

チップメーカーとの戦略的パートナーシップ

オンチップデバッグインタフェースを備えたマイクロコントローラが初めて市場に登場した当時のデバッグソリューションは、その頃普及していたインサーキットエミュレータに比べると比較的単純なものでした。トリガとトレースのオプションを持たない純然たるデバッグは複雑な組み込みシステムの効率的開発には向いていないことが、まもなく判明しました。デバッグとトレースのオンチップインタフェースの用途は徐々に拡大されてきていますので、今日の開発ツールによる非常に複雑なテストと分析が可能となっています。



チップメーカーとの協力

従来、デバッグとトレースのオンチップロジックの機能性は、半導体メーカーによって定義されていました。今日では、製品を市場に投入する前にマイクロコントローラと開発ツールとの最適な相互運用を保証できるようにと、開発の初期段階でローターバッチと緊密な共同作業を望まれるチップベンダーが増加の一途をたどっています。この傾向は、プロセッサの新製品における要件として多くのエンドカスタマやOEMがローターバッチのサポートを挙げていることによって、強固なものとなっています。この結果、多数の大手チップメーカーが戦略的パートナーとしてのローターバッチに期待しており、自身の最新プロセッサファミリー用開発ツールがオンタイムで利用できるように、ローターバッチと契約を交わしています。

国際組織への参加

多くのお客様は、より高いレベルのオンチップデバッグ・トレースロジックの標準化とより少ないピンカウントが、性能の劣化なしに実現されることを望んでいます。デバッグとトレースの革新的技術の開発において積極的な役割を果たすため、ローターバッチは、以下のような各種の国際組織に参加しています：

- ローターバッチはNexus 5001™ Forumの設立時以来のメンバーであり、NEXUS仕様に準拠するツールを世界で初めて発表しました。
- MIPI Allianceのテスト&デバッグワーキンググループにおいて、ローターバッチは、インタフェースならびに携帯電話のテストとデバッグのツールの定義に携わっています。
- IEEE P1149.7 ワーキンググループ設立時以来、ローターバッチはJTAG標準の定義に積極的に関与しています。

2006年、ローターバッチはエンジニアリング部門のスタッフを大幅に増員、広範なプロセッサアーキテクチャを対象とした高性能開発ツールのリーダーとしての地位をさらに確固たるものとししました。本号では、2007年に発表予定の製品、および最近実現された最新かつ最強の機能の数々を紹介いたします。

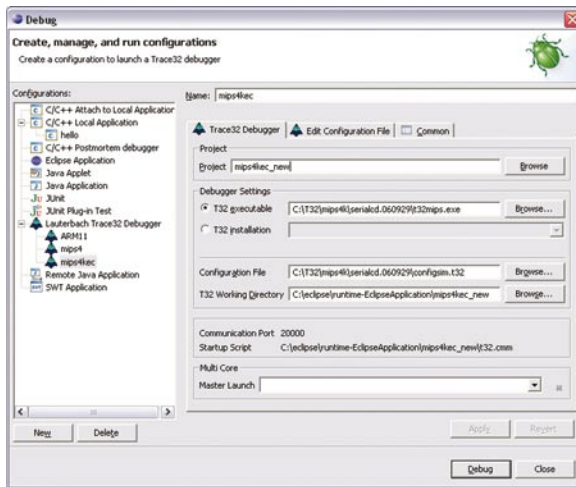
コンテンツ

新たに導入されたIDEリンク	2
組み込みLinuxのデバッグ	3
新たなサポートプロセッサ	7
Cortex-M Series用の新ツール	8
ARM-ETM関連ニュース	10
エネルギー消費の最適化	11

新たに導入されたIDEリンク

Eclipse

2006年9月以来、ローターバッハはEclipse C/C++ Development Tooling (CDT)との連携をサポートしていますが、CDTへの完全統合化が2007年10月に予定されています。



CDT リンク

TRACE32は、Eclipse version 3.1.2のCDTリンクをサポート、以下の機能を提供しています。：

- TRACE32デバッガをEclipseから直接起動できます。複雑なマルチコア/マルチプロセッサ構成もサポートされています。
- デバッガのブレークポイントをEclipseソースコードエディタで設定できます。
- ユーザーは、ソースコードが入っているあらゆるTRACE32ウィンドウからEclipseソースコードエディタ内の対応行にジャンプできます。

完全統合

CDTデバッガへの完全統合に備えるため、ローターバッハはDebugger Service Framework (DSF)に関する協力を行っています。DSFの目標は、CDTデバッガとハードウェアデバッグバックエンドとの高性能標準インタフェースを開発することです。ローターバッハは早期採用者としてDSFによる完全統合を2007年10月までに提供する予定です。

ただし、CDTデバッガへの完全統合がローターバッハGUI PowerViewに取って代わるわけではありません。

詳細につきましては下記URLをご参照ください：

<http://www.lauterbach.com/inteclipse.html>

Windows CE Platform Builder

2006年12月の発表の通り、TRACE32開発ツールを、ハードウェアデバッグバックエンドとしてWindows CE Platform Builderで利用することができます。

Extended Debugging Interface (eXDI2)によってTRACE32開発ツールをWindows CE Platform Builderのデバッグインタフェースに接続するドライバが、統合用に開発されました。

