

# Linux SMPや大量の実行履歴に対応した JTAG ICEを開発



マルチコアやSMP(Symmetric Multi Processing)の普及に伴って、組み込みシステム開発環境は大きな変革期を迎えようとしている。4年間でユーザーへの実売5,000台超の出荷実績を持つ京都マイクロコンピュータのJTAG ICE「PARTNER-Jet」は、こうした変革に対応するべく、現在も進化し続けている。今回は、同社 東京オフィス ゼネラルマネージャの辻 邦彦氏に、JTAG ICE製品の現状と今後の動向について話を伺った。聞き手はInterface企画室。



京都マイクロコンピュータ株式会社  
東京オフィス  
ゼネラルマネージャ 辻 邦彦氏

——Linuxが備えるSMP対応機能(以下、Linux SMP)をサポートするJTAG ICEを開発されたそうですね。

辻氏：これまでPARTNER-Jetは、Linux対応のデバッグ、Windows CEのデバッグ、マルチコアのデバッグなどのオプション機能を提供してきました。これらと同じ位置づけで、今回、Linux SMPのための機能を追加しました。

PARTNER-Jetでは、マルチコア対応はすでに実現しています。ここで言う「マルチコア対応」とは、複数のCPUコアを内蔵するSoC(System on a Chip)に1台のPARTNER-Jetを接続すると、あたかも複数台のPARTNER-Jetを接続しているかのような感覚でデバッグできる機能です。デバイス・レベルで見るとSMP向けプロセッサもまたマルチコア・プロセッサなので、従来の(マルチコア対応の)JTAG ICEによってデバッグできます。しかし、それだけで「SMP対応」と呼ぶには不十分であると私たちは考えています。今回は、その不十分な部分を追加し、SMP対応として発表しています。現状、動作を確認しているSMP対応のマルチコア・プロセッサは、「ARM11 MP Core」,「SH-4Aのマルチコア版」,「AM34(松下電器産業製)」などです。今後、MIPSの動作検証も行う予定です。

——具体的には、従来のマルチコア対応とどのような点が異なるのでしょうか。

辻氏：シンメトリックなマルチプロセッサは、それに対応したOSが搭載されて初めて、本当のシンメトリックになります。例えば、シンメトリックな構成がとれるマルチコア・プロセッサであっても、各CPUコアが個別にリアルタイムOSを搭載していたら、それはシンメトリックではない形で動作することになります。今回PARTNER-Jetが新たにサポートしたのは、「SMPデバイス」ではなく、

「SMP用OS」です。

マルチコア対応のJTAG ICEの場合、各CPUコアに対して1:1の関係でデバッグ環境が存在します。JTAGのケーブルは1本ですが、ICEは複数あるというイメージです。例えば、コア0でプログラムをデバッグする場合、最後までそのプログラムはコア0の上で動いています。つまり、ICEはコア0のデバッグだけを考えていればよいことになります。

一方SMPでは、ある瞬間にあるアプリケーションがコア0で動いていたとしても、別の瞬間にOSのスケジューラが、このアプリケーションをコア3に移動させる可能性があります(図1)。このようにSMPでは、実行スレッドがプロセッサ(コア)をまたがって移動します。これは、従来のマルチコア対応の機能では、対処できません。

SMPに対応したJTAG ICEは、ユーザから見ると、複数のCPUも一つのCPUのような感覚で取り扱うことができるようになってきました。先ほど説明した、デバッグしているスレッドが、CPUコアをまたがって移動した時でも、そういったことを全く意識せずに、一つのデバッグでデバッグすることができます。これは、PARTNER-JetがCPUの切り替えに対して、その変化に自動的に追従することができるからです。

よく質問されるのが、Linuxサーバなどで使われるGDBデバッガとどう違うのかということですが、サーバ側のGDBでデバッグする際には、Linuxカーネル上でデバッガを動かしています。カーネルはアプリケーションをどのコアに移動させたかを自分で知っているのだから、デバッグできるわけです。一方今回の機能は、カーネル内ではなく、外からデバッグしている点が大きな違いです。外からデバッグすると、デバイス・ドライバやOSと一緒にデバッグできます。「外のツールがカーネルの内部を解析してこれに

