高性能ロボット用総合プラットフォーム WR-X の開発

斎藤 卓也^{*1} 村岡 洋一^{*2}

Development of WR-X, an Integrated Platform for Use in High-performance Robots

Takuya Saito^{*1} and Yoichi Muraoka^{*2}

WR-X is an integrated platform for use in high-performance robots, making it possible to build simple, low-cost ultra-high-performance robots that can withstand the rigors of full-scale research use. Using WR-X makes it easy for anybody to build a wide variety of robots simply by combining modules as with a Lego construction set. To keep costs low and realize suitability to a wide range of uses, WR-X's actuator employs a standard over-the-counter RC servo used in radio-controlled models. Using an RC servo for a robot's actuator presents a wide range of difficulties. To overcome these, we developed specialized servo-control hardware to control the RC servo motor directly at a high level. This has made it possible to achieve an actuator with linear torque properties for use in ultra-high-performance robots, using a low-cost RC servo. Furthermore, we also developed an accelerator for conducting hardware-based image-analysis computation, in order to achieve real-time image analysis, which had been difficult with traditional small robots. This system has successfully computed the positions of eight physical objects on a screen in a mere 15.7 [μ s], achieving a speed increase to approximately 60,000 times the speed of software-based computation.

Key Words: Robot Platform, H-Bridge Control, RC Servo, Image Analysis Accelerator, Robot Vision

1. はじめに

大学等でロボット関連の研究を行う場合、まずはその研究対象となるロボット本体を用意する必要がある。研究対象のロボットは形状や構造が多岐にわたり高い性能が要求されるため、既存のロボットを購入すればそれですむとは限らない。そのため目的のロボット本体を独自に製作する必要が多々生じる。ロボットを製作する最も簡単な方法は MIT と LEGO 社により共同開発された LEGO MINDSTORMS NXT [1] 等のロボット・キットを用いることである。しかし、それらは子供向けのロボット教育キットであるため、大学レベルの研究用ロボットを製作するには性能や機能が不足する。

もっと本格的にロボットを構築したい場合、アクチュエータに RC サーボを用い、マイコン・ボードで RC サーボをコントロールするロボットを構築することになる。これは ERATO 北野共生システムプロジェクトで開発されたヒューマノイド・ロボット PINO [2] [3] の製作手法をベースとしたものであり、二足歩行ロボット大会 ROBO-ONE [4] 用ロボット [5] は、ほぼすべてこの手法で作られている。また、KONDO 社の KHR-1 [6]

をはじめとして近年 10~30 万円程度の安価にて市販されているヒューマノイド・ロボットはこれを製品化したものである。

これらのロボットはアクチュエータに RC サーボを流用しているため、ハンチングしたり滑らかに動かなかったりなど、動作に多くの問題を抱えているため、歩行動作等の研究を本格的に行うのは困難である。また、これらロボット用のマイコンの能力では、ラジコンからの指示によりあらかじめ指定しておいたモーションを再生して動作するロボットを構築するのが精一杯であり、それ以上高度なものを要求すると計算能力が不足する。大学の研究レベルではリアルタイム画像解析により周囲の状況を判断しながら動くロボットや、人工知能により自律動作するロボット、または音声認識や音声合成により人間とコミュニケーションを取れるロボットなどが考えられるが、これらの構築は計算能力的にほとんど不可能である。

本格的な研究用ロボットとしては富士通の HOAP3 [7] が市販されている。これは研究用途を想定したヒューマノイド・ロボットであり、大学での研究に適した性能や機能を有したものであるが、価格が約 600 万円と大変高価であることや、与えられたヒューマノイド・ロボット以外の形状の研究には利用できないという欠点がある。このようなことから大学研究レベルに使用できる性能や機能を持ったオリジナル・ロボットを製作するのは非常に難しく、ロボット研究の敷居は想像以上に高い。

そこで、これら問題を一举に解決するため、高性能ロボット用総合プラットフォーム WR-X を開発した。WR-X は、LEGO

原稿受付 2007 年 12 月 28 日

^{*1}早稲田大学大学院理工学研究科情報・ネットワーク専攻

^{*2}早稲田大学理工学術院

^{*1}Graduate School of Science and Engineering, Waseda University

^{*2}Department of Science and Engineering, Waseda University

■ 本論文は有用性で評価されました。

のようにモジュールを組み合わせるだけで、あらゆる種類のロボットを誰でも簡単に構築することを可能にする。しかも、WR-Xにより構築されたロボットはPCに匹敵する本格的な計算能力をもち、HONDA ASIMO [8] [9] や産総研のHRP-2 [10] のような本格的なアクチュエータ性能を有するのである。

WR-XのCPUは432 [MIPS], 1.7 [GFLOPS] という非常に高い演算能力を有するため、従来自作小型ロボットでは難しかったリアルタイム画像解析、人工知能、音声認識などの本格的な研究も可能になる。アクチュエータにはRCサーボを使用することにより、超小型サーボから大型大トルク・サーボまで様々な要求に応えられる柔軟性および低コストを実現した。RCサーボをロボット用アクチュエータとして使用すると動作に様々な問題が生じるが、WR-Xでは直接RCサーボ・モータを高度に制御する専用ハードウェアを開発することにより、この問題を解決した。そのためRCサーボでありながらリニアなトルク特性を有した高度なロボット用アクチュエータを実現した。

小型ロボットに用いられる組込マイコンでは性能的にカメラから画像をキャプチャすることさえ難しく、さらにリアルタイム画像解析を行うなどほとんど不可能に近い。この問題を解決するため、WR-XではCMOSカメラからの画像キャプチャだけではなく画像解析計算までもハードウェアで行う専用の画像解析アクセラレータを開発した。画像解析アクセラレータにより、8個の物体の重心座標を僅か15.7 [μ s] で計算可能にするなど、従来のソフトウェアによる計算と比べて約6万倍の高速化に成功した。これにより、従来小型ロボットでは難しかったリアルタイム画像解析機能をほとんどCPU負荷なしに実現した。

さらに、ジャイロ・センサ、加速度センサなど、様々なセンシングを高度に実現するため、線形位相特性ベッセル型フィルタを用いた多チャンネル高解像度高サンプリング・レートA/Dコンバータを開発した。波形を重視するロボット・センシングを高度にサポートしている。

いくら高性能であってもコストが高すぎでは意味がない。そこでWR-Xは可能な限り低価格にて実現可能できるように設計されている。WR-Xは安価で誰でも入手可能な部品のみを使い設計されており、アクチュエータも市販RCサーボをベースとすることにより低コストを実現している。WR-Xにより富士通HOAP3クラスのロボットを構築した場合、部品代や基板製造代等原価においてHOAP3の1/5~1/10程度の低コストで製作可能である。

この高性能なハードウェアのプログラミング環境として、PCのプログラミング感覚でのロボットプログラミングを実現するWR-X OSという専用OSを開発した。

さらに、このWR-Xは誰でも自由に利用できるようにWR-Xの全回路図、全基板パターン、全Verilog HDLプログラム、WR-X OS等をインターネット [11] 上で公開している。

このように、WR-Xは従来のロボット・キットとは一線を画する本格的な研究用ロボットを簡単かつ低コストにて構築可能にする高性能ロボット用総合プラットフォーム・フォームである。

