレース	10
4.1	操作手順	10
5	トレースを有効にして ERCOS ^{EK} アプリケーションをビルドする	12
5.1	TRACE_EG (サンプルアプリケーション)	12
5.1.1	tskLeader	12
5.1.2	tskFollower	12
5.1.3	tskInterfere	12
5.1.4	tskIdle	13
5.2	ERCOS ^{EK} 用の TRACE_EG をビルドする	13

5.2.1	ファイルの場所	13
5.2.2	ビルド	14
6	RTA-TRACE クライアントの表示内容	15
6.1	RTA-TRACE クライアントを起動する	15
6.2	各種コントロール	15
6.2.1	トランスポートコントロール	15
6.2.2	バッファステータスインジケータ	16
6.2.3	タイムベースコントロール	16
6.2.4	トリガコントロール	16
6.3	サンプルトレースデータ	16
7	お問い合わせ先	17

1 本書について

本書は RTA-TRACE の入門書です。RTA-TRACE のインストール方法や、トレースを有効にして簡単なアプリケーションをビルドする方法について説明します。また、RTA-TRACE ユーザーインターフェースの基本的な操作方法についても説明します。

RTA-TRACE のホストソフトウェアコンポーネントは、Windows XP、Windows 98SE、Windows NT、Windows ME の下で稼働します。

1.1 本書の対象ユーザー

本書『RTA-TRACE 入門ガイド』は、RTOS (ERCOS^{EK} や RTA-OSEK などのリアルタイム OS) を使用した組み込みシステムを C 言語でプログラミングした経験のある技術者を対象としています。また本書を読む際には、RTOS がすでにインストールされていて、その RTOS に付属しているチュートリアルアプリケーションのビルドや実行が可能な状態になっていることが必要です。

1.2 表記上の規約

重要：このように表記されている注記には、ユーザーが知っておく必要のある重要な情報が記載されています。内容をよく読み、記載されているすべての指示に必ず従ってください。

移植性：このように表記されている注記では、RTA-OSEK コンポーネントが実行されるプロセッサ上で実行できるコードを作成する場合に知っておく必要がある事柄について説明されています。

本書では、プログラムコード、ヘッダファイル名、C のデータ型名、C 関数および API 関数名はすべてクーリエ体 (*courier*) で表記されています。オブジェクトの名前も、プログラマに公開され次第やはりクーリエ体で表記されます。たとえば、Task1 という名前のタスクは、Task1 という名前のタスクハンドルとして表記されます。

コマンドプロンプトのセッションについては、出力される部分がクーリエ体 (*courier*) で表記され、ユーザーが入力する部分がクーリエボールド体 (*courier bold*) で表記されます。

GUI エlementとのインタラクションについての記述では、Elementのキャプションは**ボールド体 (bold)** で表記されています。また、メニューなどの階層的なナビゲーションは矢印でレベルを区切り、たとえば、「メニューコマンド **Edit → Select All** を選択します。」、または「メニューから **Edit → Select All** を選択します。」のように表記されています。

また PDF 文書において、索引、および他の部分を参照する箇所（例：「第 3 章を参照してください」の部分）については、その参照先へのリンクが設けられているので、必要な参照箇所を素早く見つけることができます。

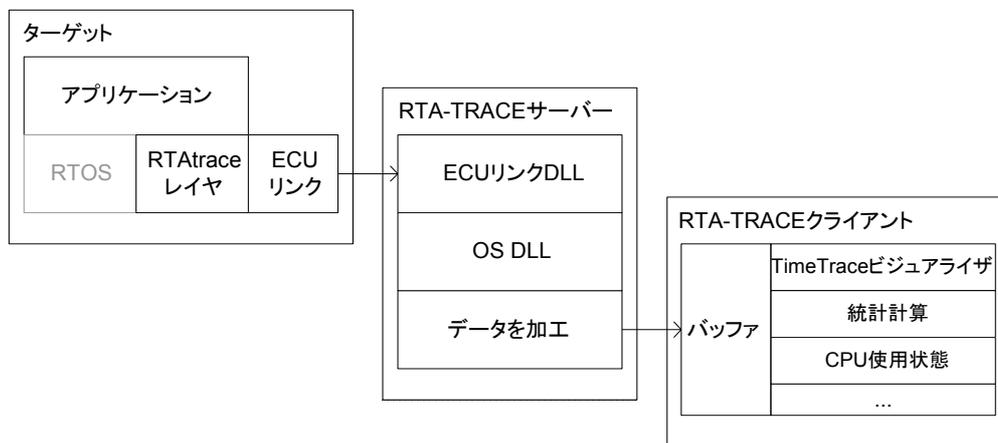
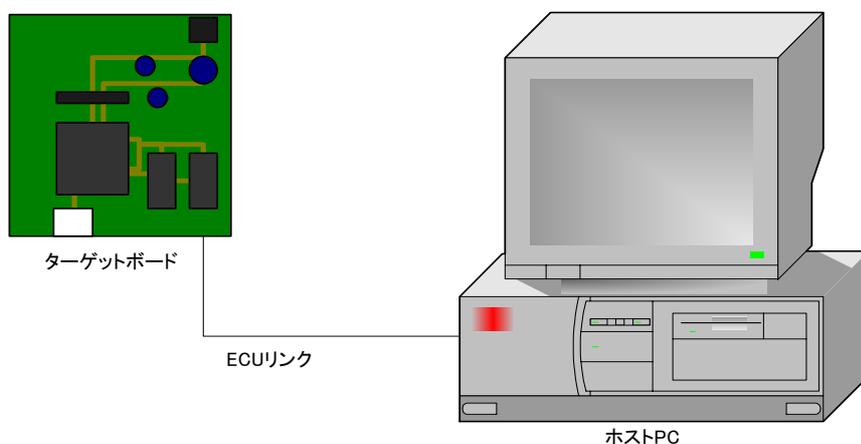
2 RTA-TRACE とは

