

ロボットプラットフォーム “オープンロボットコントローラアーキテクチャ”

Open Robot Controller Architecture (ORCA), the Robot Platform

尾崎 文夫 橋本 英昭

■ OZAKI Fumio

■ HASHIMOTO Hideaki

家庭内の環境に応じて動くロボットを作り出すためには、運動制御に関するだけでも非常に広範な技術を必要とする。この広範な技術をコンパクトにかつ多様に組み合わせてロボットに実装するには、個々の技術をコンポーネントとして組み合わせる枠組みが必要である。

そのような枠組みの一つとして、東芝ではロボットの運動制御技術を階層化したロボット技術 (RT) 参照モデルを提案し、これに基づいて“オープンロボットコントローラアーキテクチャ (ORCA)”を提唱している。ORCAでは、オブジェクト指向技術を利用してコンポーネント化を図るとともに、分散オブジェクト技術を利用してコンポーネントをネットワークのどこからでも利用できるようにした。また、ORCA対応のボードの供給により、ソフトウェアだけでなくハードウェアにおいてもロボット技術のコンポーネント化を図っている。

Toshiba has proposed the open robot controller architecture (ORCA) based on the robot technology (RT) reference model proposed by Toshiba, so as to allow RT components to be easily packaged. Many technologies are necessary to develop a home robot that acts in response to its environment. A framework is required that can integrate reusable components for which various companies and individuals contribute their technologies. ORCA is such a framework. ORCA uses distributed object technology to enable such components to be used transparently anywhere via the network. We have developed motion control boards that are compatible with the ORCA architecture. ORCA supports the packaging of robot technologies into components, not only as software but also as hardware.

1 まえがき

東芝では、ロボット技術 (以降、RTと略記) の研究開発を行っている。ロボットは、長い間人々の期待を集めてきている。ロボットが世の中でこのように期待されているにもかかわらず、いまだに一般家庭で利用できるような状況にないのはなぜだろうか。ロボットは、工場や宇宙環境、原子力発電所の保全作業など事前にロボットのために環境を整えることができる場所では活躍しているが、ロボットのために環境を整えられない一般家庭やオフィスなどではいまだに普及していないのである。それはロボットが自分の周りの環境を認識することが非常に難しく、環境に応じて動作することができないためである。

環境に応じて動くロボットを作り出すためには、非常に広範な技術を必要とする。例えば、“ドアが開いていたか確認する”ことを考えると、まずドアがどこにあるかを理解し、そこへ行くためにはどういう経路で行けばよいかを計画する。そして家の中のテーブルやいすなどの事物を認識し、避けながら目的のドアへの移動を行う必要がある。認識から制御まで、更に機構部の製作まで含めると、これを1社でやり遂げることは難しい。たくさんの専門家がそれぞれの技術を

ロボットを作り上げるためのコンポーネントとして製作し、このコンポーネントを誰もが使えるようなオープンな枠組みを作ることで、広範な技術を簡単に利用できるようにすることが解決の手段となると考える。このような枠組みがあれば、ロボット製作に必要な広範な技術を1社あるいは1者で持たないでも、他の人が作ったコンポーネントを利用することでロボットを製作することができる。このようにいろいろなロボットを作ることができるになれば、ロボット市場が活性化し、コストダウンも図れる。

このようなコンポーネント化を促進するため、当社ではRTの階層化を考えたRT参照モデルを提案し、これを基にオープンロボットコントローラアーキテクチャ (ORCA) ^{(1), (2)} の研究開発を行っている。ロボット技術のオープン化については、現在、国のレベルでも新エネルギー・産業技術総合開発機構 (NEDO) のRTミドルウェアプロジェクト ⁽³⁾ や (社) 日本ロボット工業会のORiN (Open Robot interface for the Network) ⁽⁴⁾ などのオープン化プロジェクトがあり、オープン化への期待が高まっている。

ここでは、第2章でORCAの概要、RT参照モデル、ORCAアーキテクチャを紹介し、第3章でORCA対応ハードウェアである運動制御ボードについて述べる。

2 オープンロボットコントローラーアーキテクチャ (ORCA)

