

次世代ロボット知能化技術開発プロジェクト  
ロボット知能ソフトウェア再利用性向上技術の開発

機能仕様書  
ロボットアーム分解運動速度制御モジュール  
(Linux)

V e r . 1 . 0

2011年1月19日

R T C 再利用技術研究センター

## 改版履歷

[illegible]

# 目次

1. はじめに .....	1
1. 1. 本書の適用範囲 .....	1
1. 2. 関連文書 .....	1
1. 3. 本書を読むにあたって .....	1
2. RTC 仕様 .....	2
2. 1. RMRC（分解運動速度制御コンポーネント） .....	2
3. 特記事項 .....	10

# 1. はじめに

## 1. 1. 本書の適用範囲

本書はロボット向けミドルウェア OpenRTM 上で多軸アームロボットの分解運動速度制御を行う知能モジュールの仕様について記述した文書である。

## 1. 2. 関連文書

本書の関連文書は下表の通り。

表 1-1 関連文書

No.	文書名	備考
1	-	-

## 1. 3. 本書を読むにあたって

本書は RT ミドルウェア、RT コンポーネント(以下、RTC)に関する基本知識を備えた利用者を対象としている。RT ミドルウェア、RTC については下記を参照のこと。

OpenRTM-aist Official Website:

<http://www.openrtm.org/>

## 2. RTC 仕様

### 2. 1. RMRC（分解運動速度制御コンポーネント）

#### 2. 1. 1. 機能概要

本知能モジュールはロボットアームの分解運動速度制御を実現するものである。分解運動速度制御とは、手先の並進運動および姿勢の更新を行うために、ロボットアームの各関節の運動速度を制御することである。

本知能モジュールでは、目標とする手先位置・姿勢の指定、動作速度の上限指定、手先に取り付けるツール長とその姿勢の設定を行う機能がサービスとして提供されている。

#### 2. 1. 2. 動作環境

コンポーネント動作環境を以下に示す。

動作 OS	Ubuntu10.04
開発言語	C++
コンパイラ	g++4.4.3-1
RT ミドルウェア／バージョン	OpenRTM-aist-1.0.0-RELEASE（C++版）
依存ライブラリ(OpenRTM)	OmniORB-4.1.2-1
	libace-5.6.3-6
依存ライブラリ(その他)	行列演算ライブラリ（自作）

#### 2. 1. 3. 動作条件

実行周期	デフォルト（1000Hz）
------	---------------

## 2. 1. 4. ポート情報

### A) データポート (InPort)

名称	型	データ長	説明
Th	TimedDoubleSeq	n_size	現在の各軸角度情報 (単位 : rad)

### B) データポート (OutPort)

名称	型	データ長	説明
Vel	TimedDoubleSeq	n_size	各軸の動作角速度 n_size : 軸数 (単位 : rad/sec)

### C) サービスポート (Provider)

サービスポート名	インターフェース名	説明
ComRMRC	com_frm_ctrl	手先位置・姿勢の軌跡制御
ComRMRC	com_fk	順運動学計算
ComRMRC	com_move	算出した目標手先位置・姿勢を指定

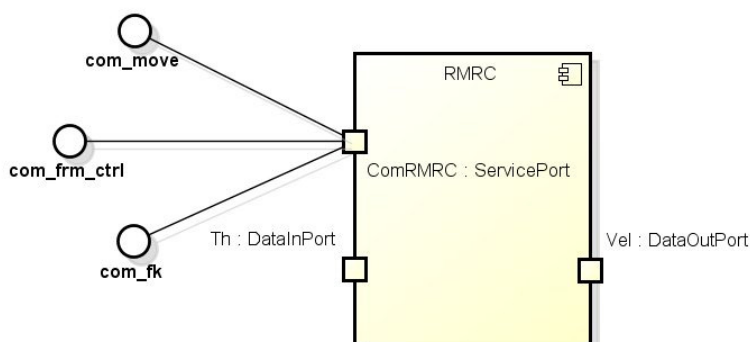


図 2-1 RMRC コンポーネント

## 2. 1. 5. コンフィグレーション情報

名称	型	デフォルト値	説明
n_size	int	7	関節数 (≥1) 本モジュールではアーム自由度のデフォルト値を7とする。
m_max	int	100	特異値分解行列の最大行数 (≥1)
min_joint_vel	vector<double>	[0.0,0.0,0.0,0.0,0.0,0.0,0.0]	各関節の最小角速度

## 2. 1. 6. 入出力データフォーマット

### 2. 1. 6. 1. 入力：Th（関節角度）

データ位置	格納値
0	第 1 軸の角速度
1	第 2 軸の角速度
2	第 3 軸の角速度
:	:
n	第 n 軸の角速度 (n : n_size)