RTA-TRACE は、実行中のアプリケーションの挙動を記録し、即時情報と要約情報を画面に表示します。これにより、実際のシステム挙動を正確に把握でき、リアルタイムに表示される情報や作成されたレポート、またさらに測定ツールを用いて、異常時の状態を詳しく分析することができます。

RTA-TRACE では、OS の動作（タスク起動、リソースのロック、アラームなど）とユーザー定義の事象を記録できます。この際、「トレースポイント」、「タスクトレースポイント」、および「インターバル」を定義して、アプリケーション内の特定の部分をマークすることができます。これらの用語の定義は、『RTA-TRACE ユーザーズガイド』およびオンラインヘルプに詳しく記載されています。

2.1 RTA-TRACE の概要

RTA-TRACE を使用してアプリケーションをトレースするためには、ターゲットハードウェア（トレース対象となるアプリケーションを実行するハードウェア）とホスト PC（RTA-TRACE ツールを実行する PC）が必要です。



RTA-TRACE ツールは、RTA-TRACE サーバー（'RTA-TRACE Server'）と RTA-TRACE クライアント（'RTA-TRACE Client'）という 2 つのモジュールで構成されています。サーバーはターゲットソフトウェアからデータを受け取り、それを RTA-TRACE クライアントで使用できるように加工します。クライアントはサーバーから受け取ったデータを読み取ってトレースバッファに格納し、ユーザーが各種プラグインツール（TimeTrace ビジュアライザなど）を利用してトレースデータの視覚化や分析をできるようにします。

ターゲットソフトウェアは、コアの RTA-TRACE ライブラリ（ヘッダファイルを含む）と、必要に応じて追加された ECU リンクライブラリ（RS232 など）で構成されます。また、RTA-TRACE には、デバッガを使用してトレースデータをターゲットから抽出する、という非常に便利な機能も用意されています。ECU リンクについては、『RTA-TRACE ECU リンクガイド』に詳しく説明されています。

2.2 トレースを有効にする方法

トレースを有効にするための最も簡単な方法として、コンフィギュレーションファイルとビルド実行ファイル（Make ファイルまたはバッチファイル）に少しの変更を加えるだけで、基本的なイベントを RTA-TRACE でトレースすることができます。他の C コード部分を変更する必要はありません。

この方法で RTA-OSEK や ERCOS^{EK} のトレースを有効にすると、ビルドプロセスにおいて、トレースデータを記録する機能を含んだ「計装（'instrumented'）バージョン」のオペレーティングシステムを使用するので、ユーザーはデバッグスクリプトを用いてトレースされたデータをダンプしたり、RTA-TRACE クライアントにロードして分析することができます。

しかし、一般的にはこれ以外の機能（ランタイムにトレースデータを RS232 経由でアップロードする機能や、トレース情報をユーザー定義できる機能など）も必要になります。このような場合、ユーザーはアプリケーション内でトレース用 API 関数をコールする必要があり、さらに、ECU リンクによってコールされる関数を作成する必要がある場合もあります（『RTA-TRACE ユーザーズガイド』および『RTA-TRACE ECU リンクガイド』を参照してください）。

2.3 ERCOS^{EK} のコンフィギュレーション

ERCOS^{EK} のトレース設定は run.bat や project_settings.mk が格納されているビルドディレクトリ内にある RTAtrace.cfg というファイルによって行います。このファイルにより、タスク起動やリソースロックといったさまざまなクラスのトレースイベントのオン/オフを切り替えることができます。