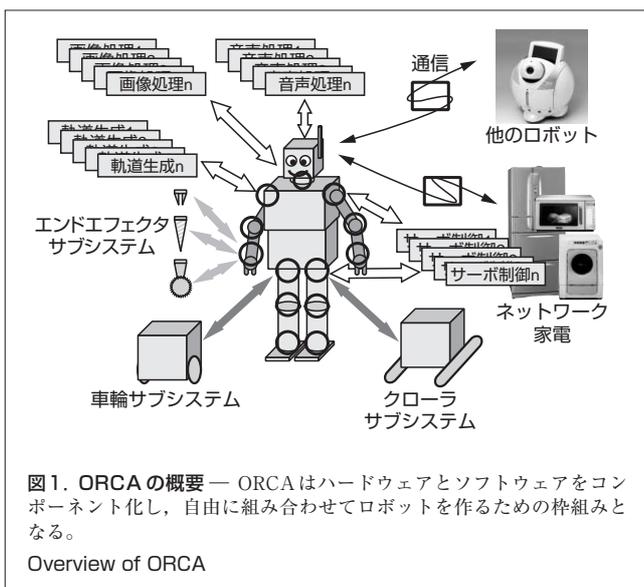
2.1 ORCAの概要

ORCAでは部品化を実現するために表1に示すようなロボット技術を階層化したRT参照モデルに沿って運動制御のソフトウェアを構成する。RT参照モデルで構成される1単位をサブシステムと呼んでいる。例えば1本のアームや移動台車のようなものがサブシステムである。これらを組み合わせると、更に大きなサブシステムを構成することも可能である。

表1. RT参照モデル
RT reference model

階層	名称	概要
第6層	タスク制御層 (TC層)	ロボットに行わせる作業を指定し、この作業の実行をMC層のロボットの動きへの変換を規定
第5層	モーション制御層 (MC層)	作業座標系(ワールド、ベース、手先、ワークなど)でのロボットの動きを軸での動きへの変換を規定
第4層	軸制御層 (JC層)	軸(関節角)での動きをアクチュエータの動きへの変換を規定
第3層	アクチュエータ制御層 (AC層)	ロボットを動かす各アクチュエータの動きに対する物理量への変換を規定
第2層	I/Oリンク層 (IO層)	ロボットのハードウェアへの入出力とロボット動作の物理量(トルクや角度など)の変換を規定
第1層	物理層 (PH層)	ロボットとコントローラをつなぐI/O機器を規定

外部との通信に関しては、既に存在する規格や標準を採用できるところは採用し、汎用性のあるアーキテクチャを目指している。例えば、ユニバーサルプラグ&プレイ (UPnP™^(注1):ネットワークに接続した機器を他の機器から自動的に認識する技術)にも対応できるようになっており、



UPnP™機器との連携を図ることができる。

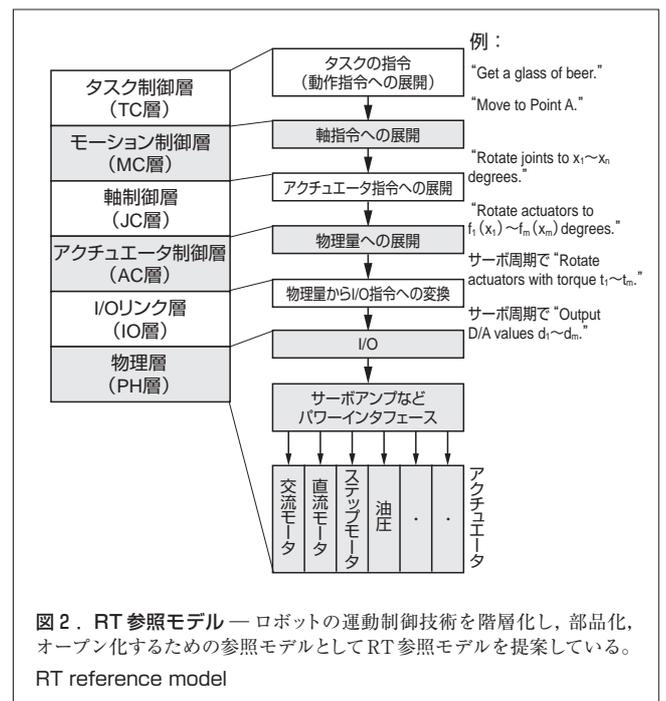
図1に示すようにロボットと他のロボットやネットワーク家電との通信(図中、□で示す)についてはUPnP™的に自動的に相互認識し、お互いを利用できる。ロボット内のサブシステムである脚と腕、ハンドなどとの通信(図中、○で示す)についても同様にプラグ&プレイ(PnP:ある機器を他の機器に接続するときに自動的に認識する技術)的につなげば利用できることを目標とする。図に示すようなエンドエフェクタサブシステム、車輪サブシステム、クローラサブシステムなどが簡易に交換できるようにする。更に画像処理、音声処理、軌道生成、サーボ制御などのソフトウェアを差し替え可能なように構成し、いろいろなメーカーがロボット用ソフトウェアの製作を行えることも目指している。