2. WR-X システム・アーキテクチャ

2.1 機能別モジュール設計

WR-Xは機能単位でモジュール化されており、それらモジュールを自由に組み合わせることにより様々なタイプのロボット構築に対応している。モジュールは理論上最大4,096個まで接続可能であり、各モジュールはWR-X BUSと呼ばれる専用バスにより接続される。WR-Xは以下の五つのモジュールにより構成される。

- (1) Mother module
- (2) PWM Servo module
- (3) RC Servo module
- (4) A/D module
- (5) Vision module

WR-X BUS システム全体の外観を **Fig. 1** に示す。Fig. 1 では五つのモジュールすべてをスタック接続しており、図右下のWR-X BUS コネクタで各モジュールは接続されている。システム全体の構成図を **Fig. 2** に示す。WR-Xの各モジュールはWR-X BUSにより接続される。Mother モジュールがマスターとなり、その他のモジュールを集中制御する。また、各モジュールはRCサーボやCMOSカメラ等各種外部デバイスと接続される。

Mother モジュールはPCのマザーボードに相当するものであり、CPU、RAM、ROMなどコンピュータを構成するために必要なすべてのものが集約されている。Mother 以外のモジュールは周辺装置として動作する。PWM Servo モジュールは市販のほぼすべてのRCサーボを高性能ロボット用アクチュエータへと改造するためのモジュールである。このモジュールにより、市販RCサーボをロボット用アクチュエータとして使用したときに生じる様々な問題を解決し、高性能なロボット用アクチュエータへ

Fig. 1 WR-X system overview

Fig. 2 WR-X system structure

と変えることができる。RC Servo モジュールは市販 RC サーボを改造せずにそのまま利用するとき用いるモジュールである。1 モジュールで 32 個の RC サーボを制御することができる。さらに、KONDO のロボット用サーボ KRS Red Version の ICS 通信機能にも対応しており、キャラクタースティック・チェンジ、パワー・リダクションのコマンド送信およびポジション・キャプチャによる角度受信が行える。また、KONDO KRS HV ハイボルテージ・タイプにも対応しており、ほぼすべての市販ロボット用サーボの機能をフルに引き出して利用することができる。A/D モジュールは 32 [ch] のアナログ信号をデジタルに変換する A/D 変換モジュールである。PWM Servo モジュールと連携して RC サーボの高性能化に利用するほか、ジャイロ・センサ、加速度センサ、圧力センサ等の様々なセンシングを、12 [bit] の高解像度、10 [kS/s] の高サンプリング・レートで実現している。また線形位相特性ベッセル型アンチ・エイリアシング・フィルタにより正確な波形観察を可能にしている。Vision モジュールは画像解析アクセラレータである。このモジュールは、カメラから画像データをキャプチャするだけでなく、キャプチャすると同時に画像解析計算を高速演算するハードウェア・アクセラレータ機能を内蔵している。これにより計算量が莫大なためリアルタイム処理が難しい画像解析処理を、CPU でほとんど計算することなしに実現している。

2.2 WR-X モジュール概要

各モジュールはすべて 10 [cm] × 10 [cm] の大きさに統一されている。各モジュールは WR-X BUS により接続を行い、積み重ね設置することにより省スペース化を実現している。モジュール制御に要する CPU 負荷を最小限に減らすため、各モジュールには CPU へのハードウェア割り込み要求機能が搭載されている。ハードウェア割り込みは 15 段階のプライオリティが設定可能である。割り込み要求信号線を減らすため、15 段階の割り込み要求信号を 4 [bit] 信号にエンコードし、各モジュール間で 4 [bit] 割り込み要求信号をデジタイズ・チェーン接続する仕組みを採用している。ロボットではアクチュエータ動作やカメラの画像取得などにおいて動作タイミングの同期が重要である。そのため、WR-X ではモジュール間同期機能を搭載している。同期端子をモジュール間でデジタイズ・チェーン接続することにより、全モジュールを同期動作させることが可能である。

3. Mother Module

Mother モジュールは WR-X システムの中核を成すものであり、PC におけるマザーボードと同じ役割を果たすものである。この Mother モジュールには CPU、RAM、Flash ROM 等、コンピュータを構成するのに必要なすべてのものが凝縮されている。Mother モジュールの外観を Fig. 3 に示す。

すべての WR-X システムは必ず一つの Mother モジュールを必要とし、このモジュールがすべての他モジュールを集中制御する。小型ロボット用 Mother モジュールには、必要十分な計算能力を有しながら小型軽量かつ低消費電力であることが要求される。一般的な市販ヒューマノイド・ロボットでは CPU に Microchip PIC シリーズや Renesas Technology H8, SuperH/SH2 が用いられている。過去に第 4 回 ROBO-ONE 大

Fig. 3 WR-X mother module

Fig. 4 WATOM

Table 1 Specifications of WR-X mother module

CPU	Renesas Technology SuperH/SH4 SH7750R
CPU Clock Frequency	240 [MHz]
Cache Memory	IC : 8 [KB] / OC : 16 [KB]
SPEC	432 [MIPS] / 1.7 [GFLOPS]
Bus Clock Frequency	60 [MHz]
RAM	1 [MB] Static RAM 32 [bit] access
ROM	512 [KB] Flash ROM 8 [bit] access
Serial Port	2 [ch] CMOS & RS-232C level
PCB	4 [Layer]

会に出場するためにヒューマノイド・ロボット WATOM を製作した。このロボットでは CPU に SuperH/SH2 を使用した。WATOM の外観を Fig. 4 に示す。

このロボットで 6 自由度の Inverse Kinematics 計算により各関節角度をリアルタイムで計算しながら歩行するプログラムを作成して歩行させたところ、搭載された SuperH/SH2 CPU ではその計算に約 16 [ms] もの時間が必要であった。RC サーボの制御周期は 20 [ms] であるため、Inverse Kinematics 計算だけで約 80% の CPU パワーを使用している計算になる。このことから、歩行計算以外にさらに画像解析や音声認識等の複雑な計算を行うには、SuperH/SH2 クラスの CPU では計算能力的に不十分であると考えられる。

この問題を解決するために専用のマイコン・モジュール「WR-X Mother モジュール」を開発した。Mother モジュールの仕様を Table 1 に示す。Mother モジュールでは組み込み用 CPU として国産最速クラスの性能を有し、CPU 本体や開発環境等の入手も容易な Renesas Technology SuperH/SH4 SH7750R 240 [MHz] [12] を採用した。この CPU は 432 [MIPS], 1.7 [GFLOPS] という数年前の PC に匹敵するほどの計算能力を有していながら、最大約 1.9 [W] と非常に低消費電力であるため、冷却用ヒートシンクやファンを必要とせず、小型軽量が要求される小型ロボット用途には最適であると判断した。

主記憶装置には 1 [MB] の容量を持ち、32 [bit], 12 [ns] アクセスの超高速 SRAM と 512 [KB] の Flash ROM を搭載している。Flash ROM には Flash ROM ライタ、ブート・ローダおよびユーザ・ソフトウェアが格納される。実際に実行されるソフトウェアはブート・ローダにより Flash ROM から SRAM へとコピーされ、SRAM 上で実行される。SRAM アクセスはランダムアクセス時 480 [Mbit/s] という高メモリ帯域を実現しているため、ソフトウェアは高速に実行される。また PC 等との