さらにこのファイルには、トレースポイント、タスクトレースポイント、およびインターバルを定義でき、また、ランタイムにおけるターゲットのトレース用コンフィギュレーションも設定できます。詳細については『RTA-TRACE コンフィギュレーションガイド』を参照してください。

注記

トレースを有効にする処理は、一般に、コンフィギュレーションファイル内で指示語を用いて行うのではなく、アプリケーションコード内の "OS" 節で属性を設定することにより行います。ERCOS^{EK} の場合は以下のようなコードを使用します。

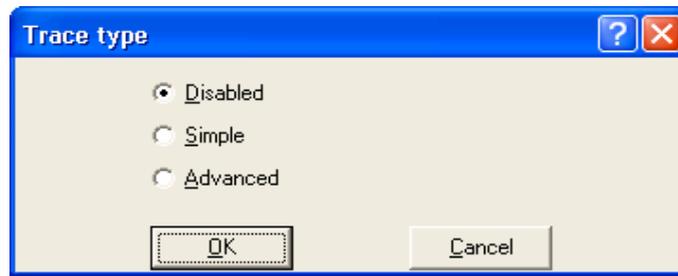
```
OS {  
    RT_ENABLE = TRUE;  
};
```

2.4 RTA-OSEK のコンフィギュレーション

RTA-OSEK のトレース設定はコンフィギュレーションツールを使用して行います。'Trace' ボタンをクリックするとトレース設定ページが開き、ここでタスク起動やリソースロックなどのトレースイベントクラスのオン/オフを切り替えることができます。

さらに、トレースポイント、タスクトレースポイント、およびインターバルをここで定義でき、また、ランタイムにおけるターゲットのトレース用コンフィギュレーションも設定できます。詳細については『RTA-TRACE コンフィギュレーションガイド』を参照してください

トレースを有効にするには、GUI の Trace/Configuration セクションで、Trace Type として Simple または Advanced を選択します。



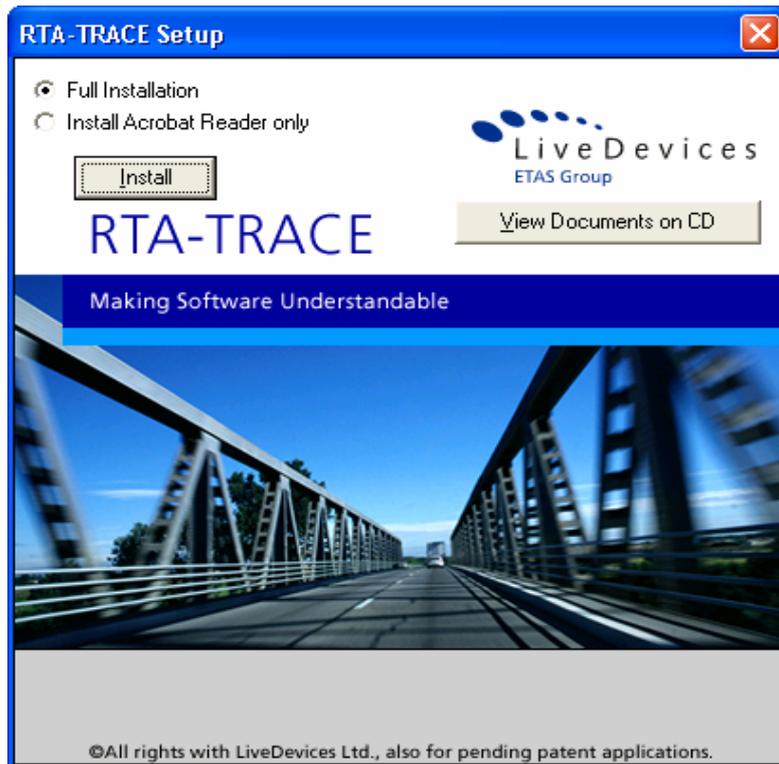
Advanced を選択すると、Simple を選択した場合よりも詳しいトレースが行われ、それに伴いトレースデータの量も多くなります。

3 インストールの手順

3.1 ホスト PC

RTA-TRACE のホストソフトウェアは、CD からインストールします。

1. RTA-TRACE のツール CD を CD ドライブに挿入します。
2. インストーラプログラムが自動起動しない場合は、CD 内の 'setup.exe' というファイルを実行してください。すると、下図のsplashウィンドウが開きます。