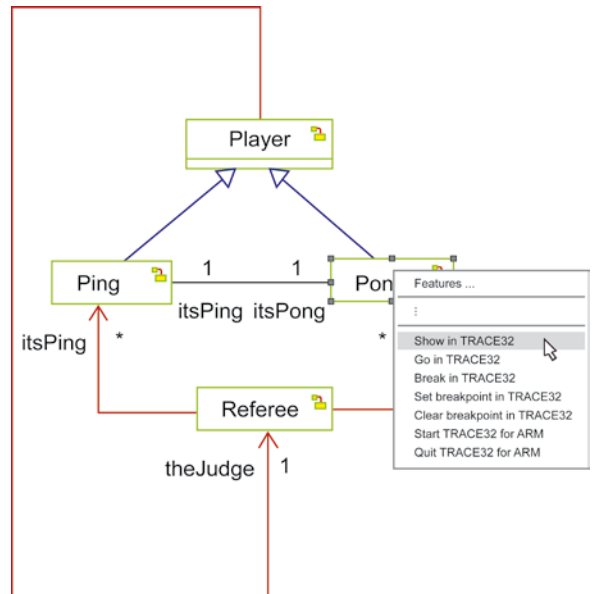
Rhapsody

ローターバッハは、I-Logix社のModel Driven DevelopmentプラットフォームRhapsodyを対象とした統合を2007年に予定しています。

統合の第一段として、次の機能を予定しています：

- TRACE32開発ツールをハードウェアデバッグバックエンドとしてRhapsodyで使用できるようにします。
- ユーザーは、ソースコードが入っているあらゆるTRACE32ウィンドウからRhapsody要素への分岐を行えるようになります。

2007年の夏までには、TRACE32が集録したリアルタイムトレースデータからメッセージシーケンス図を生成することも可能となります。



組み込みLinux用 実行・停止モード統合型デバッグ

組み込みLinuxアプリケーションの開発では、2種類のデバッガを使用することがこれまでの一般的な方法でした。ターゲットハードウェアの起動には、JTAGデバッガが通常使用されます。組み込みLinuxの基本コンポーネントがターゲット上で起動されると同時に、デバッグプロセスはGDBによって続行されます。

ローターバッチは、この2つのデバッグコンセプトを結合した統合型LinuxデバッガをEmbedded Systems Conference 2007で紹介する予定です。このデバッガにより、2つの方法の利点を単一のユーザーインターフェースで利用できることになり、組み込みLinuxアプリケーションの開発時間を大幅に短縮できます。

ARMアーキテクチャを例として使用した統合型Linuxデバッガのコンセプトを、以下に示します。

停止モードデバッグ

JTAGデバッガは、いわゆる「停止モードデバッグ」で機能します。プロセッサは、当然システム全体も、ブレークポイントで停止します。プロセッサやターゲットハードウェアの状態に関する情報は、JTAGインターフェースによって読み取ることができます。(図1参照)

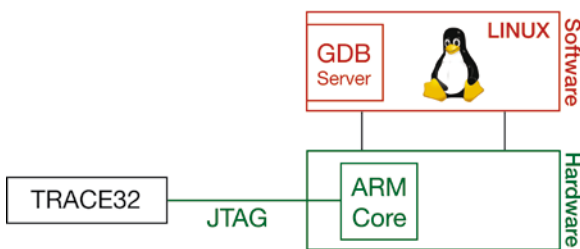


図1: 「停止モードデバッグ」では、JTAGインターフェースを介してプロセスとシステム全体が停止されます。

「停止モードデバッグ」の長所:

- 「停止モードデバッグ」における要件は、JTAGインターフェースが機能することだけです。このため、リセットベクタからのデバッグが可能です。
- LinuxとMMUの両方をサポートするデバッガにより、カーネルおよびプロセス境界を越えての

実行&停止モード デバッグの統合

デバッグエージェントとしてGDBを用いる組み込みLinux

ARM

DCCによる実行モードデバッグをサポート

ARM

Ethernetによる実行モードデバッグは2007年第2四半期にサポート

PowerPC

Ethernetによる実行モードデバッグは2007年第2四半期にサポート

デバッグエージェントとしてTRKを用いるSymbian OS

ARM

利用可能なDCCによる実行モードデバッグ

デバッグが可能です。

- ソフトウェアが反応しなくなった場合、プロセッサが誤停止したコード内の箇所をプロセスを停止して調べることができます。
- プロセッサが停止しても、カーネルや他のプロセスが問題を引き起こすことはありません。

ただし、「停止モードデバッグ」には、大きな短所が1つあります:

それは、プロセッサが停止すると同時にすべて通信インターフェースも停止する、という点です。この結果、Ethernet、Bluetooth、またはCANによってLinuxアプリケーションと通信する外部デバイスは、アプリケーションが応答していないことから接続を切断してしまいます。したがって、ブレークポイントでの停止がシステム全体の状態の変更を引き起こす恐れがあります。この場合、デバッグを続行しようとしても意味がない可能性があります。

PowerView

PowerDebug

PowerTrace

PowerProbe

Power-Integrator

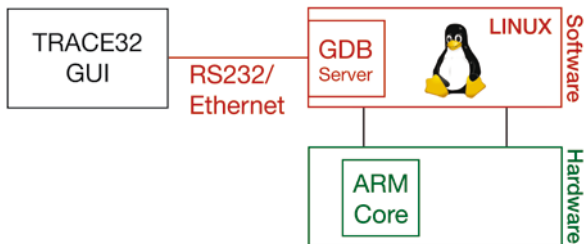


図2: 「実行モードデバッグ」では選択されたプロセスは停止しますがシステム全体は実行し続けます。

実行モードデバッグ

GDBはいわゆる「実行モードデバッグ」を行います。つまり、ブレークポイントでは、選択されたプロセスだけが停止し、カーネルや他のプロセスはそのまま実行を継続します。