Table 2 Specifications of AKI-SH2/7045F

CPU	Renesas Technology SuperH/SH2 SH7045F
CPU Clock Frequency	28.64 [MHz]
Cache memory	IC : 1 [KB]
RAM	1 [MB] Static RAM 32 [bit] access
ROM	256 [KB] CPU built-in flash ROM

Table 3 Performance comparison between WR-X mother module and AKI-SH2/7045F

	AKI-SH2	WR-X Mother Module	WR-X / AKI-SH2
Loop 200,000 times	92 [ms]	7.5 [ms]	12.3 [times]
Trigonometric computation 120 times	60 [ms]	83 [μ s]	723 [times]
6 DOF $\times$ 2 inverse kinematics computation	16 [ms]	45.4 [μ s]	352 [times]

通信用に 3.3 [V] CMOS および RS-232C 切り替え可能なシリアル・ポートを 2 [ch] 用意している。

3.1 WR-X BUS の構造

他の WR-X モジュールとの接続には WR-X BUS と呼ばれる専用 BUS を用いる。WR-X BUS には WR-X BUS 1 と WR-X BUS 2 の 2 種類があるが、通常は WR-X BUS 1 のみを使用する。WR-X BUS 1 は多数の外部 I/O デバイスとの高速通信を想定し、8 [bit] データバス、12 [bit] アドレスバス、アクセスコントロール信号等により構成されている。WR-X BUS 2 は 32 [bit] データバス、26 [bit] アドレスバス、CS 信号、DRAM 制御信号など、SuperH/SH4 とフルに接続する必要があるときにのみ使用する。

WR-X BUS は WR-X モジュールとのバス接続をすることにより、デジチェーン接続やスター型接続等と比較し、モジュール間接続のケーブル数や重量を軽減している。また、WR-X BUS は CPU のバスと直結しており、WR-X BUS 1 のみで 30 [MB/s]、WR-X BUS 2 との併用で 120 [MB/s] の高速通信を実現している。

3.2 Mother Module の性能評価

WATOM や多くの安価なヒューマノイド・ロボットに用いられている SuperH/SH2 CPU と WR-X Mother モジュールの性能比較実験を行った。性能比較対象の SuperH/SH2 の CPU ボードは秋月電子通商 [13] で販売されているマイコン・ボードで最高速の「AKI-SH2/7045F 超高速 RISC マイコン・ボード」[14] を用いた。AKI-SH2/7045F の仕様を **Table 2** に示す。このマイコン・ボードは、ROBO-ONE 大会で優勝をしているロボットや、市販の ROBO-ONE ベースのヒューマノイド・ロボットで最高性能クラスのものとはほぼ同性能のものである。

計算時間の測定結果を **Table 3** に示す。Table 3 より 20 万回ループの場合 AKI-SH2/7045F と比較し 12.3 倍高速になり、「三角関数計算 120 回」と「6 自由度 Inverse Kinematics 計算 2 足分」では、それぞれ 723 倍、352 倍と大幅に計算速度が向上している。これは、SuperH/SH2 には搭載されていない FPU が SuperH/SH4 には搭載されていることによるものであると

Fig. 5 HEW for GNUSH **Fig. 6** WR-X OS

考えられる。ロボットで良く用いられる Inverse Kinematics 計算等の科学技術計算では浮動小数点演算を多用するため、FPU により高速に浮動小数点演算が行えることは大変有用である。

一方 Table 3 より、AKI-SH2/7045F では 6 自由度の Inverse Kinematics 計算 2 足分で 16 [ms] もの計算時間が必要であり、RC サーボの制御周期は約 20 [ms] であるため、これでは Inverse Kinematics 計算だけで CPU 負荷の 80% も使用してしまうことになる。これでは画像解析計算や音声認識計算などの複雑な計算をする余裕はない。しかし、WR-X Mother モジュールでは同じ計算が 352 倍も高速になり、わずか 45.4 [μ s] で計算されている。これなら同じ 20 [ms] のサーボ制御周期の場合、CPU 負荷はわずか約 0.23% でしかない。WR-X では 1 [ms] のサーボ制御を目標としているが、その場合でさえわずか約 4.5% の CPU 負荷である。残りの 95% 以上の計算能力を画像解析や音声認識など、複雑な計算処理に回すことができる。

3.3 WR-X Mother Module のプログラミング環境

WR-X ではプログラム開発用コンパイラに KPIT GNUSH v0802 を、IDE に Renesas HEW for KPIT GNU Tools with Simulator を用いている。**Fig. 5** に WR-X プログラミング中の HEW の外観を示す。これら WR-X の開発環境はすべてフリーでインターネットから入手および使用できる。これらは Windows XP 上で動作し、Visual Studio とほぼ同じ感覚でプログラミングできる。

一般に組込システムではスタートアップ・ルーチン部分から一からすべてプログラムを組む必要があるが、それにはハードウェアの完全な理解が必要になるため、一般ユーザには敷居が高くなりすぎてしまう。そこで、WR-X では WR-X OS という WR-X 専用の OS を用意している。WR-X OS は **Fig. 6** に示すように、WR-X ハードウェアとユーザプログラムとの間に位置することにより、様々な WR-X モジュールを抽象化し、通常の PC のプログラムのような手軽さで WR-X モジュールを扱うことを可能にしている。

4. PWM Servo Module

PWM Servo モジュールは安価な市販 RC サーボを高性能なロボット用アクチュエータへと変えるためのモジュールである。RC サーボは元来ロボット用アクチュエータとして設計されたものではない。そのためロボット用アクチュエータとして使用したときにはハンチングが生じたり、滑らかに動かなかったりなど様々な問題が生じる。この問題を解決するため PWM Servo Module を開発した。PWM Servo モジュールは RC サーボ内部

Fig. 7 WR-X PWM Servo module

Table 4 Specifications of PWM Servo module

Number of channels	8 [ch]
Maximum PWM frequency	500 [kHz]
H-Bridge control kind	4
H-Bridge control pattern	6
H-Bridge dead band	4 [bit]
PWM Duty resolution	8 [bit]
Maximum pulse current	30 [A]
Maximum constant current	4.2 [A]
Maximum motor voltage	10 [V]

Table 5 Features by control method

	Centralized control	Distributed control
Advanced servo control	○	×
Applicability to various servos	○	×
Synchronized operation	○	×
Communication time cost	○	×
Communication error	○	×
Amount of wiring	×	○
Weight decentralization	×	○

の DC モータを直接高度に制御することにより、安価かつ汎用性を保ちながら、高性能なロボット用アクチュエータを実現している。PWM Servo モジュールの外観を Fig. 7 に示す。PWM Servo モジュールの制御ロジック回路は Xilinx Spartan III 20 万ゲート大容量 FPGA に Verilog HDL にてプログラムされている。PWM Servo モジュールの主な仕様を Table 4 に示す。

4.1 集中制御方式

ロボットのアクチュエータ制御は、アクチュエータごとに制御マイコンを内蔵し、ロボットのメイン・マイコンと体内 LAN により通信して各アクチュエータを制御する分散制御方式と、メイン CPU と直結された高性能なサーボ制御回路が、集中的に各アクチュエータを制御する集中制御方式の 2 種類に分けられる。市販の RC サーボやロボット用サーボは分散制御方式が採用されている。しかし、WR-X PWM Servo モジュールでは集中制御方式を採用している。集中制御方式と分散制御方式は様々な要素においてトレード・オフの関係にある。各要素のトレード・オフ関係について Table 5 に示す。

