

組込み分散リアルタイムシステムを20年にわたり研究 科学技術を結集し理論どおりに動作するシステムを設計可能に

「リアルタイム」をキーワードに、プロセッサアーキテクチャ、ネットワークアーキテクチャ、並列分散処理、オペレーティングシステム、ネットワークミドルウェア、システムLSI等々すべて独自開発し、ヒューマノイドロボットや宇宙機システムとして活用できる環境を構築している山崎研究室。JASA主催展示会でも注目を集める分散リアルタイム処理CPUをはじめ、その研究概要を紹介する。

分散リアルタイム制御に必要な機能のすべてを網羅

「難しくなければ研究にならない。普通ではできないことを実現するのが科学技術」と話す山崎信行氏。それらの成果は、多数の著作物や国際論文誌での発表、国際標準化といったかたちで世界に示されている。

例えばロボティクス。蹴っても倒れないヒューマノイドロボットをいま目にしても大きな驚きはないかもしれないが、それが10年以上前なら、誰もがその技術に目を見張ったはずだ。「仮に1ミリ秒単位でタスクを実行していたら普通の制御ならばそれでOKですが、そうしたロボット制御はできません。周期が長くジッターが大きいとQoSが低くなり精密な制御は無理で、例えば100マイクロ秒と非常に短い周期でジッターレスに制御することによって可能になる技術です」

宇宙機制御やロボット制御にはリアルタイム性や応答性、AI処理、消費電力などいろいろな要素が関係するが、汎用のCPUやネットワーク、OSなどで容易に実現できるものではない。それらすべてを研究し、新たに開発することが必要となる。それら先進技術をIPまたは方式として供給可能にしているのが研究室の大きな特徴だ。「上から下まであらゆるIPを開発しています。持っていないのはファブだけです」

主な研究開発の概要は以下となる。

- 分散リアルタイム制御に必要なすべての機能を集積したプロセッサ(CPU)「RMTP (Responsive Multithreaded Processor)」
- 分散リアルタイム制御に必要な機能を1チップに集積したSoC(System-on-Chip)
- PCの機能+組込みマイコン+リアルタイム処理機能+リアルタイム通信機能を集積したSiP(System-in-Package)

●効率的な分散制御を可能にする、ISO/IEC 24740として国際標準化されているリアルタイム通信規格「Responsive Link」

●リアルタイムタスクを周期実行する機構を実装したRMTP向けリアルタイムOS「FavorOS」

●開発環境として自前のCPU用のコンパイラ、デバッグ、ILS等

「主には国のプロジェクトとして研究してきました。以前からプロセッサやネットワークのアーキテクチャ、OSなど基盤技術は、日本として自前でソースコードを持っておくべきという考えで、開発したそれらのシステムは組込み用としては機能的・性能的に従来のものを上回るものになっています」

理論どおりに動作するリアルタイムシステムを追求

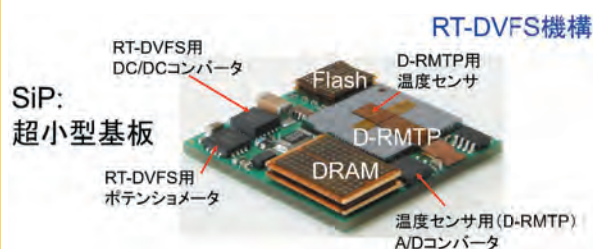
リアルタイム処理や通信は、デッドラインや周期等の時間制約を優先度に変換し、実行中のタスクを一時停止して優先度の高いタスクに移るというプリエンプティブに実行する。そのため演算処理や通信をおこなうハードウェア、ソフトウェアはプリエンプティブであることが求められる。その実現は、さまざまなタスク間で対象を切り替えるコンテキストスイッチ、優先度の高いパケットが低いパケットの追い越しをおこなう通信制御で可能になる。このプリエンプションにともなうオーバーヘッドの時間が“0(ゼロ)”に近づくほど理論どおりに動作することになる。

「理論どおりにトップダウンに、分散リアルタイムシステムを設計可能にすることが研究開発の第一の目的です」

リアルタイム処理は国内と海外ではまったく異なっているという。「欧米ではリアルタイムOSを使って、EDF (Earliest Deadline First) やRate Monotonicといったスケジューラベースでおこないます。例えばLinuxだとコンテキストスイッチのオーバーヘッドが8000クロックほどあり、日本の企業はタイムテーブルを書いてTDMなどでボトムアップに対応しますが、それは理論どおりに動かしているわけではありません。われわれはリアルタイムスケジューラで出したものが制約を守り、そのまま