ただし、GDBは完全なソフトウェアデバッガです。このため、以下の要件が揃わなければデバッグを行うことができません：

- ターゲットマシン上で一つのLinuxプロセスとして実行されるGDBサーバー
- ホストマシン上で実行されるデバッガソフトウェア - ここでは TRACE32 (図2を参照)

TRACE32は、RS232やEthernetインタフェースを介しGDBサーバーと通信、その時点で停止しているプロセスに関する情報について問合せます。

「実行モードデバッグ」は、以下の条件が整っているときに完全に機能します：

- ターゲットハードウェアの起動が完了している。
- GDBサーバーがいつでもアクティブな状態に移行できる。つまり、通信インタフェースが正しく実行されており、かつプロセッサがコードポイントで誤動作してハングアップしていない。

言うまでもなく、どちらのデバッグ手法も長所と短所を併せ持っています。そこで、ローターバッチではこの2つの手法の利点をすべて維持しつつ、弱点を完全に排するように両者を融合したデバッガを開発しました。

実行・停止モード 統合型デバッグ

TRACE32デバッガは、組み込みLinux向けに「実行・停止モード統合型デバッグ」機能を搭載し、以下のように動作します：

1. TRACE32デバッガはJTAGインタフェースを経由して「停止モードデバッグ」状態で立ち上がります。最初のステップでは、ターゲットハードウェアと「実行モードデバッグ」(GDB)の設定を行います。

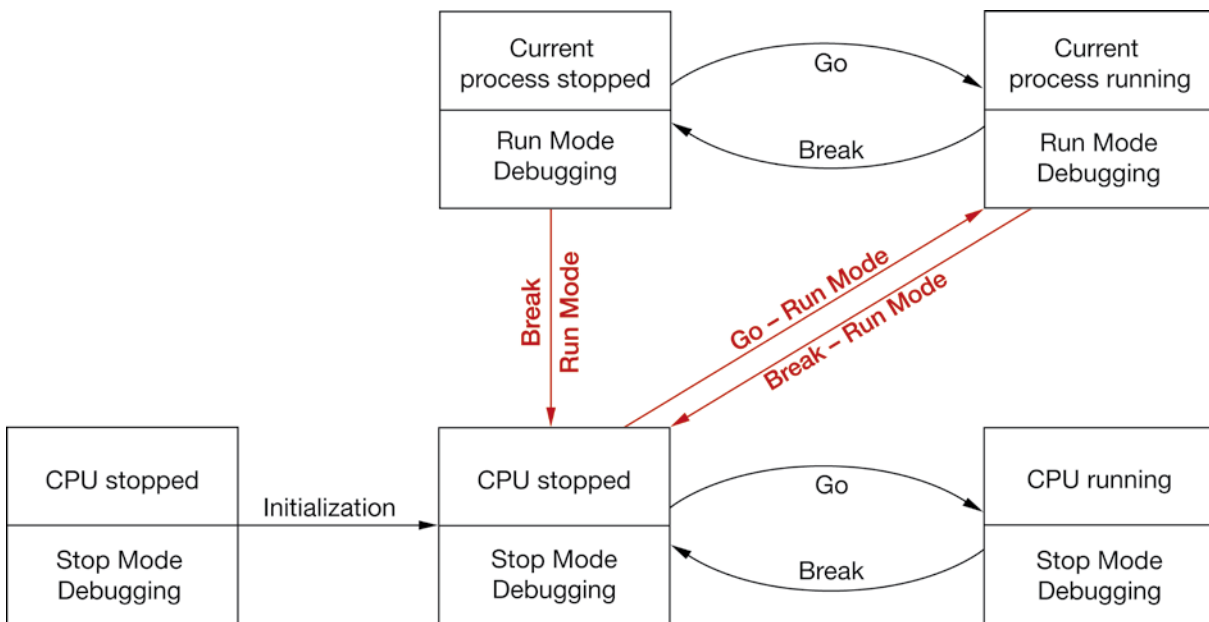


図3: テストを実行するには、要件に応じて実行モードデバッグまたは停止モードデバッグを選択します。

- ターゲットハードウェアの起動ステップには、「停止モードデバッグ」(JTAG)が使用されます。
- ハードウェアの起動が完了すると、TRACE32は「実行モードデバッグ」(GDB)に切り替わり、アプリケーションのデバッグを行えるようになります。全システムが動作している状態で、個々のプロセスをテストすることが可能です。
- 「実行モードデバッグ」中にGDBサーバーとの接続が絶たれた場合は、いつでも「停止モードデバッグ」に切り替えて、問題の原因を突き止めることができます。

「実行・停止モード統合型デバッグ」の実装と並行して、以下の機能が「実行モードデバッグ」に追加されています：

- ARMアーキテクチャに対応するため、通信インタフェースとしてEthernetとRS232のほかに、デバッグ通信チャンネル(DCC)を追加できるようになりました。この対策により、「実行・停止モード統合型デバッグ」はJTAGインタフェースのみで機能します(ヘッドレス、通信ポート不要)。
- 必要があれば、2つ以上のプロセスを同時にデバッグすることが可能です。