3. **Install** ボタンをクリックすると RTA-TRACE の Windows インストールウィザードが開き、製品と共に供給される RTA-TRACE のライセンスファイルの格納場所を入力するよう要求されます。インストールが完了したら **Finish** ボタンをクリックします。

3.2 ターゲットソフトウェア

3.2.1 ERCOS^{EK}

バージョン 4.3 より前の ERCOS^{EK} を使用する場合には、ターゲットソフトウェアのインストールとビルドプロセスの変更を行う必要がありますので、ETAS のサポート窓口までお問い合わせください。バージョン 4.3 以降には、トレースをサポートする機能が組み込まれています。

ツールとターゲットソフトウェアのインストールが終わると、'TRACE_EG' というサンプルアプリケーションをビルドできるようになります。これについては第 4 章で説明します。

3.2.2 RTA-OSEK

バージョン 4 以降の RTA-OSEK には RTA-TRACE に対応する機能が組み込まれているので、これ以上何もインストールする必要はありません。

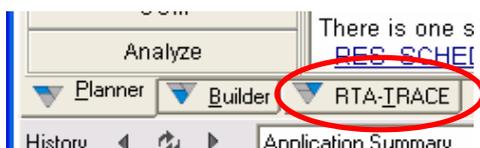
4 RTA-OSEK のサンプルアプリケーションのトレース

4.1 操作手順

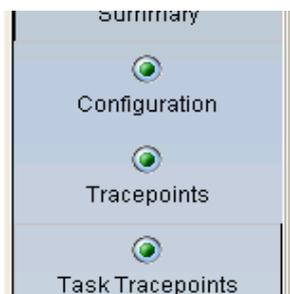
RTA-OSEK には、ビルド環境の検証を行うためのサンプルアプリケーションが付属しています。このアプリケーションのトレースを有効にすることにより、RTA-OSEK 用の RTA-TRACE が正しく機能しているかどうかを確認することができます。

以下の手順でトレースを有効にします。

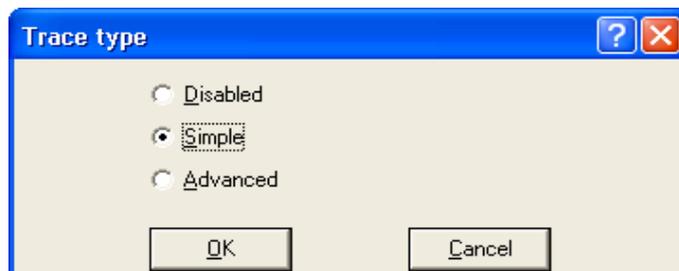
1. RTrs232.c というファイルを <target>\Drivers ディレクトリからサンプルディレクトリにコピーします。
2. ウィンドウ内のパネルで、RTA-TRACE タブを選択します。



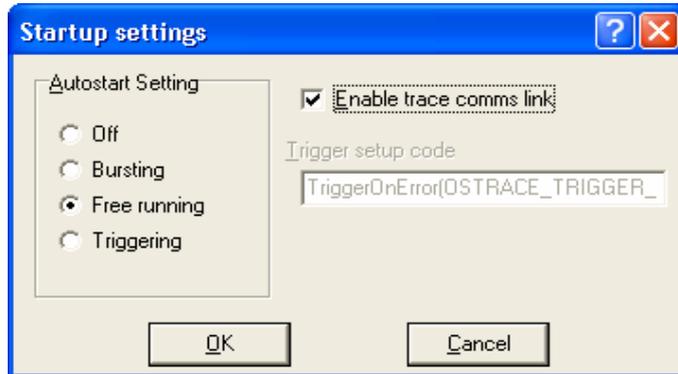
3. Configuration ペーンを選択します。



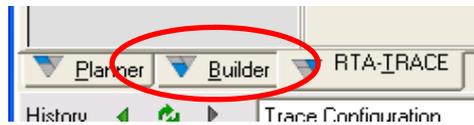
4. それから Trace Type ボタンをクリックし、Simple トレースを有効にします。



5. **Autostart** ボタンをクリックして “Free running” を選択します。また、“Enable trace comms link” も有効にします。



6. Builder タブを選択します。



7. **Custom Build** ボタンをクリックしてから **Configure** をクリックします。すると “Custom Build Options” ダイアログボックスが開きます。
8. “Build Script” 入力フィールドの “call rtkbuild.bat” の下に、以下のコードを追加します。

```
%CC% %COPTS% RTrs232.c
```

9. リンカコントロール行を編集して、RTserby.t.\$(LIBEXT) というライブラリファイルと RTrs232.\$(OBJEXT) というオブジェクトファイルを追加します。
10. target.h の以下の部分を編集します。