分散制御方式の利点は配線の数が減ることや、重量が分散することであるが、どちらも致命的なほど大きな欠点ではない。そして、それ以外の点では集中制御方式の方が優れている。

PWM Servo モジュールでは、RC サーボを改造して高性能

(a) WR-X Motor module

(b) RC Servo with WR-X Motor module

Fig. 8 WR-X Motor module

化することが重要であるため、高度なサーボ制御、シンクロ動作、通信時間コスト、通信エラーなど、サーボ制御の根幹にかかわる部分には妥協が許されない。その上で超小型サーボから大型大トルクサーボまで、様々な RC サーボを改造できる汎用性も確保したい。これらの理由により WR-X では集中制御方式を採用した。

4.2 RC サーボの改造

WR-X PWM Servo モジュールでは既存の RC サーボを改造して PWM Servo モジュールと接続することにより高性能化する。ここではその改造方法について述べる。市販 RC サーボ内部には、Fig. 8 (b) の左側のような制御基板が入っている。これを WR-X PWM Servo モジュールと接続するためのインターフェース基板である WR-X Motor モジュールに置き換える。Motor モジュールの外観を Fig. 8 (a) に示す。また、RC サーボの制御基板を Motor モジュールに置き換えた RC サーボの外観を Fig. 8 (b) の右側に示す。このように、基板を置き換えるだけで簡単に市販 RC サーボを WR-X PWM Servo モジュールと接続し、高性能化することができる。また、このインターフェース回路はオペアンプ一つと数個の抵抗およびコンデンサにより構成されている単純な回路であるため、他のあらゆる RC サーボのインターフェース回路基板も簡単に作ることができるように工夫されている。

4.3 H-Bridge

WR-X Servo モジュールでは DC モータを PWM 制御するために Power MOSFET を用いた H-Bridge 回路を使用している。小型で大電流、高速スイッチングを実現するため、小型フラット・パッケージ Power MOSFET として最高性能クラスの International Rectifier 社製の IRF7389 を使用している。IRF7389 は最大ドレイン・ソース間電圧 30 [V]、最大パルス・ドレイン電流 30 [A]、最大コンスタント・ドレイン電流 4.2 [A]、ドレイン・ソース間抵抗 0.058 [Ω] という高性能なスイッチング性能を小型フラット・パッケージにて実現している。

4.4 FET 高速スイッチング技術

高性能な RC サーボにはコアレス DC モータが採用されておりインダクタンスが非常に小さいため、電流連続モードで使用するには数十～100 [kHz] を超えるほどの高い周波数で PWM 駆動する必要がある。また、PWM 周波数が 10 [kHz] 程度まではモータ・コアが物理的に動くため、少なくともそれ以上の PWM 周波数で駆動する必要がある。そこで、WR-X PWM Servo モジュールではそれより遥かに高い 500 [kHz] での正確な PWM

Fig. 9 Switching waveform of Power MOSFET at 500 [kHz]

Fig. 10 H-Bridge circuit using Power MOSFET

Table 6 On/Off mode

FET gate	from 0 to duty	from duty to cycle
1	On	Off
2	Off	Off
3	Off	Off
4	On	Off

Table 7 Low side On/Through mode

FET gate	from 0 to duty	from duty to cycle
1	On	On
2	Off	Off
3	Off	Off
4	On	Off

駆動を目標に FET スイッチング回路の高速化を行った。

まず FPGA の出力電圧 3.3 [V] をモータ駆動電圧 6~10 [V] に上げるため、ダーリントン接続トランジスタによる電圧変換回路を用いている。トランジスタのスイッチング速度を上げるため、スピードアップ・コンデンサ [15] を用いている。電圧変換回路の出力で直接 Power MOSFET のゲートを駆動すると、ゲート容量のため数十 [kHz] 程度までしかスイッチングさせることができない。そこでフラット・パッケージ・トランジスタを用いた PUSH-PULL 回路により、軽量省スペースかつ強力な FET ゲート・ドライバ回路を構成している。

Fig. 9 に、実際に 500 [kHz] で Low Side On/Through モード時のモータ駆動出力波形を示す。500 [kHz] であってもきれいな矩形波で正確にスイッチングできている様子が確認できる。

4.5 H-Bridge 制御パターン

次に H-Bridge 回路の制御方法について述べる。**Fig. 10** に H-Bridge の回路図を示す。通常の RC サーボでは H-Bridge 回路を使用していても、単純な ON/OFF 制御を行っているのみである。しかし実際には H-Bridge の制御方式には On/Off, On/Through, On/Break, LINEAR の 4 種類の動作モードがあり、さらに On/Through モードと On/Break モードでは High-side 制御, Low-side 制御の 2 種類に分けられる。さらに H-Bridge に貫通電流が流れないようにするため、デッドバンドが設定できる必要がある。**Table 6~9** に、それぞれの動作モードにおける動作パターンを示す。ただし、複雑になるためデッドバンドは省略してある。

4.6 高度 H-Bridge 制御によるリニアトルク特性の実現

H-Bridge の制御パターンや PWM 周波数により、サーボモータが発生するトルク特性は変化する。そこで、KO PROPO

Table 8 Low side On/Break mode

FET gate	from 0 to duty	from duty to cycle
1	On	On
2	Off	On
3	Off	Off
4	On	Off

Table 9 Linear mode

FET gate	from 0 to duty	from duty to cycle
1	On	Off
2	Off	On
3	Off	On
4	On	Off

Fig. 11 On/Off mode

Fig. 12 Low side On/Through

Fig. 13 Low side On/Break

Fig. 14 Linear

PDS-2344FET サーボを PWM Servo モジュールで制御し、実際にトルク特性がどのように変化するのかを測定した。

On/Off モード時のトルク測定結果を **Fig. 11** に示す。このグラフによると周波数が高くなるにつれデューティー比に対して次第にリニアな特性からズレが大きくなって行くことが読み取れる。しかし、周波数が 31.25 [kHz] の低い場合には、ほぼリニアな特性が得られているため、On/Off モードで使用する場合は低い PWM 周波数を用いるのがよいと考えられる。

次に、Low Side On/Through モード時の測定結果を **Fig. 12** に示す。こちらも On/Off モード同様にリニア特性からのズレが大きいが、若干改善されている。

Low Side On/Break モード時の測定結果を **Fig. 13** に示すが、このモードではリニアリティが大幅に改善され、ほぼ全 PWM 周波数においてリニアなトルク特性が得られている。これは On の直後に Break をかける（ショートする）ことにより、モータのインダクタンス成分の影響がうまく排除されているためだと考えられる。また、後述の Linear モードと比べると消費電力が少ない点もメリットである。

最後に Linear モードのトルク測定結果を **Fig. 14** に示す。この結果より、Linear モードが全周波数において最もリニアなトルク変動を生じていることが確認できる。しかし、Linear モードは正転・逆転を交互に行うモードであるため、モータでの電

力消費が大きいというデメリットがある。

以上 Fig. 11~14 より, PWM 周波数が低い場合には, On/Off モード, Low side on/off モードおよび Linear モードで, また PWM 周波数が高い場合には Low side on/off モードと Linear モードにて, PWM デューティ比に応じてほぼリニアなトルクが得られることを示した。このように, WR-X PWM Servo モジュールの大規模 FPGA ロジック回路により, 考えうるすべてのパターンの H-Bridge 制御が可能となり, 低価格な RC サーボであってもリニアなトルク特性をもった高度なロボット用アクチュエータを実現している。