通信インタフェースとしてのDCC

ARMアーキテクチャ用JTAGインタフェースには、いわゆるDebug Communications Channel (DCC) が組み込まれています。原則として、

- ホスト上のデバッグソフトウェア (TRACE32)
- ターゲットシステム上のアプリケーション - ここではGDBサーバーを介して

との間で、アプリケーションがプロセッサ上で実行中に、DCCによる情報交換が可能です。

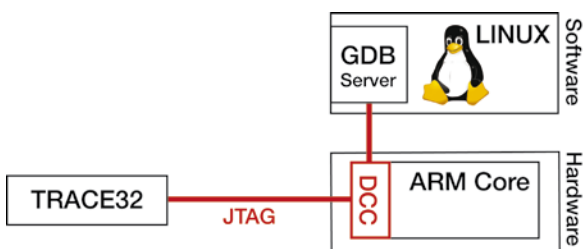


図4:外部通信インタフェースの代わりに、JTAGインタフェースのDCC機能をGDBサーバーとの通信チャンネルとして使用できます。

TRACE32がJTAGインタフェースのDCC機能を使

用して現在停止状態にあるプロセスに関する情報をGDBサーバーに問い合わせれば、「実行モードデバッグ」において外部通信インタフェースは不要となります(図4参照)。

複数プロセスの同時デバッグ

複数のプロセスを同時にデバッグする必要性が生じることがあります。これを実現するため、ローターバハでは現在、「実行モードデバッグ」用のT32Serverを提供しています。

ターミナルウィンドウからLinuxプロセスとしてT32Serverを起動すると、TRACE32コマンドによって次の操作を行います：

- プロセスの起動 (TASK.RUN)
- 実行中プロセスへの接続 (TASK.SELect)
- プロセスの終了 (TASK.KILL)

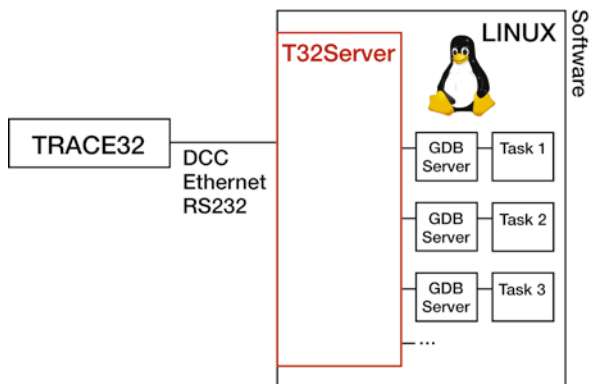


図5: T32Serverを使用すれば、プロセスごとに別々のGDBサーバーを割り当てることができるので、複数のプロセスを同時にデバッグできます。

プロセスが起動/接続されると、T32ServerがプロセスごとにGDBサーバーを割り当てます(図5参照)。

次ページの図6は、TASK.Listウィンドウの例を使ったTRACE32「実行モードデバッグ」を示しています。

まとめ

「実行・停止モード統合型デバッグ」は、1つの開発ツールと1つのユニバーサルユーザーインタフェースを使用することによって、ハードウェアとソフトウェアの複雑なエラーの迅速な発見を可能にします。しかも、アプリケーションやLinuxに

PowerView

PowerDebug

PowerTrace

PowerProbe

Power-Integrator

手を加える必要なく、組み込みLinuxアプリケーションの効率的開発のための最適な基盤となります。

「実行・停止モード統合型デバッグ」は2006年11月よりARMアーキテクチャ向けサポートが開始されており、すべてのARMプロセッサ用TRACE32 JTAGデバッグで無償でご利用できます。PowerPCアーキテクチャのサポートは2007年5月です。

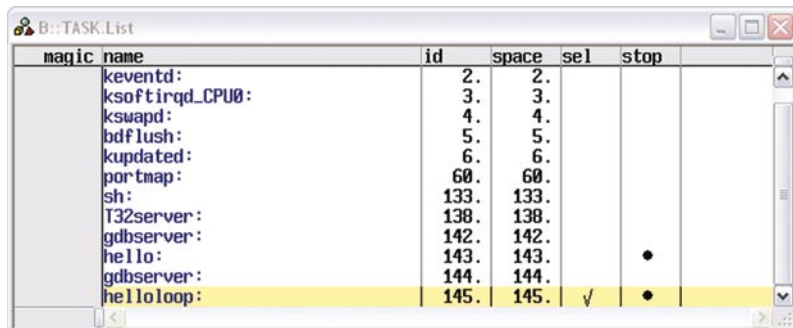


図6: helloとhelloloopのプロセスが停止されています。helloloopプロセスは、デバッグ対象として現在選択されているプロセスです。

その他のLinuxニュース

通信チャンネルとしてDCCを使用する端末ウィンドウ

2006年7月より、すべてのARMアーキテクチャを対象として、Debug Communications Channelによる端末ウィンドウの操作が可能となっています。

GDBフロントエンドとしてのTRACE32

2006年10月より、TRACE32ユーザーインタフェースをGDBデバッグのフロントエンドとしても使用できるようになっています。

標準GDBサーバーをデバッグエージェントとして使用し、個々のプロセスをデバッグすることができます。T32Serverの利用で、複数のプロセスを同時にデバッグできます。「実行・停止統合型デバッグ」と異なり、TRACE32 GDBフロントエンドではデバッグハードウェアが不要です。純然たるユーザーインタフェースであるこのフロントエンドはプロセッサに依存せず、現時点でARM、PowerPC、MIPSのアーキテクチャをサポートしています。利用できる通信インタフェースは、RS232とEthernetです。