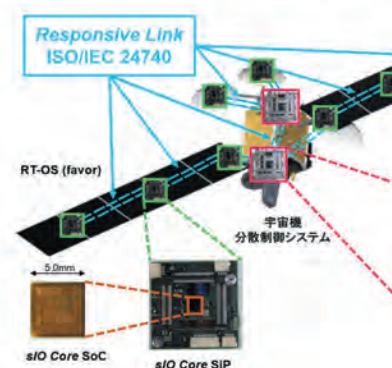
リアルタイム処理は国内と海外ではまったく異なっているという。「欧米ではリアルタイムOSを使って、EDF (Earliest Deadline First) やRate Monotonicといったスケジューラベースでおこないます。例えばLinuxだとコンテキストスイッチのオーバーヘッドが8000クロックほどあり、日本の企業はタイムテーブルを書いてTDMなどでボトムアップに対応しますが、それは理論どおりに動かしているわけではありません。われわれはリアルタイムスケジューラで出したものが制約を守り、そのまま

ロボット制御用20mm角D-RMTP SiP



PCのほぼすべての機能+組込みマイコン+リアルタイム処理機能+リアルタイム通信機能を20mm角に集積

宇宙機の分散制御例



YAMASAKI NOBUYUKI

教授/博士(工学) 山崎 信行氏

慶應義塾大学 理工学部 情報工学科



時間どおりに動作する理論、ソフトウェア、ハードウェアを一式研究開発しています。例えばコンテキストスイッチのオーバーヘッドをほぼ0にすることができます」

ソフトウェアで動的にスレッド生成する特徴的なCPU

RMTPは通常のプロセッサとはまったく異なる特徴を備えている。大きな特徴は、ソフトウェアで動的に論理コアの生成、消滅、制御できる点だ。インテルならBIOSで設定するとブート時に1コア当たり2スレッドまで同時実行できるが、RMTPは最大8スレッドまでソフトウェア命令で同時実行スレッド(論理コア)数を制御できる。

「RMTPは従来のプロセッサとしての機能はすべて備えながら特別な機能を付け加えたものです。コンテキストスイッチのオーバーヘッドを除去するため、優先度付きSMT実行に変換します。要はひとつの物理コアに最大8つの論理コアがあり、それら論理コアにソフトウェア命令で優先度を付けることで、最大8タスクを優先度順に同時リアルタイム実行します。ブート時は1物理コア1論理コアですが、mkthというスレッドを生成する命令で、プロ

グラムカウンタとかGPR、FPRなどレジスタセットで構成される優先度付きスレッド(論理コア)をソフトウェア命令で8個まで動的に生成や消滅できます。

また、コンテキストスイッチをおこなうハードウェア機構として最大32スレッド分を格納できるコンテキスト専用のキャッシュを備え、わずか4クロックでコンテキストスイッチができます。つまり1物理コアで最大8スレッド同時実行および40スレッド並行実行がハードウェアでサポートされています」

論理コアが動いていないときは電源や周波数は供給されず、電力量を抑えることができるという。また、割り込みによるスレッド起床、各スレッドの実行速度を制御するIPC制御機構といった特徴もある。

「組込みにおいて本来は敵のような割り込み処理を我々は味方にしていますが、外部割り込みやタイマー割り込みにそれぞれ対応するスレッドが1クロックで起床します。また、IPC (instructions per clock (cycle))はそのプロセッサの性能を図るためのものですが、IPC制御機構はハードウェアで各スレッドの実行速度を一定に保つことができる機能です。優先度を付けても時間とともに実行速度は変動する

ため、いつまでに終わらせるか保証ができません。ソフトウェアで速度設定をすると、ハードウェアがCPUのなかで各プログラムの実行速度がその設定値になるようフィードバック制御する設計をしています。例えばモーター制御するプログラムが2000クロックサイクルあれば実行できるとわかっているなら、実行時間が保証できる速度で一定に保つことで、非常に正確にリアルタイム処理がおこなえるというものです」

さらに、電源が遮断されても演算途中から再開できる不揮発性ノンストッププロセッサ(NV IO Core)、ノイズがある環境下でもエラー訂正しながら通信できるラインコードなど、関連する研究開発は枚挙にいとまがない。4b10bラインコードは今年中に国際標準化される見込みという。

研究室はソースコードがそろったホワイトボックス

山崎氏の研究開発の一端を取り上げただけでも感服するものばかりだが、まだまだ開発実績は豊富にある。ぜひEdgeTech+展の展示ブースなどでその実際を体験していただきたいと思う。

本来ならブラックボックスになりがちなOSやCPU、ネットワークなどすべて独自開発していることで、研究室においてはすべてがホワイトボックスになる。「すべてのソースコードがあって研究開発しているので、学生はものすごく良い経験になっています。みんな好きな分野を見つけて趣味で研究しているようなものです。知識や技術がしっかり身につく卒業生は即戦力以上です」ということで、大手メーカーからの注目も高い。

山崎氏本人は「これまでは主に国のプロジェクトとして続けてきましたが、これからは日本のメーカーへ技術移転して還元していくことを考えたい」と言う。ぜひ実践していただくことをお願いした。

ハードウェアによるリアルタイム実行

複数のパイプラインと演算機で優先度順に同時実行