4.7 制御パターンとステップ応答

制御パターンとステップ応答との関係を調べたものを Fig. 16 に示す。この実験では P 制御周期を 5 [ms], モータの PWM 周波数を 125 [kHz] に設定した。この実験結果から, On/Break 制御が最も振幅が小さく, また最も短い時間で収束することが分かった。次に On/Off 制御が優れており, On/Through 制御は振幅が最も大きく, 収束するまでの時間も長い。Linear 制御は発振している。このように H-Bridge の制御パターンを変えることにより RC サーボのステップ応答特性が変化し, 一般的な On/Off 制御と比べ On/Break 制御を用いることにより改善することを確認した。

5. RC Servo Module

RC Servo モジュールは RC サーボを改造することなくそのままコントロールするためのモジュールである。このモジュール 1 枚で 32 [ch] の RC サーボをコントロールすることができる。RC Servo モジュールの外観を Fig. 15 に示す。

RC Servo モジュールは PWM 信号により制御されるほぼすべての RC サーボを制御することができる。PWM 信号は 5 [V]~12 [V] に対応しており, KONDO の HV タイプ・サーボも制御することができる。さらに, KONDO の Red Version シリーズ [16] に搭載されているキャラクタースティック・チェンジ機能, パワー・リダクション機能およびポジション・キャプチャ機能にも対応しており, 高性能なロボット用サーボの機能をフルに使用してコントロールすることができる。コントロール回路は Xilinx Spartan Iie 30 万ゲート FPGA 内に Verilog HDL によりプログラムされている。

実際に RC Servo モジュールにより KONDO KRS-2346ICS Red Version ロボット用サーボを制御したときのパルス波形を Fig. 17 に示す。Fig. 17 (a) はニュートラル位置の 1,500 [μ s] サーボホーン角度指定パルスである。Fig. 17 (b) の最初の 150 [μ s] パルスは Red Version サーボへのポジションキャプチャ用コマンドである。その後, RC Servo モジュールは信号線を入力に切り替える。Fig. 17 (b) の二つ目のパルス (約 1,500 [μ s]) は RC サーボから送られてきたポジションキャプチャパルスである。RC Servo モジュールはこの二つ目のパルスの幅をカウンタにより測定する。高精度なパルスの発生やパルス幅測定を行うために 2 [MHz] の高い制御クロック信号を用いており, 時間分解能 500 [ns] を実現している。これにより角度指定パルスは 13 [bit], コマンド指定パルスは 9 [bit], ポジションキャプチャ 13 [bit] という高分解能にて指定および取得可能である。

Fig. 15 WR-X RC Servo module

Fig. 16 Step response

(a) Angle specification pulse

(b) Position capture pulse

Fig. 17 RC Servo pulses

Table 10 Timesharing method

	0~4 [ms]	4~8 [ms]	8~12 [ms]	12~16 [ms]
0~7 [ch]	Transmit	Break	Receive	Break
8~15 [ch]	Receive	Break	Transmit	Break
16~23 [ch]	Break	Transmit	Break	Receive
24~31 [ch]	Break	Receive	Break	Transmit

しかし, このように複雑かつ高精度なロジック回路は大規模な回路になるため, 30 万ゲートの大規模 FPGA を用いても, 32 [ch] という多チャンネルの実現は困難である。そこで, この RC Servo モジュールでは, 32 [ch] という多チャンネル制御を実現するため, 8 [ch] の送信回路と受信回路をそれぞれ独立させ, それぞれを時分割にて用いることにより 32 [ch] 動作を実現している。時分割動作方法を Table 10 に示す。各サーボから見ると Transmit と Receive が 8 [ms] ごとに行われ, 1 周期が 16 [ms] であるため, 各サーボからは一般規格内の動作をしているように見える。このとき分割により内部回路は 8 [ch] の Transmit 回路と 8 [ch] の Receive 回路および時分割回路により構成されているが, それでも合成後の FPGA における Number of occupied Slices は 99%, Number of Slices containing only related logic は 86%と, 30 万ゲート大規模 FPGA でもほぼ限界に近い回路規模に達している。

5.1 市販ロボット用サーボのハンチング現象測定

市販の RC サーボやロボット用サーボは, モータ制御 PWM 周波数が低い等の問題によりハンチング現象を発生しやすい。しかし, このハンチングの様子を正確に捉えたデータは見たことがない。そこで, WR-X RC Servo モジュールで正確な PWM 信号をロボット用サーボに送り, サーボホーンを軽く指で押して軽い負荷をかけてハンチング現象を意図的に引き起こし, その様子を WR-X RC Servo モジュールのキャプチャ機能により取得し, ハンチング角度変化データを取得した。測定対象サーボは KONDO KRS2346ICS ロボット用サーボである。サーボに

Fig. 18 Hunting phenomenon

Fig. 19 WR-X A/D module

Fig. 20 A/D module sampled results

Table 11 Specifications of WR-X A/D module

Number of channels	32 [ch]
Resolution	12 [bit]
Maximum sampling rate	10 [KS/s]
Cut off frequency of anti-aliasing filter	4 [KHz]
Type of anti-aliasing filter	Bessel lowpass filter

与えた PWM 周期は 16 [ms], PWM パルス幅は 1,500 [μ s] である。そのときのサーボホーン角度変化の測定結果を Fig. 18 に示す。Fig. 18 より、ハンチング現象により ± 0.6 [deg] ほど動いているのが確認できる。このように、市販 RC サーボやロボット用サーボには簡単にハンチング現象を起こしてしまう欠点があり、その様子を実際に測定により示した。

6. A/D Module

A/D モジュールはチャンネル数 32 [ch], 解像度 12 [bit], サンプリング・レート 10 [KS/s] の A/D 変換モジュールである。Anti-aliasing filter にはベッセル型フィルタを使用している。A/D 変換により入力される値が RC サーボのポテンシオ・メータ値や加速度センサ、ジャイロ・センサ等の値であるため、波形が重要になるというロボット特有の要求がある。そのため、通常の A/D 変換装置とは違い、線形位相応答特性を有しているベッセル型フィルタを使用している。A/D モジュールの外観を Fig. 19 に、また仕様を Table 11 示す。

A/D モジュールは加速度センサやジャイロ・センサ等からのデータ入力や、前述の PWM Servo モジュールと連携して RC サーボのポテンシオ・メータ値を高精度に取得するためにも用いられる。A/D 変換 IC には 8 [ch], 12 [bit] A/D コンバータである Microchip MCP3208 を 4 個使用することにより、小型でありながら 32 [ch] の入力を実現している。MCP3208 のコントロール回路は Xilinx Spartan III 20 万ゲート FPGA 内にプログラムされている。FPGA 内のロジック回路は Verilog HDL により記述している。これにより小型軽量多チャンネル高解像度高サンプリング・レートを実現している。

FPGA 内のコントロール・ロジックは 4 個の MCP3208 を同時に制御し、それぞれの ADC IC で 8 [ch] を連続して A/D 変換する。各 ADC IC につき 1 [ch] あたり約 10 [μ s] で A/D 変換を行うため、約 80 [μ s] で全 32 [ch] の A/D 変換は終了する。32 [ch] の A/D 変換が終わると、すべての A/D 変換済みデータを FPGA 内部のレジスタに保存し、CPU にハードウェア