RTOSサポート状況

CMX RTX for PowerPC	サポート中
DSP / BIOS for TMS320C64xx	サポート中
eCOS for MIPS	サポート中
FreeRTOS for ARM	サポート中
MQX 2.5 for ColdFire	サポート中
NetBSD for PowerPC	サポート予定
NORTi for Nios II and PowerPC	サポート中
OS9 for PowerPC	サポート中
OSE Delta 5.2 Load Modules	サポート中
PikeOS for PowerPC	サポート中

PrKERNEL for ARM	サポート中
QNX 6.3.2	サポート中
Quadros for C16x and StarCore	サポート中
SMX 3.7	サポート中
T-Kernel for MIPS	サポート予定
ThreadX V5, also for Nios II	サポート中
VxWorks 6.x	サポート中
Windows CE 6.0	サポート予定
μClinux for MicroBlaze	サポート中
μC/OS-II for MIPS	サポート中

新たなサポートプロセッサ

New Architectures

Analog Devices	BlackFin®	サポート中
ARC	ARC® 600 Core	サポート中
Freescale™	MPC8641D	サポート中
Infineon	XC800 Familie	サポート中
Marvell	ARM CPU Feroceon	サポート中
Mentor Graphics™	M8051EW	サポート中
Tensilica	Diamond Standard Processors	サポート中
	Xtensa Processors	Q2 / 2007
Texas Instruments	TMS320C2000	Q2 / 2007
Xilinx®	MicroBlaze™	サポート中

RISC/CISC

ローターバッハは、デバッグとリアルタイムトレースの高品質ツールメーカーの第一人者として、世界的に認められています。伝統的に、ローターバッハの開発ツールは、複雑な32ビットRISCアーキテクチャを使用した組み込み設計のデバッグにおもに使用されています。2006年に当社が最も成功を取めた製品は、FreescaleのMPC55xxファミリー用とARM9用のデバッグ・トレースツールでした。

ローターバッハはそのデバッグにより、コスト削減が大きな命題とされる設計でおもに使用される数多くのプロセッサアーキテクチャもサポートしています。具体的には、ARM社Cortex-Mシリーズのほか、2007年サポートのInfineon社XC800ファミリー、Mentor Graphics社M8051EWなどです。

DSPs

2003年以降、ローターバッハはデジタルシグナル

プロセッサ (DSP) 用デバッグの種類を増大に力を入れてきました。マルチコアプロセッサにおけるDSP利用の増加にローターバッハが対応できたのは、このためです。

Texas InstrumentsのDSP用デバッグのほか、CevaのDSP用やStar-Core DSP用の開発ツールはすべて、2006年に力強い成長を見せました。

ソフト コア

2005年、ローターバッハは、初めてのソフトコア用デバッグ・リアルタイムトレースツールを発表、下記機能を有する、Altera NIOS II用プロフェッショナルツールが開発されました。

- シングル及びマルチプロセッサ用JTAGデバッグ機能
- プログラム・データフローを記録するためのオンチップ/オフチップリアルタイムトレース機能

2006年8月には、Xilinx MicroBlaze用高性能デバッグが発表されました。このコアを対象としたオフチップリアルタイムトレースは、2007年に予定されています。

コンフィギャラブル コア

ARC600コアファミリー用デバッグで、ローターバッハはいわゆるコンフィギャラブルコアを初めてサポートしました。コンフィギャラブルコアのプロバイダは、エンドアプリケーションに合わせて最適にカスタマイズされたSoC (system on chip)の開発の可能性を顧客に提供しています。たとえばユーザーは、コアレジスタ数、演算ライブラリ、キャッシュの構造とサイズ、内部プロセッサバスをはじめ、コアの多くの要素を構成することができます。

構成可能コアの場合、デバッグにとっては次の2つの新たな課題があります：

1. デバッグ性能を保証するには、利用可能なリソースすべてが、コアに保存されている構成記述の中で正確に特定できなければならない。
2. エンドアプリケーション用に開発されたユーザー命令を逆アセンブラに対して指定しなければならない。Tensilicaの構成可能Xtensaプロセッサは、2007年のサポートです。

PowerView

PowerDebug

PowerTrace

PowerProbe

Power-Integrator

新たなサポートプロセッサ (続き)

New Derivatives

AMCC	PPC405 - PPC405EZ PPC44x - PPC440 EPx/GRx
Broadcom	MIPS64 - BCM1250/BCM1480
Ceva	TeakLite/OAK - XpertTeakLite-II Ceva-X - Ceva-X1622/X1641
Freescale™	MCS08 - MC9S08AW/DZ S12X - S12XE - S12FR MPC5500 - MPC551x PowerQUICC II - MPC831x ARM11/StarCore - MXC91231 - MXC91321 - MXC91331

IBM	PPC74x/75x - PPC750CL
Infineon	Tricore - TC111x/TC116x
Marvell	XScale - PXA3xx
MIPS	MIPS32 - MIPS34
NEC	V850 - V850ES/Dx3 - V850ES/Fx3/Sx3
Renesas	H8S - H8SX
StarCore LLC	StarCore - SC2000 - SC3000
Texas Instruments	TMS320C55X - C55x+ TMS320C64X - C64x+ TMS320C67X - C67x+