2.2 RT参照モデル

ロボット技術を階層構成で考えると、部品化を進めオープン化をするのに便利である。階層内で上下の層間のインタフェースを決めることで縦のオープン化(同じシステム内の別層間の通信)ができ、更にそれぞれの階層でのソフトウェア・ハードウェア部品を入れ替え可能とすることで横のオープン化(異種のロボット部品の利用)ができるようになる。

表1及び図2はこのようなロボット用ソフトウェア階層を表したものでRT参照モデルと呼ぶことにする。下位層から物理、I/O(Input/Output)リンク、アクチュエータ制御、軸制御、モーション制御、タスク制御の6層で構成する。

第1層に物理(PH)層としてI/O用のハードウェアからパワーインタフェースを介してアクチュエータまでを定義する。すなわちロボットのメカニズムとコントローラをつなぐI/O



(注1) UPnPは、UPnP Implementers Corporationの商標又は証明マーク。

機器への入出力を規定する。アクチュエータに応じて位置、速度、電流などを対応するI/Oに対して指令する層である(例えばD/A(Digital to Analog)変換器に速度指令をデジタル出力するなど)。また、アクチュエータの位置情報などをI/O機器を介して上位層であるI/Oリンク層へ入力する。

第2層はI/Oリンク(IO)層と呼び、ロボットのメカニズムやコントローラのハードウェアの入出力とロボット動作の物理量(トルクや角度など)の変換を規定する。下位層のPH層でセンシングしたデータを物理量に変換し、上位層のアクチュエータ制御層で計算した物理量をI/O機器に出力するデータとして変換する。

第3層はアクチュエータ制御(AC)層と呼び、ロボットを動かす各アクチュエータの動きを実現する物理量に変換する。下位層のIO層から得たフィードバックデータと上位層の軸制御層から渡された指令値を用いてアクチュエータの制御を行う。

第4層は軸制御(JC)層と呼び、関節角座標系でのロボットの動きをアクチュエータの動きに変換する。関節角座標でのフィードバックをする場合にはAC層からのデータを利用して行う。

第5層はモーション制御(MC)層と呼び、作業座標系(ワールド、ベース、手先、ワークなど)でのロボットの動きを軸(関節)の動きに変換する。作業座標でのフィードバックをする場合にはJC層からのデータを利用して行う。

第6層はタスク制御(TC)層と呼び、ロボットに行わせる作業を指定し、この作業をMC層でのロボットの動きの列に変換する。タスクレベルのロボット言語⁽⁵⁾による記述、あるいはユーザー自身のMC層とJC層のロボットの動きを記述したロボット言語列などがこの層に位置する。

図2の右側に例として“Get a glass of beer.”という命令に対する各層レベルのコマンドを示した。“Get a glass of beer.”というTC層のコマンドはロボットアームに対して“Move to Point A.”のような手先移動命令列に展開される。この命令を受けたMC層では各軸の動きに展開して“Rotate joints to $x_1 \sim x_n$ degrees.”(nは自由度数)のような命令列を生成する。JC層ではこれをアクチュエータ命令列“Rotate actuators to $f_1(x_1) \sim f_m(x_m)$ degrees.”(mはアクチュエータ数, fは軸移動量をアクチュエータ移動量に変換する関数)に展開する。AC層はサーボ周期で“Rotate actuators with torque $t_1 \sim t_m$.”(トルク指令ドライバの場合)に展開する。これをIO層で“Output D/A values $d_1 \sim d_m$.”のようにD/A出力し、PH層で実行する。

階層モデルを考えることで各層を個別に設計し実装することができ、ある層だけを入れ替えることが可能となり、部品化が容易となる。したがって、ロボットに求められる機能を体系的に分類し、オブジェクト指向技術を用いて階層モデル