ア割り込み要求を出す。最初の A/D 変換から 100 [μ s] 経過すると、自動的に次の A/D 変換を開始する。これにより CPU は A/D 変換を待つ必要はなく、A/D module からの 100 [μ s] ごとのハードウェア割り込み要求が来たときのみ FPGA 内の A/D 変換データを読み出すだけでよい。この仕組みにより CPU 負荷を極力低減し 32 [ch], 12 [bit], 10 [KS/s] という多チャンネル高解像度高サンプリング・レートを実現している。

6.1 サンプリング能力評価

WR-X A/D モジュールで実際にアナログデータのサンプリングを行った。アナログ信号は LG FG-7005G Function generator により発生した。アナログ信号は 300 [Hz] の sin 波、三角波および矩形波を用いた。A/D モジュールによるサンプリング結果を Fig. 20 に示す。

Fig. 20 の信号周波数はすべて 300 [Hz] であり、ロボットのセンシング用信号としては十分に高い周波数の信号である。また、ロボットではサーボ角度センサ、加速度センサおよびジャイロセンサ等の値をサンプリングすることが想定されるが、そのような場合は通常の A/D 変換装置とは違い、得られた波形が正しい形であるかどうか重要となる。通常の A/D 変換装置ではエイリアシングを嫌うため、Anti-aliasing filter に周波数特性の優れたものを使うが、そうすると位相特性が非線形なため、波形が崩れてしまう。WR-X A/D では波形の歪みを極力抑えるため、ベッセル型フィルタを採用している。そのため三角波や矩形波など高調波を多量に含んでいるものであっても、オーバーシュートやリングングのない正確な波形がサンプリングされている様子が Fig. 20 から確認できる。

7. Vision Module

Vision module [17] は画像解析アクセラレータを内蔵したカメラ画像キャプチャ・モジュールである。CMOS カメラから VGA 解像度 (640 $\times$ 480 [pixel]), 24 [bit] フルカラー RGB データをキャプチャすることができるだけでなく、ハードウェア画像解析アクセラレータ機能により、キャプチャと同時に画像解析計算をハードウェアにて行う機能をもっている。

画像解析計算は非常に多くの計算量を必要とするため、リアルタイムで画像解析計算を行いながらロボットを動かすのは、組み込み用 CPU 最速クラスの SuperH/SH4 240 [MHz] を用いても容易ではない。そこで、WR-X Vision モジュールでは CMOS カメラから画像データをキャプチャすると同時に、ハードウェアにより高速に画像解析計算を行うことを考えた。

Fig. 21 WR-X Vision module

Fig. 22 Structure of WR-X Vision module

高速キャプチャ動作や画像解析計算を行うハードウェアは複雑で大規模なロジック回路になるため、ロジック回路の大部分を Xilinx Spartan IIe 30 万ゲート大規模 FPGA に Verilog HDL により記述し、小型高集積化を実現している。WR-X Vision モジュールの外観を Fig. 21 に示す。Vision モジュールにより実現される機能は以下の三つである。

- CMOS カメラから VGA 解像度、24 [bit] フルカラー画像データをキャプチャする
- 8 色のバイナリ画像データをハードウェアで計算する
- 8 色の物体の重心座標計算の大部分をキャプチャと同時にハードウェアで計算する

7.1 Vision Module の構造

Vision モジュールの構造を Fig. 22 に示す。カメラには Omni Vision 社の OV7620 CMOS カメラを用いている。CMOS カメラからキャプチャした画像データを蓄えるために 512 [KB] の VRAM を内蔵している。VRAM は高速アクセスを実現するため、データ・バス幅 16 [bit]、アクセス・サイクル 12 [ns] の高速 SRAM を使用している。

Fig. 23 に FPGA 内部ロジック回路の構造を示す。FPGA 内部回路は「Image Capture & Analysis 回路」、「I²C Communication Interface 回路」および「Access Controller 回路」の三つの回路で構成されている。「Image Capture & Analysis 回路」は CMOS カメラから送られてきた画像データをキャプチャする回路と画像解析計算を行う回路である。OV7620 は内部コントロールに I²C を用いるために「I²C Communication Interface 回路」を用意している。

7.2 イメージ・キャプチャおよび解析動作

Fig. 24 にカメラから画像データをキャプチャするときの「Image Capture & Analyze 回路」の内部回路構成を示す。Fig. 24 左側の「Image Capture 回路」はカメラから画像データをキャプチャする回路である。Image Capture 回路は CMOS カメラから送られた画像データを順次 VRAM へと書き込んで行く。その結果 1/15 [s] 後には VGA 解像度の 24 [bit] フルカラー画像データが VRAM へとキャプチャされる。

Fig. 24 右側の回路は画像解析回路であり、左側の「Image Capture」回路と同時に並列動作する。また、画像解析回路は画像データを「RGB to YCrCb Converter」、「CDT Comparator」および「BC Adder」の 3 ステージにて処理する。最初に、CMOS カメラから送られてきた RGB データは「RGB to YCrCb

Fig. 23 Inside of FPGA

Fig. 24 Capture operation

Converter」回路により YCrCb データへと変換される。RGB から YCrCb データへの変換公式を式 (1), (2), (3) に示す。

$$Y = 0.59G + 0.31R + 0.11B \quad (1)$$

$$Cr = 0.713 \times (R - Y) \quad (2)$$

$$Cb = 0.564 \times (B - Y) \quad (3)$$

次に、変換された YCrCb データは CDT Comparator 回路により CDT (Color Detection Table) レジスタと比較される。CDT レジスタは CrMin, CrMax, CbMin および CbMax により色範囲を指定している。CDT Comparator 回路の内部構造および比較計算式を Fig. 25 に示す。

CDT レジスタおよび比較回路は Y の値により 4 段階独立して用意されている。四つの比較回路の出力は OR 演算され、どれか一つでも結果が true であれば true、そうでなければ false が出力される。Vision モジュールでは 8 色の色認識を可能にするために、この CDT Comparator は八つ独立に用意されている。

CDT Comparator の出力は BC Adder (Barycentric Coordinates Adder) 回路へと入力される。BC Adder 内には TVoX (Total Value of X), TVoY (Total Value of Y), TNoP (Total Number of Pixels) の三つのレジスタが用意されている。BC Adder は CDT Comparator の結果が true のときに、その時点での X 座標値、Y 座標値をそれぞれ TVoX, TVoY レジスタに加算し、TNoP レジスタの値を +1 加算する。BC Adder 回路は八つ独立して用意されており、それぞれ八つの CDT Comparator と接続されている。

この結果、画像キャプチャが終了すると、TVoX レジスタには指定範囲内の色であったピクセルの X 座標の合計値が、TVoY レジスタには Y 座標の合計値が、そして TNoP にはその総ピクセル数書き込まれる。CPU はこの三つのレジスタの値を読み込み、式 (4), (5) の 2 回の割り算計算をするだけで、指定された色の物体の重心座標 (GX, GY) を求めることができる。

$$GX = TVoX / TNoP \quad (4)$$

$$GY = TVoY / TNoP \quad (5)$$

7.3 イメージ・データ読み込み動作

Fig. 26 に読み出し動作時の Image Capture & Analysis 回路の内部構成を示す。Fig. 26 内のそれぞれの回路モジュールは Fig. 24 のものと同じであるが、Data Read Controller により