Cortex-Mシリーズ用の新しいデバッグケーブル

2007年春より、ローターバッハはCortex-Mファミリーデバッグケーブルの新バージョンを提供する予定です。最も重大なイノベーションは、標準JTAGのみならず、2ピンデバッグインタフェースであるcJTAG、SW-DPも、このデバッグケーブルがサポートするという点です。

cJTAG

cJTAG (IEEE P1149.7)は、MIPI Alliance Inc.によって定義されたオンチップデバッグインタフェースです。5ピン標準JTAGインタフェースの代替品として、2ピンインタフェースは、クロックライン1本と双方向データライン1本とで構成されると定義されています(9ページの図7参照)

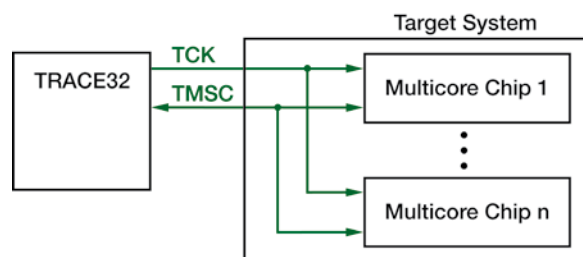


図7: ターゲットハードウェアがマルチチップである場合のcJTAGインタフェース

シリアルワイヤ デバッグ ポート(SW-DP)

デバッグ・トレース機能に CoreSightテクノロジーが採用されている SoC (system on chip) の場合、外部デバッガは従来のように個々のコアの TAP コントローラと直接通信するのではなく、いわゆる Debug Access Port (DAP) を介して通信します。DAP の役目は、デバッグコマンドを個々のコアに送ることです。オンチップデバッグがコアにどのように実装されているかに従い、通信は下記のように行われます：

- メモリマッピングされたデバッグレジスタを持つコアは、バスアクセスの助けを借りて制御されます。たとえば Cortex-M は、このメカニズムで機能します。
- 従来型の TAP コントローラによってデバッグを行うコアは、依然として JTAG シーケンスを使用して制御されます。

デバッガと DAP とのインターフェースは、ARM 規定の 2 ピン幅のシリアルワイヤデバッグポートと標準 JTAG のどちらかを使用します (図8参照)。

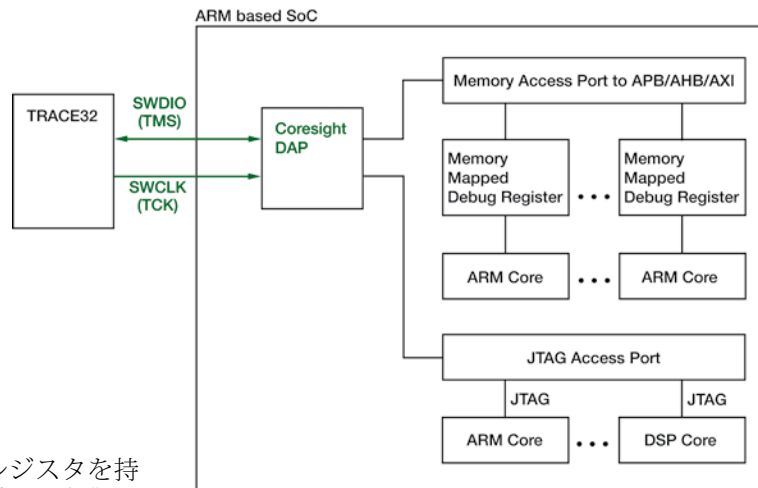


図8: デバッガとDAPの2ピンインターフェースとしてのシリアルワイヤデバッグポート

2ピンデバッグインターフェースの高速ダウンロードを実現するため、オンチップデバッグインターフェースは最大100 MHzの周波数で動作します。高周波数でのデバッグケーブルによる反射を排除するため、デバッグ信号の着信はターゲットハードウェアコネクタに移動してあります。

CombiProbe - Cortex-M用デバッガ/トレース



図9: Cortex-Mシリーズ用低価格開発ツール、PowerDebug モジュールおよびCombiProbe

2007年8月、ローターバッハはARM社Cortex-M用の低価格開発ツールを発売します。

この新しい開発ツールは、ユニバーサルデバッガハードウェアPowerDebugとCombiProbeとで構成されます。CombiProbeの最も重要なコンポーネン

トは、下記のとおりです：

- 128メガバイトトレースメモリ搭載プローブハードウェア
- ターゲットハードウェア用34ピン高速プラグ
- 34ピン高速プラグをターゲットハードウェアのデバッグ/トレースインターフェースに接続するためのアダプタケーブル

デバッグインターフェース

標準JTAGのほか、CombiProbeは2ピンデバッグインターフェースcJTAG (図7参照) とシリアルワイヤデバッグポート (図8参照) もサポートしています。

トレースインターフェース

CombiProbeシステムを使用すれば、以下のトレースログの記録と分析を行えます：

- 連続モードの4ビットETM
- CoreSight Single Wire Viewer (SWV)

最大トレース周波数は、200 MHzです。

ARM-ETMプリプロセッサ AutoFocus II



2007年始め、ローターバッハは、ARM-ETM用プリプロセッサの新バージョンを発売を開始しました。PowerTrace IIとの組合せで、500MHzを超える周波数でのテストにAutoFocus IIが合格しました。