変更前 :

```
#define TARGET_IDLE() ((void)0) /* nothing */
```

変更後 :

```
#define TARGET_IDLE() {\
    CheckTraceOutput();\
    UploadTraceData();\
}
```

11. target.c を編集して、ストップウォッチ関数を追加します。関数のプロトタイプは以下のとおりです。

```
OS_NONREENTRANT (StopwatchTickType)
osTraceStopwatch(void)
```

12. 以上の変更が終わったら、**Create ‘rtkbuild.bat’** をクリックして、ビルドファイルを作成し直します。
13. 続いて **Build Now** をクリックして、サンプルアプリケーションをビルドします。トレースツールの使用法については、第 6 章を参照してください。

5 トレースを有効にして ERCOS^{EK} アプリケーションをビルドする

ERCOS^{EK} 用 RTA-TRACE には、TRACE_EG というサンプルアプリケーションが付属しています。このアプリケーションは、RTA-TRACE ツールの使用方法を理解するための簡単なもので、さらにこのアプリケーションで、RTA-TRACE が正しくセットアップされていることを簡単に確認できるようになっています。

本章では、サンプルアプリケーションの内容について説明し、さらにそれをビルドしてその挙動を RS232 リンク経由で調べる方法について説明します。ここにはターゲットハードウェアに依存する関数も含まれていますので、場合によっては、これらの関数を実際に使用するターゲットに合わせて調整する必要があります。ここで紹介する例は、Motorola PowerPC 56x 用 Diab ツールチェーン向けに作成されたものですが、他のターゲットにも適用できます。

5.1 TRACE_EG (サンプルアプリケーション)

このサンプルアプリケーションは、4 つのタスクと 2 つのアラームで構成されるシンプルなシステムです。以下に、これらのタスクについて優先度の高いものから順に説明します。

5.1.1 tskLeader

このタスクは 250,000 クロックチェックごとに almLeader により起動されます。

そして、短い時間だけ「仕事」(実際にはビジーウエイトループ)を行い、最後に tskFollower を起動します。

このタスクの開始時と終了時には、leaderBounds というタスクトレースポイントがロギングされます。タスクトレースポイントは、画面上の tskLeader の帯(行)に表示されます。

5.1.2 tskFollower

このタスクは、リソース rscContention を取得した後に「仕事」を行い、それからリソースを解放する、という非常にシンプルなタスクです。

リソース rscContention は、次に説明するタスク tskInterfere との間で共有されます。

5.1.3 tskInterfere

このタスクは、240,000 クロックチェックごとに almInterfere により起動されます。つまり、tskInterfere は tskLeader (およびそれに続く tskFollower) とは少しずれて起動されます。

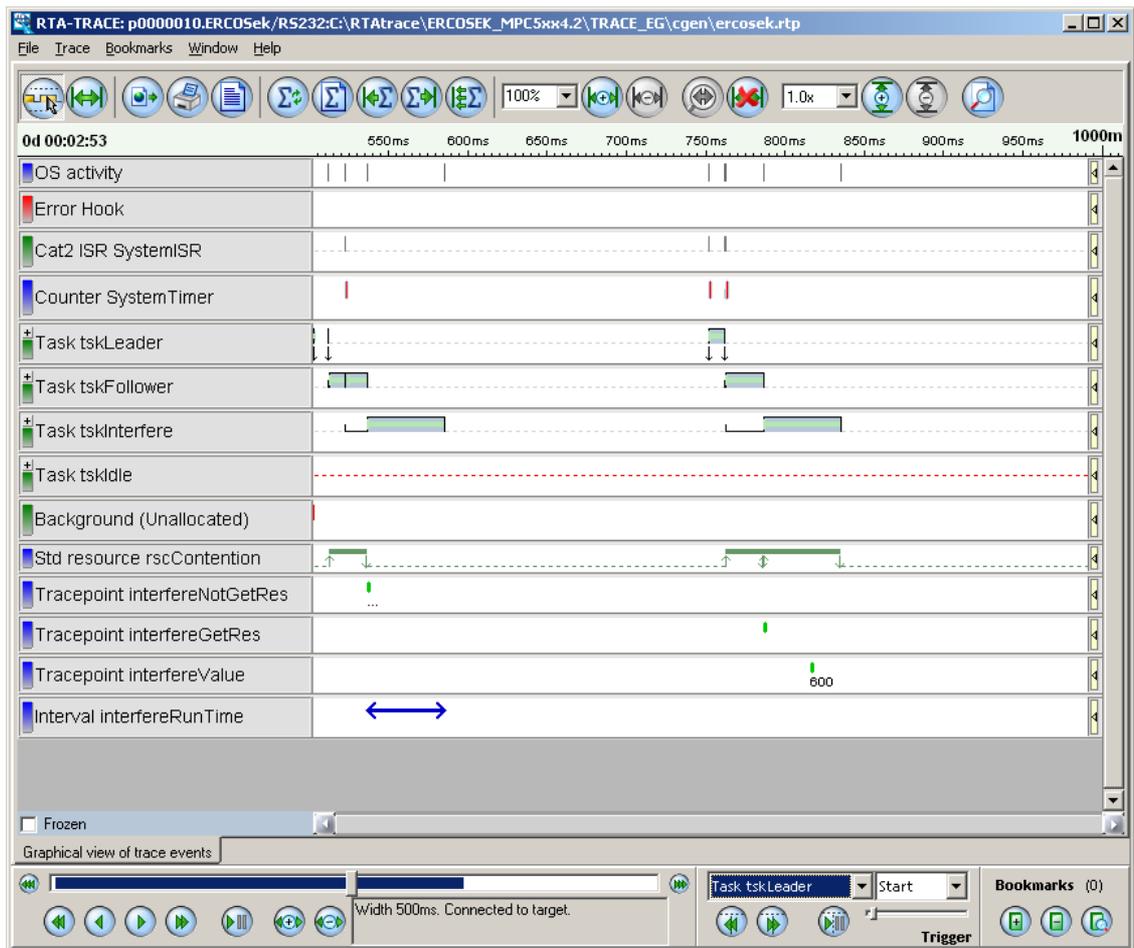
tskInterfere はシステム中で最も複雑な処理を行うタスクで、実行されるたびに実行モードが A と B とに切り替わります。各モードでの機能は以下のとおりです。