Fig. 25 Structure of CDT comparator

Fig. 26 Read operation

Fig. 27 Experimental environment

Fig. 28 Experimental software for Vision

各回路の接続方法が変更されている。Data Read Controller は CPU から指定された画像データ、バイナリ画像データおよび BC Adder へと自由にアクセスできるようになっている。8 色のバイナリ画像データは VRAM から取得した RGB データをそのまま画像解析回路へ送り、その結果を CPU へと送ることにより、バイナリ画像データを別に用意しておく必要性を省いている。また、VRAM から取得する画像データの解像度は、必要に応じて 320×240 [pixel], 160×120 [pixel] および 80×60 [pixel] の 3 段階に設定することができる。 320×240 [pixel] 時には、色数 24 [bit] のほか、対角線上に 2 個の緑色ピクセルを配置した 32 [bit] データで取得できる。32 [bit] データを用いることにより、 640×480 [pixel]、解像度 24 [bit] フルカラー画像を得ることができる。

7.4 WR-X Vision Module の性能評価実験

WR-X Vision モジュールの性能評価を行うための実験を行った。実験環境を Fig. 27 に示す。WR-X Mother モジュール用の実験用ソフトウェアは C 言語を用いてプログラムした。C 言語のコンパイラには KPIT GNUSH v6.02 を用いた。計算時間は WR-X Mother モジュールの I/O ポートに計算中に HIGH パルス信号を出力させ、そのパルス幅を Tektronix TDS2022B オシロスコープで観察することにより測定した。また、Vision Module へコマンドを送信したり、キャプチャした画像データを表示したりするために、Fig. 28 に示す PC ソフトウェアを開発した。

7.5 Vision によるイメージ・キャプチャおよびバイナリ画像データ計算

Fig. 29 (a) の画像は WR-X Vision モジュールによりキ

Fig. 29 Captured image & analyzed data by hardware

Table 12 Time to read image data from Vision module

Reading pattern	Reading time [ms]
All	124.0
RGB	47.20
RGB & 8 binaries	62.40
YCrCb	61.20
YCrCb & 8 binaries	78.00
8 binaries	32.00

ャプチャされた画像データである。画像取得フォーマットは 320×240 [pixel]、24 [bit] フルカラー RGB データである。Vision モジュールは 6 パターンのデータ読み出しに対応している。各読み出しパターンにおけるデータ取得時間の測定結果を Table 12 に示す。Fig. 29 (a) の画像データから緑色、オレンジ色、黄色の各成分を抜き出したバイナリ画像データを WR-X Vision のハードウェアにより計算した結果を Fig. 29 (b)~(e) に示す。Fig. 29 (b)~(d) はそれぞれ緑色、オレンジ色、黄色の各色バイナリ画像データであり、Fig. 29 (e) は 3 色バイナリ画像データをまとめて表示したものである。このバイナリ画像データは Vision モジュールのハードウェア・アクセラレータが計算したものである。8 色分のバイナリ画像データを Vision モジュールから読み出す時間は Table 12 より 32 [ms] であった。

7.6 Vision のハードウェア・アクセラレータ機能による重心座標計算

次に Fig. 29 (b)~(d) のバイナリ画像データから緑色、オレンジ色、黄色の三つの物体の重心座標を計算した結果を Fig. 29 (f)

Table 13 Computational time for 8 binary image data by software

Contents of computation	Computational time [ms]
Getting RGB data	47.2
Conversion from RGB to YCrCb	684.0
CDT Comparison	162.0
Total	893.2

Table 14 Total computational time for 8 barycentric coordinates

Contents of computation	Computational time [ms]
8 binary images	893.2
8 barycentric coordinates	61.6
Total	954.8

に示す。重心座標を計算するには、バイナリ画像データの値が1であるすべての (X, Y) 座標の合計値およびその点の数を計算する必要がある。WR-X Vision モジュールではこの三つの値の計算を画像キャプチャと同時にハードウェアにより計算を行う。そのため、CPU は Vision モジュールから三つのレジスタ値を読み出し、式 (4)、(5) の計算を実行するだけで各物体の重心座標を求めることができる。WR-X Mother モジュールにより 8 色分の物体の重心座標計算に要した時間は 15.7 [μ s] であった。

7.7 ソフトウェアによる重心座標計算

WR-X Mother モジュールの CPU でソフトウェアにより画像データから 8 色分のバイナリ画像データを求める計算を行い、その計算時間を測定した。使用した画像データは VGA 解像度 24 [bit] RGB フルカラー画像である。バイナリ画像データを求めるには、以下の計算を順次行う必要がある。

- (1) RGB データを YCrCb データに変換する
- (2) YCrCb データの値が指定された色範囲に入っているかどうかを計算してバイナリ画像データを求める

これを WR-X Mother モジュールによりソフトウェアで計算したときの計算時間測定結果を **Table 13** に示す。Table 13 より、8 色分のバイナリ画像データをソフトウェアで計算するのに必要な計算時間は 893.2 [ms] であった。

次に、求めたバイナリ画像データから 8 色分の重心座標の計算を行い、その計算時間を測定した。この計算を行うには次の計算を順次行う必要がある

- (1) バイナリ画像データから $TVoX, TVoY$ および $TNoP$ を計算する
- (2) $(GX, GY) = (TVoX, TVoY) / TNoP$ を計算して重心座標を求める

この計算を WR-X Mother モジュールによりソフトウェアで計算したところ、8 色分のバイナリ画像データから重心座標を計算するのに必要な計算時間は 61.6 [ms] であった。

Table 13 およびこの結果から、ソフトウェアにより RGB データから 8 色分の重心座標を計算したときの計算時間測定結果を **Table 14** に示す。Table 14 より、ソフトウェアにより RGB データをもとに 8 色分の重心座標を計算するのに必要な総計算時間は 954.8 [ms] であった。

Table 15 Comparison with software and WR-X vision at computational time

Contents of computation	Software [ms]	Vision module [ms]	Speed-up ratio [times]
Conversion from RGB to YCrCb	731.2	61.2 [ms]	119
8 binary images	893.2	32 [ms]	27.9
8 barycentric coordinates	954.8	15.7 [μ s]	60,815

Fig. 30 KHR-2 HV with WR-X front view **Fig. 31** KHR-2 HV with WR-X rear view

7.8 Vision Module の性能評価

実験結果をまとめたものを **Table 15** に示す。Table 15 より、RGB データから YCrCb データへの変換は、WR-X Vision アクセラレータ・モジュールを用いることによりソフトウェアで計算するよりも約 119 倍高速になり、次に、八つのバイナリ画像データ計算は 27.9 倍高速になった。そして、最終的に求めたい八つの重心座標の計算は約 6 万倍も高速化することに成功した。この計算時間は、ロボットの動作制御周期を 1 [ms] とかなり高速に設定した場合であってもわずか 1.6% でしかない。つまり、Vision モジュールのアクセラレータ機能を使うことにより、わずか 1.6% の CPU 負荷で 8 個の物体の画像認識計算が行えるようになり、残り 98% 以上の CPU 能力をその他複雑な計算に使うことを可能にした。

8. 応 用 実 例

実際に WR-X システムを実ロボットに適用した例を **Fig. 30**, **Fig. 31** に示す。ここではベース・ロボットに KONDO KHR-2HV を用いている。WR-X モジュールとして Mother Module, RC Servo Module および Vision Module を用いている。このように WR-X と市販ロボットを組み合わせることにより、リアルタイムで画像解析も行える本格的なロボットを簡単に作る事が可能である。