AutoFocusテクノロジーはさらに洗練され、技術的に可能な境界でトレースポート速度のトレース信号を最適サンプリングできるようになっています。

- 64クロック・24データ遅延、78 ps分解能
- 40トレースチャンネルすべてにおける低信号間ス

シリアルETM

非常に大きなトレースポート帯域幅を数本の伝送回線のみで実現するため、ローターバッハはETMデータのシリアル伝送のコンセプトと取り組んでいます。第一世代では、最大6.25 Gbit/sのデータ速度を目標にしています。

多くのメーカーはすでにASICでGbit-PHYトランシーバを提供しているので、論理的にはトレースポートでも利用できるはずですが。これにより、トレースポート帯域幅の増加とピン数の減少とのどちらかまたは両方が可能となります。

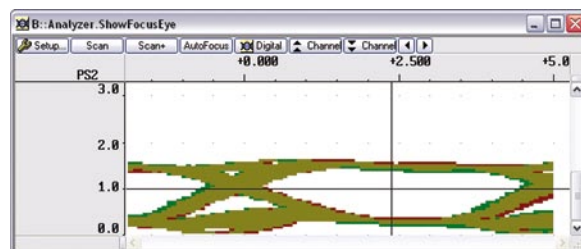
ASICメーカーと協議の上、ARM-SoCにおけるETMデータのシリアル化は、Xilinx Aurora Protocolに基づいて計画されます。

開発ツール側のシリアルプリプロセッサは、マルチギガビットトランシーバによってデータを受信し、逆シリアル化とクロック回復を処理します。この方法で回復されたパラレルETMデータは、PowerTrace IIのトレースRAMに通常どおり保存されます。

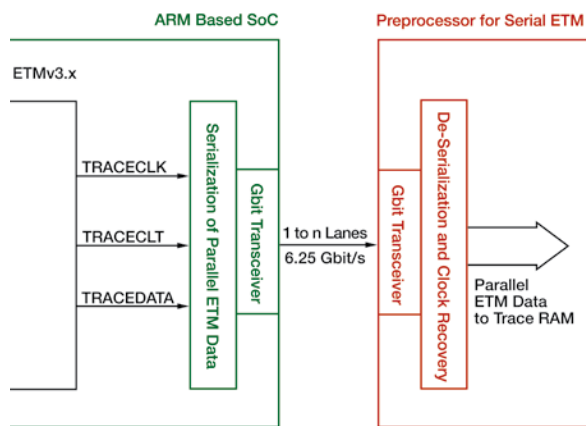
キュー

- クロック・データチャンネルの終端のデカップリング

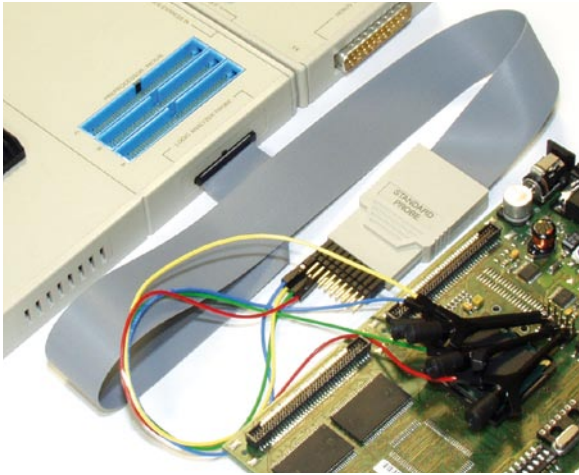
プリプロセッサ自動構成と自己テストの総合的既存メカニズムに、ターゲットハードウェア上のETMチャンネル品質をチェックする三次元アイファインダーが加わりました。このアイファインダーは、タイミング誤差範囲 $-1.8 \sim +4.9$ ns、電圧範囲 $0 \sim 3.3$ Vで全ETMチャンネルのデータアイを表示します。第三次元は、チャンネル番号です。ユーザーは、個々のチャンネルを分析することも、全チャンネルを重ね合わせて見ることもできます。このチェックの結果、個々のETMチャンネルのデータアイが平均より小さいことが判明した場合、ハードウェアレイアウトをチェックしてこれらの信号に異常な特性が見られるかどうかを調べる必要があります。信号品質悪化の原因としては、長い配線、スタブライン、ドライブ能力不足が考えられます。



現バージョンのPowerTrace IIを変更することなくそのままシリアルプリプロセッサと一緒に使用できるので、あらゆるローターバッハツールに採用されているモジュラーコンセプトの柔軟性がここでも証明されています。



PowerTrace II用ロジックアナライザプローブ



2006年7月以降、プログラム・データフローの記録だけではなく、最大17のデジタルデータチャンネルまたは4電圧/3電流チャンネルの記録にも使用できるようになっています。

PowerTrace IIのLogic Analyzer Probeコネクタでは、以下のプローブを接続できます(写真参照)：

- 最大17のデジタルデータチャンネルを記録する標準プローブ
- 4電圧チャンネルまたは3電流チャンネルを記録するアナログプローブ

PowerTrace IIのリアルタイムトレースシステムは、以下を搭載しています：

- プログラム・データフローを記録する最大2ギガバイトのトレースメモリ
- その他のデジタル/アナログ信号を記録する512kトレースエントリメモリ

この2つのトレースメモリのタイムスタンプは同一

エネルギー消費の最適化

バッテリー駆動式装置では、エネルギー消費の削減がますます重要になっています。待機時間や使用時間といった要素は、携帯電話を販売するにあたっての仕様の重要な部分です。

