

第7回（社）計測自動制御学会
システムインテグレーション部門 講演会

7th SICE System Integration Division
Annual Conference

SICE
SI 2006

講演概要集

SICE[®]



2006. 12. 14-17 札幌コンベンションセンター

組込プロセッサ+Windows CE への RT ミドルウェアの実装

○三浦 俊宏, 水川 真, 安藤 吉伸, 土屋 裕 (芝浦工業大学)
池添 明宏 (株式会社セック), 安藤 慶昭 (産業技術総合研究所)

Implementation of RT middleware with Windows CE on Embedded Processor

○Toshihiro MIURA, Makoto MIZUKAWA, Yoshinobu ANDO, Yutaka TSUCHIYA (SIT)
Akihiro IKEZOE (SEC) Noriaki ANDO (AIST)

Abstract: In this paper, we propose porting of RT middleware to embedded system ,adopting Windows CE. It becomes possible to increase the reusability of software also to various devices This combination of Windows OS and RTM-aist is expected to improve development efficiency by using the state of the art development environment.

1. 背景

現在のロボットの多くは単純な繰り返し作業を行う産業用ロボットが大半を占めている。しかし、今後のロボットには人間の生活を支援する為の様々なサービスが要求されている。したがって、各ユーザの様々な要求を満たす、様々なロボットを提供しなくてはならない。しかし、現在のロボット開発では各製造者がそれぞれ独自の仕様でロボットの開発を行っている。そのため、ロボットには再利用性がなく開発に長い開発期間を要し、膨大な開発技術者と費用が必要になっている。こうした問題点を解消するために、ロボットの技術要素をソフトウェアレベルでモジュール化し、再利用性を高める為のミドルウェアとして RT ミドルウェア (Robot Technology Middleware)^[1]の研究・開発が行われている。

この RT ミドルウェアは、まだ開発段階にあり開発がしやすいデスクトップ PC で開発が行われている。しかし、今後ロボット技術が一般的に普及するにあたって価格・消費電力・搭載スペース等の問題や開発効率の問題を考慮すると、組込系プロセッサ上に RT ミドルウェアを実装することが必要となると考えられる。

この問題に対してわれわれは組込プロセッサ上に Linux 系 OS を使用し、RT ミドルウェアを実装する方法を検討し実装した。^[2]本報告では、組込みプロセッサ上へ RT ミドルウェアを普及させるために、windows 系 OS が搭載された組込みプロセッサに対して RT ミドルウェアを実装するための手法についての検討を行ったので報告する。

2. RT ミドルウェア

RT ミドルウェアは次世代のロボット開発用プログラム開発フレームワークであり、ロボット制御プログラムの再利用性向上のためにネットワーク分散コンポーネント化技術による共通プラットフォームを確立することを目指して、産業技術総合研究所で開発が進められている。RT ミドルウェアのシステム構成は、Fig.1 に示すように RT コンポーネント (Robot Technology Component) と呼ばれるソフトウェアコンポーネントを複数接続する構成である。このようなコンポーネントを複数つなぎ合わせることで、柔軟なシステム開発が可

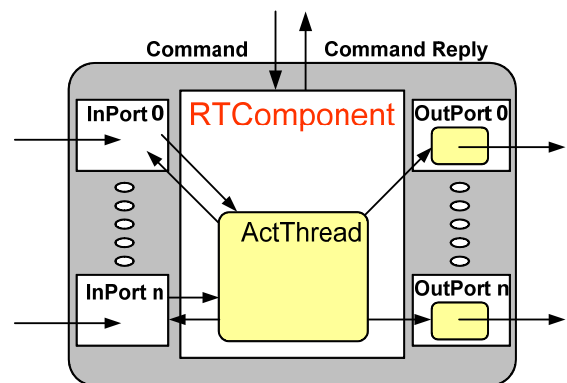


Fig.1 Composition of RT Component

能となる。コンポーネントの接続作業は GUI (Graphical User Interface) を用いて動的に構成、変更することが可能である。RT ミドルウェアは、OMG が策定している分散オブジェクト技術 CORBA をベースとしている。そのため、様々なプラットフォームでプログラムの修正なしで動作させることが可能となり、再利用性が非常に高い。

3. Windows CE への RT ミドルウェアの実装

3.1 実装によるメリット

Windows CE へ RT ミドルウェアを実装することによりロボットだけではなく組込系 Windows OS を用いた PDA やモバイル携帯など様々なものでもソフトウェアの再利用性が格段に向上する。

また、今まで実装例がなかった組込系 Windows OS の実装例を確立することにより、Windows 系 OS と Linux 系 OS の全てに RT ミドルウェアを実装することが可能となる。つまり、デスクトップ系と組込系に存在する Windows 系 OS と Linux 系 OS の様々な OS に依存することなく開発を行うことが可能となる。

3.1 実験プラットフォーム

RT ミドルウェアを実装する為の実験環境として、我々が開発している物理エージェントロボット PAR04R の統括・制御を行っているロボットコントローラサブシステムを用いる。使用する組込系プロ

セッサは、今後の PAR の研究開発を考慮し、現存する組込系プロセッサの中でも上位系のプロセッサを用いる必要がある。また、ロボットコントローラには、外部との通信に必要な Ethernet コントローラと、内部の各サブシステムとの通信に必要な CAN コントローラが必要である。以上の要求仕様より、組込系プロセッサにはルネサステクノロジ社製の SH4(SH7760)を選定し、ボードにはこの SH7760 が実装されている ZMP 社製の SH4 ボードを選定した。本ボードには、組込系 Windows OS の 1 つである Windows CE.NET 4.2 が搭載されている。SH4 ボードの概観と詳細を、それぞれ Fig.2, Table1 に示す