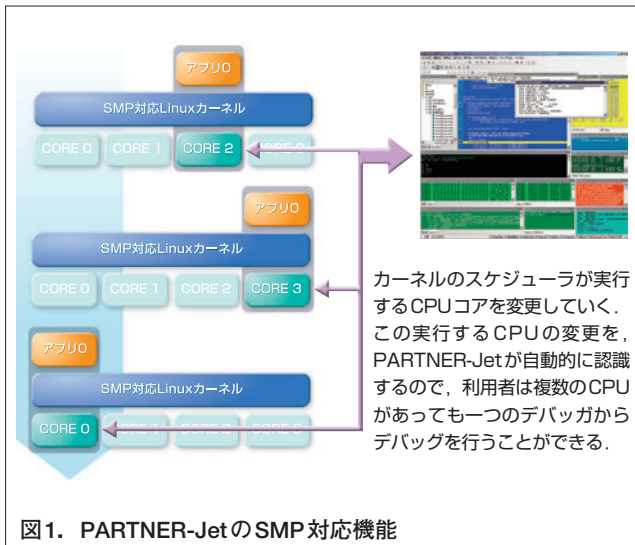


図1. PARTNER-JetのSMP対応機能

追随する」これが今回の機能の技術的なポイントです。

——次に、新たに発売されるPARTNER-Jetの最上位機種について教えてください。

辻氏：弊社は過去4年間に、PARTNER-Jet M10, M20, M30を提供してきました。今回新たに最上位機種として、「ギガトレース機能」を搭載したPARTNER-Jet M40を提供します。

最近の傾向として、バグの解析や性能解析に対して非常に大きな実行履歴を残したいという要求があります。M30でも、ARM11で160万命令程度の履歴を残せたのですが、現在ではギガ・オーダ(数十億命令)の履歴を記録することが要求されています。

M40は、ギガ・オーダの履歴の記録に対応しています。これにより、例えば10秒～20秒の実行履歴を全部記録できます。携帯電話でいえば、起動から待ち受け画面表示までの全処理を記録できることもあります。これが実現できると、どの時間にどの処理を行っていたかを知ることができます。つまり、バグ解析だけでなく、性能解析にも利用できるわけです。弊社としては、PARTNER-Jet M40のメインの用途は、性能解析になるだろうと予測しています。

——他社のツールとはどこが異なるのでしょうか。

辻氏：ただ単に記録用メモリを増やただけでなく、OSのリソースやスケジューラの状態をビジュアルに表示する機能を追加しました。エンジニアが書いたプログラムがどのタイミングでどう動いているのかを、後から表示して確認できます。また、ギガ・オーダのトレースに合わせて命令レベルの実行履歴とOSレベルの実行遷移を連携して表示したり、検索したりできるようになりました。これらの機能を、既存のモデルを含む全機種で利用できるようにしますが、特にギガトレース機能搭載のModel 40では、その大容量の実行履歴とOSの履歴をあわせることで、より効果が発揮されると思っています。

タスク遷移を表示するツールは従来からありましたが、命令レベルの実行履歴と連携して表示するのは少なかった



PARTNER-Jetの外観

と思います。特に弊社では、LinuxやWindows CE 6のような、ユーザランド(OSの中のカーネル以外の部分)で実行アドレスが多重化された環境でも、命令レベルの履歴を再現することができており、ITRONなどのRTOSだけでなく、MMUの機能を最大限に使ったOS環境においても、命令履歴とOS履歴の両方を併せてみられることは重要だと思います。これらをビジュアルに表示・検索するツールを提供します。

——統合環境のEclipseには対応していますか？

辻氏：Eclipse対応については、以前から発表していましたが、今回ようやくβ版から正式版に移りました。これまでのようにEclipseからgdbやgdbserverを介したり、Eclipseはエディタとして利用し、別にデバッガを立ち上げたり、といった方式ではありません。C/C++開発支援用プラグインであるCDTを直接PARTNER-Jetにつなぐ方式をとっています。PARTNER-JetがGDBプロトコルを実装しているので、GCC以外のコンパイラを扱えますし、Eclipseでデバッグもできます。あくまでもEclipseだけで、ビルドからデバッグまで行える環境となっています。

——貴社の今後の方向性について教えてください。

辻氏：ありきたりですが、「見えないものを、見えるようにする」、これが京都マイクロコンピュータの仕事です。元々デバッガは目に見えないCPUの動きを見えるようにする道具です。近年は、さらにマルチコア化やOSの搭載で、開発者からは見えにくい部分、見えない部分が増えていきます。PARTNER-Jetは、開発時からのべ5年以上経っていますが、その時点からSMP対応、マルチコア対応、LinuxやWindows CEへの対応を視野に入れてきました。その意味では、4年前に購入していただいた製品でも、特にH/Wなどを変更することなく最新の機能が使えます。PARTNER-Jetの発表時にロードマップを提示し、「成長し続けます」と説明していますが、今のところその言葉通りに毎年進化しています。今後もそれは変わらないと思います。