このため、エネルギー消費を削減する総合的手段が、バッテリー駆動式装置のハードウェア設計における不可欠な部分となっています。ただし、こうした手段が効果を発揮するのは、装置を制御するソフトウェアがハードウェアのあらゆる節電機能を一貫して活用する場合だけです。

の48ビットタイマーによって生成されるので、プログラム・データフローとその他のチャンネルとの時間関係が明瞭に表示されます。その他のチャンネルの対時間動作の分析は、テーブルやチャートを使って直観的にすばやく行えます。チャンネルデータは簡単にファイルとしてエクスポートできるので、必要に応じて外部ソフトウェアによる処理を行えます。

標準プローブデータ

- 17デジタルデータチャンネル
- 電圧範囲 0-5V
- 最大200 MHzのトランジェントレコード

アナログプローブデータ

- 4電圧チャンネル 0-5V
- 3電流チャンネル (ターゲットハードウェア上の分流器が必要)
- 12ビット分解能
- 625 kHzサンプリングレート

以下について問う必要があります：

- プログラムがマイクロコントローラを常に正しい節電モードに切り替えるか？
- プログラム変更が電力消費にどのように影響するか？
- 予期しない電力ピークがあるか？

これらの問いに答えるには、プログラム・データフローの他、電圧と電流の測定と記録を行い、その結果をユーザーフレンドリーな形式で表示できるシステムが必要です(12ページに続く)。

PowerView

PowerDebug

PowerTrace

PowerProbe

Power-Integrator

エネルギー消費の最適化(続き)

システムは、以下で構成されます：

- デバッグとリアルタイムトレース
- アナログプローブとロジックアナライザ

ロジックアナライザとしては、PowerTrace IIとPowerIntegratorのどちらのロジックアナライザで

も使用できます。この測定方式の場合、電流/電圧の使用状況とプログラムフローの両方に同期タイムスタンプを使用できるので、制御ソフトウェアと現在の電流/電力消費との関係を簡単に表示し、分析することができます。

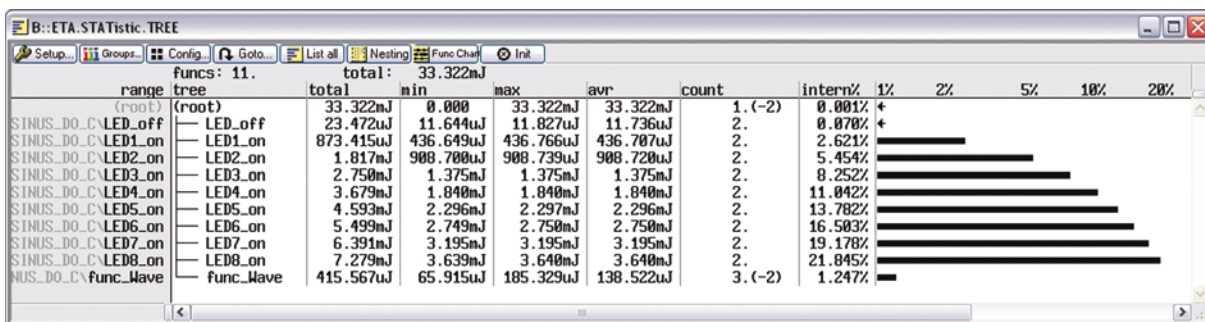


図10: 制御ソフトウェアが実行する関数のエネルギー消費の統計分析

将来の展望

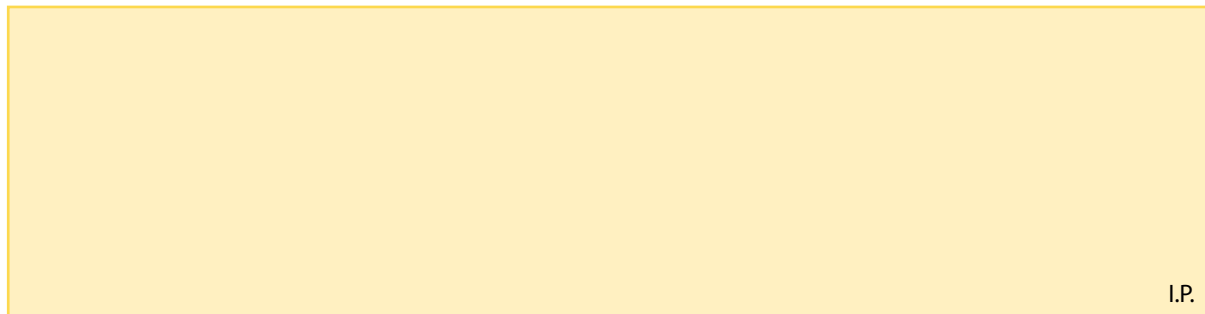
ローターバッハが広範なプロセッサアーキテクチャのための高性能開発ツールの提供において第一人者であり続けられるようにするため、当社では開発グループの規模を大幅に拡大しました。現在建築中の新社屋は2008年に完成する予定であり、

完成の暁には、今後数年間の当社の成長を収容できるはずで。

<http://www.arcone-tec.de>



図. 11: ドイツ、ミュンヘンに建築予定の本社



I.P.