に従った設計をすることにより、異なるロボット及びロボット部品間の相互運用性を向上し、新しい機能の追加や変更を容易にすることができる。ある層でロボットに指示を出せば、それより下の層についてのこまごました命令は考えなくてよくなる。例えば作業座標系の動作を指示するときに実際のアクチュエータがどう動くかについて意識する必要はない。階層モデルに沿って開発すれば、保守性がよく再利用しやすいソフトウェア部品を作ることができる。

2.3 ORCAアーキテクチャ

図3にORCAのアーキテクチャを示す。ORCAではサブシステムを基本単位としてロボットを構成する。図は一つのサブシステムを中心に描いている。ヒューマンインタフェース(HI)部や遠隔操作機器、ネットワーク入出力、データベース(DB)が外部につながっている。RT参照モデルに照らし合わせると、ロボット言語ファイルの発行がTC層、軌道生成がMC層/JC層、サーボ制御がAC層、I/O変換がIO層、I/OからパワーI/F(InterFace)、ロボット、センサがPH層に

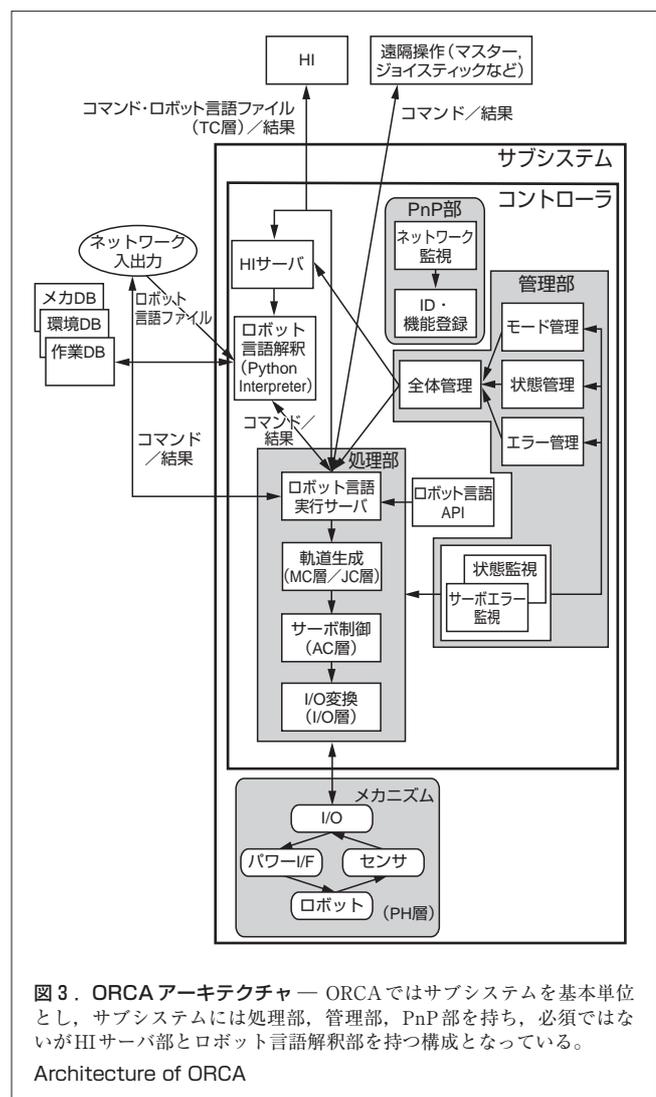


図3. ORCAアーキテクチャ ORCAではサブシステムを基本単位とし、サブシステムには処理部、管理部、PnP部を持ち、必須ではないがHIサーバ部とロボット言語解釈部を持つ構成となっている。