WR-X の各モジュールを組み合わせることによるパフォーマンスについてだが、PWM 信号発生や A/D 変換、そして画像解析計算まですべてを WR-X ではハードウェアにより行うため、各モジュールとの通信にかかる時間は 1 アクセスあたり 150 [ns] 程度である。そのため、複数モジュールを接続することによるパフォーマンス低下はロボットの一般的な制御周期であ

る 1~20 [ms] と比較して無視できるレベルであると考えられる。

9. お わ り に

研究用ロボットは様々な形態のものが想定され、さらに高い計算能力や高度なアクチュエータ能力を要求される。このような研究用ロボットを毎回ゼロから作り上げていくのは大変効率が悪い。そこで、このような要望に応えられる研究用ロボット・プラットフォーム WR-X を開発した。

WR-X は機能単位でモジュール化することにより、様々な形態のロボット構築を可能とした。

RC サーボを改造することにより、本格的な研究ロボット用アクチュエータとして必要十分な性能・機能を実現する WR-X PWM Servo を提案した。超小型サーボから大型大トルクサーボまで、ほとんどすべての RC サーボを本格的な研究ロボット用サーボとして用いることができるようになるため、様々な研究用ロボットを簡単に、そして低コストで作ることを可能にした。

KONDO KHR Red Version や HV シリーズ等のロボット用サーボを利用する場合でも、ICS 通信機能によるキャラクタースティック・チェンジやポジション・キャプチャなどのフィードバック機能をフルに利用できる WR-X RC Servo モジュールも開発した。PWM Servo モジュールほどの性能が必要に用途においても、既存サーボの性能をフルに発揮して使用することを可能にした。

WR-X システムは高性能なロボットを構築するための総合プラットフォームとして設計されている。そのため、従来のロボット用組込 CPU では実現が難しかったリアルタイムでの画像解析処理や音声認識処理などの複雑な演算を実現するため、WR-X Mother Module と呼ばれる超高性能な組込マザーボードも開発した。

本格的な研究用途を考慮し、高度なセンシングを実現するため、32 [ch], 12 [bit], 10 [kS/s] という多チャンネル高分解能高サンプリング・レートを実現し、かつ線形位相特性ベッセル式アンチ・エイリアシング・フィルタを搭載したロボット向け A/D Module も用意した。これにより小型ロボットでありながら正確な波形観察が行えるなど、本格的な研究用ロボット向けリアルタイム A/D コンバータを実現した。

さらに、WR-X Vision モジュールというロボット用画像解析ハードウェア・アクセラレータも提案した。画像解析計算は莫大な計算量を必要とするため、従来の組み込み用 CPU ではリアルタイムでの画像解析は大変困難であった。そこで、WR-X Vision モジュールではロボットの画像解析処理で良く用いられるバイナリ画像データ計算や重心画像計算を専用のハードウェア

で計算することにより、ほとんど CPU で計算することなしにリアルタイム画像解析を可能にした。このハードウェア画像解析アクセラレータにより、8 個の物体の重心座標をわずか 15.7 [μ s] で計算可能にするなど、従来のソフトウェアによる計算と比較して約 6 万倍もの高速化に成功した。

参 考 文 献

- [1] The LEGO Group: LEGO.com MINDSTORMS NXT Home, <http://mindstorms.lego.com/>, Accessed 2007 Dec 11.
- [2] F. Yamasaki, T. Miyashita, T. Matsui and H. Kitano: "PINO the Humanoid: A basic architecture," Proc. of The Fourth International Workshop on RoboCup, 2000.
- [3] F. Yamasaki, T. Miyashita, T. Matsui and H. Kitano: "PINO the Humanoid that walk," Proc. of The First IEEE-RAS International conference on Humanoid Robots, 2000.
- [4] ROBO-ONE OFFICIAL SITE, <http://www.robo-one.com/>, Accessed 2007 Dec 13.
- [5] 梓: "ROBO-ONE から生まれた 2 足歩行ロボットキット", 日本ロボット学会誌, vol.24, no.1, pp.36-40, 2006.
- [6] J.H. KIM: "Development of a humanoid biped walking robot platform KHR-1-initial design and its performance evaluation," Proc. of IAPR Int. Workshop on Humanoid and Human Friendly Robotics, 2002.
- [7] 木村, 村瀬: "[HOAP] シリーズの事業化", 日本ロボット学会誌, vol.22, no.1, pp.10-12, 2004.
- [8] K. Hirai, M. Hirose, Y. Haikawa and T. Takenaka: "The development of the Honda humanoid robot," IEEE International Conference on Robotics and Automation (ICRA '98), pp.1321-1326, 1998.
- [9] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki and K. Fujimura: "The intelligent ASIMO; System overview and integration," IEEE/RSJ Int. Conf. On Intelligent Robots and Systems (IROS '02), pp.2478-2483, 2002.
- [10] 五十棲, 赤地, 平田, 金子, 梶田, 比留川: "ヒューマノイドロボット HRP-2 の開発", 日本ロボット学会誌, vol.22, no.8, pp.1004-1012, 2004.
- [11] Takuya Saito: WR-X Homepage, <http://www.tutinoko.org/wrx/>, Accessed 2008 Oct 16.
- [12] F. Arakawa: "SH4 RISC multimedia microprocessor," IEEE Micro Journal Article, pp.26-34, 1998.
- [13] 秋月電子通商, <http://akizukidenshi.com/>, Accessed 2007 Dec 12.
- [14] AKI-SH2/7045F 超高速 RISC マイコンボード, <http://akizukidenshi.com/catalog/items2.php?q=%22K-00206%22&s=popularity&p=1&r=1&page=>, Accessed 2007 Dec 12.
- [15] 浅井紳哉: "パワー・デバイスのスイッチング駆動テクニック", トランジスタ技術, vol.41, no.3, pp.153-162, 2004.
- [16] KONDO KAGAKU CO., LTD, http://www.kondo-robot.com/html/Product_main.html, Accessed 2007 Dec 11.
- [17] T. Saito and Y. Muraoka: "Development of WR-X Vision, Image Analysis Accelerator for Robots," Proc. of The 2007 International Conference on Embedded Systems & Applications (WORLDCOMP'07), pp.148-154, 2007.

斎藤卓也 (Takuya Saito)

2003 年早稲田大学理工学部情報学科卒業。2005 年早稲田大学大学院理工学研究科情報・ネットワーク専攻修士課程修了。現在早稲田大学大学院理工学研究科情報・ネットワーク専攻博士課程在籍中。コンピュータ・サイエンス分野におけるロボティクスの研究や音楽情報学の研究に従事。情報処理学会、電子情報通信学会の各会員。(日本ロボット学会学生会員)

村岡洋一 (Yoichi Muraoka)

1965 年早稲田大学理工学部電気通信学科卒業。1969 年アメリカイリノイ大学電子計算機学科修士課程修了。1971 年同大学博士課程修了 Ph.D.. 1985 年早稲田大学理工学部教授。2000 年教務部情報化推進担当部長を経て 2002 年 11 月より早稲田大学理工学術院教授。並列分散処理等の研究に従事。情報処理学会副会長。ソフトウェア科学会理事。人工知能学会理事など多数歴任。