- モード A では、トレースポイント interfereGetRes をロギングしてリソース rscContention を取得します。このリソースはこのタスクの処理が終わるまで確保されているため、このリソースが解放されるまで tskFollower の実行が保留されることがあります。
- モード B では、表示される時間トレースに、トレースポイント interfereNotGetRes を小さなデータブロックとともに挿入します。

その後は、どちらのモードにおいても 2 つのループに順に入って何かの「仕事」を行います。ループ実行時間は常に一定です。

- モード A では、2 つのループの間でトレースポイント interfereValue がロギングされ、それまでに実行された「仕事」の量に相当する値が表示されます。
- モード B では、2 つのループの間でインターバルが開始されます。このインターバルは、このタスクの終わりに終了します。

このサンプルアプリケーションの実行の様子は、TimeTrace ビジュアライザを実行する RTA-TRACE のスクリーンショット(下図)で見ることができます(これは ERCOS^{EK} バージョンの例です)。



このスクリーンショットには基本的なトレース情報が表示されています。トリガは、トレース表示が `tskLeader` の開始を起点にして固定されるようになっています。最初は、各タスクの「帯」(行) が上から優先度順に表示され、その下にトレースポイントとインターバルを示す帯が表示されます。タスクトレースポイントは、その発生原因になったタスク (`tskLeader` など) の帯に表示されます。

5.1.4 tskIdle

このタスクは `modeMain` というアプリケーションモードで実行を開始し、システム内の 2 つの周期アラーム (`almLeader` および `almInterfere`) を起動します。それからループに入って `UploadTraceData()` をポーリングし、トレースデータをホストに送ります。

また、RTA-TRACE ターゲットコードの起動も行います。このタスクはトレース開始時点ですでに実行中になっているので、TimeTrace ビジュアライザで見ることができません。

5.2 ERCOS^{EK} 用の TRACE_EG をビルドする

5.2.1 ファイルの場所

TRACE_EG は ERCOS^{EK} のインストールディレクトリ (たとえば `C:\¥ETAS¥ERSOEK_MPC5xx4.x`) の下にディレクトリツリーの形でインストールされます。重要なファイルは以下のとおりです。

...¥TRACE_EG¥RTAtrace.cfg	RTA-TRACE コンフィギュレーションファイル
...¥TRACE_EG¥SOURCE¥tsk*.c	各タスクのコード
...¥TRACE_EG¥SOURCE¥target.c	ターゲット固有のコード
...¥TRACE_EG¥SOURCE¥tgt_ser1.c	ターゲット固有のシリアル通信ドライバ

Windows のコマンドプロンプトウィンドウを開き、以下の例のように、ERCOS^{EK} のインストールディレクトリ下に入ってください。

```
C:¥>cd ETAS
C:¥ETAS>cd ERCOSEK_MPC5xx4.x
```