Architecture of ORCA

当たる。

サブシステムは、処理部、管理部、PnP部を持ち、必須ではないがHIサーバ部とロボット言語解釈部を持つ構成となっている。

処理部はロボット言語実行サーバと軌道生成、サーボ制御の三つの機能に分かれている。ロボット言語実行サーバでロボット言語コマンドを受け付け、ロボット言語API (Application Programming Interface) を利用して対応する軌道生成部 (MC層/JC層) を呼び出す。分散オブジェクト技術 (ネットワーク上に分散するソフトウェアを簡単に利用する技術) の1実装であるHORB⁽⁶⁾を利用してこのサーバを実装することで、ネットワークのどこからでもORCAインタフェースを介してロボット言語を実行することができるようになっている。軌道生成部では、コマンドに対する軌道 (例えば直線移動命令であれば直線軌道用のパラメータ) を生成し、サーボ制御部 (AC層) に送る。サーボ制御部はこの軌道を用いてフィードバック制御を行い、I/O変換部 (IO層) で実際のロボットで使うデータに変換して、ロボット (PH層) 側のI/O部へ出力する。I/Oを介してパワーI/F (サーボドライバなど) を通し、ロボットを駆動する。ロボットの動作はセンサによりI/Oを介してI/O変換部へフィードバックされる。

管理部では処理部の状態やエラーを監視し、それらをモード管理、状態管理、エラー管理へ送る。これらをまとめて管理する全体管理部でサブシステムのモード (ロボットの場合は自動実行モードや遠隔操作モードなど)、状態、エラーを管理し、実行許可や禁止などを行う。

PnP部は、ネットワークに自身がつながる、あるいは新たなORCA対応サブシステムがつながるのを監視している。自身がネットワークにつながったことを検出したら、自身のID (Identification) や提供できる機能などを通知する。

HIサーバ部は、HIを例えばウェブサービスのような手段で外部に対して提供し、HIからの入力をロボット言語解釈部やロボット言語実行サーバに伝える。ロボット言語解釈部はHIからのロボット言語ファイル (ロボット言語列のスク립トを記述したファイル、TC層に相当) を解釈して、ロボット言語実行サーバに伝える。

HIはネットワークなどでORCAサブシステムに接続し、ORCAのHIサーバの機能をブラウザなどから利用したり、HI用のクライアントソフトウェアを利用したりしてサブシステムへ人間の要求を伝える。HIからはロボット言語ファイルやロボット言語の1命令をHIサーバ、ロボット言語解釈部、ロボット言語実行サーバに対して送信し、実行結果などを受け取る。ネットワーク入出力は他のORCAシステムからのロボット言語ファイルやコマンドの送受信の機能を表している。分散オブジェクト技術を利用しているため、このようなネットワーク入出力を簡単に扱える。

3 ORCA運動制御ボード

1章で述べたように、ロボットが普及するためにはロボットを構築するためのコンポーネントが手軽に手に入り、簡単に組み合わせてロボットを作れるようなコンポーネントの流通が不可欠である。そのためORCAでは、前述のような分散オブジェクトを基盤とするソフトウェアだけでなく、このソフトウェアを動かすことができる低消費電力で低価格のハードウェアの提供をも目指している。

現在、ORCAでは分散オブジェクト技術として産業技術総合研究所で開発されたHORBを利用することでネットワークの透過性を実現している。そのためORCA対応ハードウェアの基本仕様としては何らかのオペレーティングシステム (OS (基本ソフトウェア)、モジュールの機能によってはリアルタイムOS) 上で、JavaTM (注2) が動作することが前提となる。

一般家庭で活躍するロボットのコントローラを考えると、現状のマイコンレベルでは要求される画像や音声の認識処理を満足することは困難であり、現状レベルでハードウェアを固定することは得策ではない。一方、モータ制御を中心とした動作制御系では既存のマイコンクラスでも十分に機能が達成できモジュール化に適している。そこで通信機能を持った多軸の運動制御ボードをORCA用モジュールとして開発した。

開発した運動制御ボードの仕様を表2に示し、外観 (表面) を図4に示す。TX4937は32ビットのMIPS^(注3)系RISC (縮小命令セットコンピュータ) で、FPU (浮動小数点演算プロセッサ) とPCI (Peripheral Component Interconnect) バスI/Fの内蔵を特長としている。外部通信には、現状の主要通信I/FであるEthernet, USB (Universal Serial Bus) 2.0,

表2. ORCA運動制御ボード仕様
Specifications of ORCA motion control board

項目	仕様
CPU	TX4937-300 MHz
外部IF	LAN, USB2.0 2ch, IEEE1394 2ch, SIO
入出力	AD 12ビット 16ch, DA 13ビット 12ch エンコーダ入力 12ch, DIO セレクタブル 48ch
超音波センサ	信号送受信回路 8ch
モータ制御	PWM, バルス列 12ch
その他	SO-DIMMソケット, コンパクトPCMCIAソケット
大きさ	146.05 × 126.0 mm