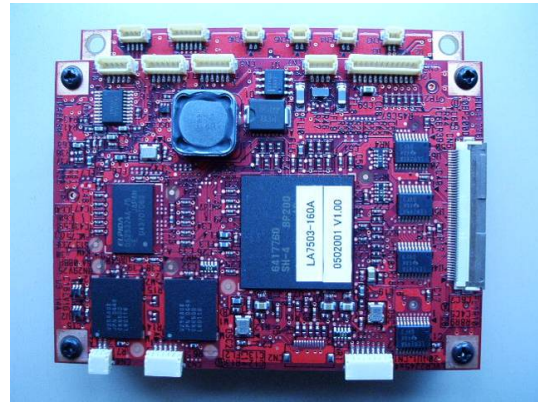


Fig.2 SH4 board of made by ZMP company

3.2 RT ミドルウェアの実装方法の検討

ZMP 社製の SH4 ボードに搭載されている Windows CE.NET 4.2 へは RT ミドルウェア、OpenRTM-aist-0.2.0 の開発段階の Windows 版をベースにし実装していく。この OpenRTM-aist-0.2.0 の構成を Fig.3 に示す。Fig.3 の OpenRTM-aist-0.2.0 は CORBA 実装部分に大部分を CORBA2.6 仕様に追従している OmniORB を使用し動作している。しかし、OmniORB はデスクトップ系の Windows や Linux 系 OS にしか対応しておらず組込系である Windows CE には対応していないという問題がある。そのため、そのまま OpenRTM-aist-0.2.0 を Windows CE へ実装することは不可能である。そこで、本研究では OmniORB の部分を Windows CE に対応した TAO^[3]に変更し、Fig.3 の TAO 版 OpenRTM のように実装を行う。この TAO とは OmniORB や Java などと同様に CORBA の実装の 1 つであり、プラットフォームとして Windows, Linux などの他に組込系 OS の Windows CE が対応している。本研究では Windows CE に対応した TAO を使用し ZMP 社製 SH4 ボードの SDK をプラットフォームとして実装を行う。

現段階までに Windows 版 OpenRTM-aist-0.2.0 の IDL を TAO でコンパイルしてスタブ・スケルトンを作成し TAO への移植を行った。今後、Windows CE への実装し動作確認を行っていく。

4. 研究課題

RTミドルウェアは、まだ開発段階でありデスクトップPCをメインに開発が行われている。従って、デスクトップ系より処理速度が遅く機能が制限されている組込系で動作させたときに、一般的に使用できる動作速度を確保できるかという問題が生じると考えられる。この問題点については、RT ミドルウェアを実装した PAR04R を用いて検証していく。

5. まとめ

本報告では、ロボットシステムの汎用性と拡張性を考慮し、RT ミドルウェアを Windows CE が実装された組込みプロセッサへと適応させる方法について検討を行った。具体的には、ZMP 社製の SH4 ボードを使用し Windows CE へ OpenRTM-aist-0.2.0 の CORBA 部分の OmniORB を TAO へと変更することで RT ミドルウェアを実装する。今後の課題として Windows CE へ RT ミドルウェアを実装したときの動作速度を検証する必要がある。

Table1 SH4 board

CPU		SH4 (SH7760) 200MHz
メモリ	SDRAM	64MB (32MB をユーザプログラムに使用可能)
	ROM (SuperAND)	64MB (32MB をディスクとして使用可能)
各種インターフェイス	シリアル	TTL
	CAN	1Mbps(max) CPU 内蔵 HCAN2, HA13721
	Ethernet	100BASE-T (AX88796L, TG110-S050N2) RJ45 コネクタ外付け
	サウンド	AC97Codec (CPU 内蔵 HAC, STAC9757T) AMP (LM4863) マイク入力, ステレオ 8Ωスピーカ出力, ステレオ, 1W
	CF	2 スロット (3.3V/5V 対応)
無線 LAN (オプション)		IEEE802.11b
		CF 型ワイヤレス LAN カード対応
電源仕様(ボード単体)		DC 5.0V/2.5A 入力 1.5V, 3.3V: ボード内で生成
外形	基板寸法	基板2枚構成 : 組合せ高さ 30mm max A 基板 (CPU, RAM 等): 84.4 × 60.0mm B 基板 (CF, I/O コネクタ等): 84.4 × 70.0mm

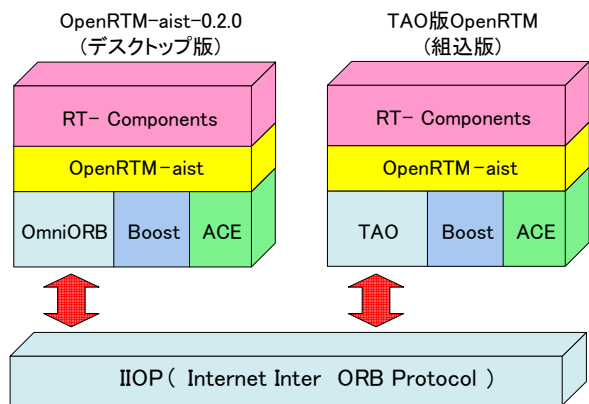


Fig.3 The transplant of OpenRTM-aist-0.2.0

参考文献

[1] OpenRTM-aist-0.2.0 デベロッパーズガイド
独立法人 産業技術総合研究所知能システム部門・タスクインテリジェンス研究グループ

[2] RT ミドルウェアを用いた PAS 分散処理系の考察
土屋 他, SI2005

[3] TAO (The ACE ORB) HP
<http://www.cs.wustl.edu/~schmidt/TAO.html>

[4] RT ミドルウェアプロジェクト HP
<http://www.is.aist.go.jp/rt>

[5] 分散オブジェクト技術のすべて
日本事業出版社・小泉修(著)