5.2.2 ビルド

1. TRACE_EG サブディレクトリにはビルドスクリプトがあります。

```
C:\ETAS\ERCOSEK_MPC5xx4.x>cd TRACE_EG
C:\ETAS\ERCOSEK_MPC5xx4.x\TRACE_EG>dir /b
RTAtrace.cfg
project_settings.mk
SOURCE
T32
run.bat
cgen
```

2. SOURCE¥tgt_ser1.c というファイルを編集し、ターゲットボード上のシリアルハードウェアを駆動するようにします。このためには、関数 osTraceInitUART、osTraceCommTxByte、osTraceCommDisableTxIST、osTraceCommEnableTxIST、および osTraceCommTxReady を修正して、RTArs232.a がボードのシリアルハードウェアを正しく駆動できるようにする必要があります。
3. 供給されている Make ファイルとバッチファイルを使用してアプリケーションをビルドします。

```
...\TRACE_EG>run.bat project_settings.mk build
build step "ESCAPE analysing, preprocessing, and parsing"
analysing include dependencies trace_eg.esc
preprocessing and parsing trace_eg.esc
build step "ECCO code generation"
```

```
ECCO V4.2 -- (C) ETAS GmbH, Stuttgart
Ok, doing first analysis pass ...
analyzing c_bastyp.db ... done in 0.00 sec
...
<ECCO output removed>
...
...\TRACE_EG>
```

4. このビルドプロセスでは2つの主要なファイルが生成されます。1つは実行ファイル (trace_eg.elf)、もう1つは RTA-TRACE アプリケーションディスクリプションファイル (ercosek.rtp) で、RTA-TRACE サーバーにより使用されます。
5. これで、生成されたアプリケーションをターゲットボードにダウンロードして実行を開始できるようになりました。

6 RTA-TRACE クライアントの表示内容

6.1 RTA-TRACE クライアントを起動する

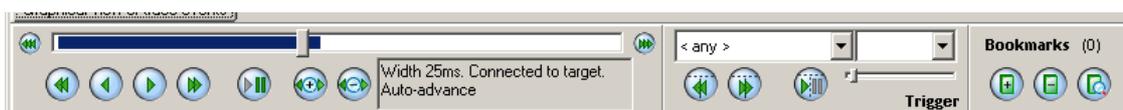
ターゲットソフトウェアを実行すると、サンプルアプリケーションがトレースデータの送信を始めます。

1. **Start** → **Programs** → **RTA** → **RTA-TRACE** を選択して、RTA-TRACE クライアントを起動します。
クライアントプログラムが起動すると、**File** メニューと **Help** メニュー、およびブランクのウィンドウが表示されます。
2. **File** → **New Connection...** を選択します。
RTA-TRACE サーバーの場所の入力を要求するダイアログボックスが開きます。この場所はデフォルトでは 'localhost' (つまり RTA-TRACE クライアントを実行しているコンピュータ) です。ここには IP アドレスまたは解決可能なマシン名をアドレスとして入力しますが、通常は 'localhost' と入力します。
3. 次に、このマシン上で使用可能な OS と ECU-Link の組み合わせを表示するダイアログが表示されます。
4. 使用環境に合わせた組み合わせ (つまりここでは RTA0SEK-RS232) を選択します。
ブラウズダイアログが表示されます。
5. ビルドプロセスで生成されたアプリケーションディスクリプション (拡張子が .rtp または .rta のファイル) の位置を指定し (通常、プログラムの実行ファイルと同じディレクトリにあります)、**OK** をクリックします。
RTA-TRACE クライアントが RTA-TRACE サーバーに接続し、デフォルトでは TimeTrace ビジュアライザが起動します。
ビジュアライザには一連の水平な帯 (紙テープに似ています) が表示され、トレース対象のタスクやその他のオブジェクトの挙動が表示されます。
6. トレースデータがすぐに表示されない場合には、**File** → **Configure Connection...** を選択して接続設定ダイアログを開き、通信パラメータをターゲットハードウェアに合わせて設定してください。

クライアント画面の一番下には、トランスポートコントロール、バッファステータスインジケータ、タイムベースコントロール、およびトリガコントロールがあります。次に、これらについて説明します。

6.2 各種コントロール

コントロール領域は下図のようになっています。各ボタンについて以下に説明します。



6.2.1 トランスポートコントロール

これらはテープレコーダやビデオデッキの操作ボタンに似ています。左から右の順に説明します。



トレースバッファの 1 ページ前に戻ります。ページサイズはタイムベースコントロール (後述) を用いて設定します。



1 ページの 10% だけ前に戻ります。



トレースバッファの 1 ページの 10% だけ進みます。



トレースバッファの 1 ページ先に進みます。



ポーズ／オートアドバンス



トレースバッファ内で最も古いデータを表示します。



トレースバッファ内で最新のデータを表示します。

6.2.2 バッファステータスインジケータ



現在表示されているデータの、トレースバッファ内の相対位置を示します。灰色のスライダコントロールが、現在表示されているトレースデータの位置を示します。バッファ領域は青色と白色で表されていて、青色の部分が、トレースデータが格納されている範囲を示します。