AD : アナログ/デジタル変換器
DA : デジタル/アナログ変換器
SIO : シリアル入出力
DIO : デジタル入出力
SO-DIMM : Small Outline-Dual In-line Memory Module
PCMCIA : PC Memory Card International Association

(注2) Javaは、米国Sun Microsystems, Inc.の米国及びその他の国における登録商標又は商標。

(注3) MIPSは、MIPS Technologies, Inc.の登録商標。

4 あとがき

ロボット技術をコンポーネント化して、そのコンポーネントを利用することで様々なロボットを作り上げ、ロボット市場を活性化するための枠組みとしてORCAを開発している。ここではORCAの概要、コンポーネント化を図るためのRT参照モデル、それに基づくORCAアーキテクチャ、更にORCA運動制御ボードについて述べた。

文 献

- (1) 尾崎文夫,ほか.分散オブジェクト技術を用いたオープンロボットコントローラ.東芝レビュー.56,9,2001,p.12-15.
- (2) 尾崎文夫,ほか.オープンロボットコントローラアーキテクチャ(ORCA).日本ロボット学会誌.21,6,2003,p.22-28.
- (3) 末廣尚士.「RTミドルウェアの基本機能に関する研究開発」の概要.ロボット.153,2003,p.1-4.
- (4) 水川 真,ほか.産業用ロボットにおけるネットワークインタフェースの標準化活動.東芝レビュー.56,9,2001,p.7-11.
- (5) 井上博允.“ロボット言語の研究課題”.日本ロボット学会誌.2,2,1984,p.3-6(この論文ではタスクレベルを作業目標レベルと呼んでいる)
- (6) 萩本順三,ほか.HORBではじめるJava分散オブジェクトプログラミング”.翔泳社.2002.

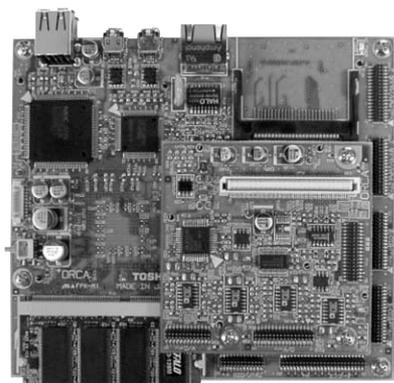


図4. ORCA運動制御ボード—ソフトウェアだけでなくハードウェアからもコンポーネント化を支援するためにORCA対応の運動制御ボードの製作を行った。

ORCA motion control board

IEEE1394(米国電気電子技術者協会規格1394)を搭載した。

モータ制御は、電圧指令型モータドライバ12チャンネル(ch), PWM(Pulse Width Modulation)指令型ドライバ12chの組合せで最大24個のモータまで対応できる。また移動ロボット用として超音波センサの信号送受信回路8chを装備した。

このボードはRT参照モデルの物理層をサポートするもので、ロボットの汎用の運動制御ボードである。このボードをORCAの更に高位層として利用するには、リアルタイム性を持つOS上でJava™とHORBを動作させる必要がある。MIPS系で動作するリアルタイムOSとしては、VxWorks^(注4)やμiTRONなどがあるが、今回は、筆者らがインテル^(注5)x86系+ART-Linuxで使っていたソフトウェア資産を有効活用するために、MontaVista^(注6)Linux^(注7)をポーティングした。

(注4) VxWorksは、Wind River Systems, Inc.の登録商標又は商標。

(注5) インテルは、米国又はその他の国における米国Intel Corporation又は子会社の登録商標又は商標。

(注6) MontaVistaは、MontaVista Software, Inc.の商標。

(注7) Linuxは、Linus Torvalds氏の登録商標。



尾崎 文夫 OZAKI Fumio

研究開発センター ヒューマンセントリックラボラトリー
主任研究員。ロボットのソフトウェアシステムの研究・
開発に従事。日本ロボット学会会員。
Humancentric Lab.



橋本 英昭 HASHIMOTO Hideaki

研究開発センター ヒューマンセントリックラボラトリー
主任研究員。ロボット制御の研究・開発に従事。日本
ロボット学会、計測自動制御学会などの会員。
Humancentric Lab.