6.2.3 タイムベースコントロール



ズームイン：ビジュアライザに表示されるトレースバッファの範囲を狭めます（たとえば 50ms から 25ms にします）。



ズームアウト：ビジュアライザに表示されるトレースバッファの範囲を広げます（たとえば、25ms から 50ms にします）。

タイムベースコントロールは、すべてのビジュアライザプラグイン用のビジュアライザバッファのサイズを決定します。これはオシロスコープのタイムベースコントロールと似ています。ただし各プラグインにおいては、この範囲全体が表示されない場合もあることに注意してください。たとえば、TimeTrace ビジュアライザの 'magnify' コントロールは、バッファの一部についてのみズームインを行います。

6.2.4 トリガコントロール



トリガコントロールは、大量のトレースデータの内容を把握しようとする場合に非常に役立ちます。

左のドロップダウンリストボックスで、トリガオブジェクトを選択します。ここには、トリガに利用できるオブジェクトのリストが表示されます。その隣にあるのはトリガイベントボックスで、このコントロールの内容は、選択されたトリガオブジェクトにより異なります（たとえば、タスクの場合には 'Activate'、'Start'、および 'Stop' です）。

スライダはプリトリガコントロールです。トリガポイントより前に表示されるデータの量を調整できます。ボタンを使って、トリガイベント間を行き来したり、トリガ機能のオン／オフを切り替えたりできます。

6.3 サンプルトレースデータ

RTA-TRACE CD の ¥Example Trace Data¥ ディレクトリには、トレースデータのサンプルがあり、RTA-TRACE ビジュアライザで見ることができます。このディレクトリにはトレースデータ (.rtt ファイル) と各サンプルについての説明（データファイルと同じような名前の .txt ファイル）があります。RTA-TRACE が正しくインストールされていれば、.rtt ファイルをダブルクリックするとそのファイルが RTA-TRACE ビジュアライザで開かれます。

RTA-TRACE の詳しい使用方法については、オンラインヘルプファイルに記載されています。

お問い合わせ先

製品サポートに関しては、各 ETAS 支社までお問い合わせください。

ドイツ (ETAS 本社)

ETAS GmbH

Borsigstr. 14	Phone:	+49 (711) 8 96 61-0
70469 Stuttgart	Fax:	+49 (711) 8 96 61-105
Germany	E-mail:	sales@etas.de
	WWW:	www.etasgroup.com

日本

イータス株式会社

〒 220-6217	Phone:	(045) 222-0900
神奈川県横浜市西区	Fax:	(045) 222-0956
みなとみらい 2-3-5 クイーンズタワー C 17F	E-mail:	sales@etas.co.jp
	WWW:	www.etasgroup.com

韓国

ETAS Korea Co., Ltd.

3F Samseung Bldg	Phone:	+82 (2) 5747 016
61-1, Yangjae-dong	Fax:	+82 (2) 5747 120
Seocho-gu, Seoul	E-mail:	sungik.hong@etas.co.kr
Republic of Korea		

イギリス

ETAS Engineering Tools Application and Services Ltd.

Studio 3, Waterside Court	Phone:	+44 (0) 1283 - 546512
3rd Avenue, Centrum 100	Fax:	+44 (0) 1283 - 548767
Burton-upon-Trent	E-mail:	sales@etas-uk.net
Staffordshire DE14 2WQ	WWW:	www.etasgroup.com
UK		

フランス

ETAS S.A.S

1, place des Etats-Unis	Phone:	+33 (1) 56 70 00 50
SILIC 310	Fax:	+33 (1) 56 70 00 51
94588 Rungis Cedex	E-mail:	sales@etas.fr
France	WWW:	www.etasgroup.com

北米

ETAS Inc.

3021 Miller Road	Phone:	+1 (888) ETAS INC
Ann Arbor, MI 48103	Fax:	+1 (734) 997-9449
USA	E-mail:	sales@etasinc.com
	WWW:	www.etasgroup.com

南米

UNIT

Rua Adolfo Maraccini, 399
c/o Sergio Augusto Alves
Campinas SP 13086 - 010
Brazil

Mobile: +55 19 9772 0793
Telefax: +55 (19) 3256 1939
E-mail: unit@mpc.com.br