### 2. 1. 6. 2. 出力：Vel（関節角速度）

データ位置	格納値
0	第 1 軸の角速度
1	第 2 軸の角速度
2	第 3 軸の角速度
:	:
n	第 n 軸の角速度 (n : n_size)

## 2. 1. 7. サービスポート I/F 仕様

### 2. 1. 7. 1. com\_frm\_ctrl

#### (1) set\_param

関数名	set_param			
引数	名称	型	I/O	説明
	k_pv	double	入力	時定数 (k_gain) として設定される
	v_m	double	入力	最大速度 (v_max) として設定される (単位 : mm/sec)
	k_pr	double	入力	-
戻り値	値			説明
	なし			-
説明	並進速度、回転速度計算に使用するパラメータの設定を行う。			
備考	デフォルトでは k_pv = 1.0, v_m = 200, k_pr = 200 であり、基本的に k_pv と k_pr は変更を行わない。必要に応じて最大速度の調整を行う場合には、十分注意して適切なパラメータ設定をすること。			

#### パラメータ設定実装例

```

RTC::ReturnCode_t frm_ctrl::onActivated(RTC::Uniqueld ec_id)
{
    // default
    // k_gain = 1.0, means time constant is 1 sec.
    // v_max = 200[mm/sec]
    // k_pos_rot = 200[mm], means that sizes of objects are about this size.
    m_com_frm_ctrl.set_param(1.0, 200, 200);

```

## (2) calc\_vel

関数名	calc_vel			
引数	名称	型	I/O	説明
	geo_ref	FRAME	入力	手先の目標位置と姿勢
	geo_cur	FRAME	入力	手先の現在位置と姿勢
	p_vel	VECTOR	出力	並進速度
	r_vel	VECTOR	出力	回転速度
戻り値	値			説明
	0			正常終了
	その他			異常終了
説明	手先の目標位置・姿勢、手先の現在位置・姿勢を入力として、目標到達判定及び並進速度、回転速度の計算を行う。			
備考 1	本関数は内部処理用として使用している。			
備考 2	FRAME 型は $3 \times 3$ 行列と 3 次元ベクトルを並べた $3 \times 4$ 行列で構成されている。			

FRAME 型の定義 (geo\_std.h 内)

```
#define REAL float
typedef REAL VECTOR[3];
typedef REAL MATRIX[3][3];
typedef struct {
    MATRIX mtr;
    VECTOR vtr;
} FRAME;
```

2. 1. 7. 2. **com\_fk**(1) **set\_tool**

関数名	set_tool			
引数	名称	型	I/O	説明
	frm	const Frame&	入力	手先の位置とツールの姿勢
戻り値	値			説明
	なし			-
説明	ツール長と手先座標系から見たツールの姿勢を設定する。			

Frame 型の IDL 定義 (com\_RMRC.idl 内)

```

struct Frame{
    float mat[3][3];
    float pos[3];
}

```

手先の位置と姿勢の設定実装例

```

RTC::ReturnCode_t pa10fk::onActivated(RTC::UniqueId ec_id)
{
    FRAME tmp;
    tmp.vtr[0] = 0.0;
    tmp.vtr[1] = 0.0;
    tmp.vtr[2] = 150.0
    MATRIX m_b, m_c;
    mb(PI,m_b);
    mc(PI/4.0,m_c);
    m_mul(m_b,m_c,tmp.mtr);

    m_com_fk.f_kinema.set_tool(&tmp);
}

```

## 2. 1. 7. 3. com\_move

### (1) move\_to

関数名	move_to			
引数	名称	型	I/O	説明
	xyz	Vector	入力	目標手先位置
	abc	Vector	入力	目標手先回転角度
戻り値	値			説明
	なし			-
説明	分解運動速度制御のトリガーであり、目標手先位置・姿勢を指定する。			

Vector 型の IDL 定義 (com\_RMRC.idl 内)

```
struct Vector{
    double pos[3];
}
```

目標手先位置・回転角度の指定実装例

```
RTC::ReturnCode_t pa10fk::onActivated(RTC::UniqueId ec_id)
{
    Vector ref_xyz, ref_abc;
    ref_xyz.vtr[0] = 600.0;
    ref_xyz.vtr[1] = 0.0;
    ref_xyz.vtr[2] = 500.0
    ref_abc.vtr[0] = 0.0;
    ref_abc.vtr[1] = PI
    ref_abc.vtr[2] = PI/4.0

    m_com_move.move_to(ref_xyz, ref_abc);
}
```

## 2. 1. 8. 依存モジュール

モジュール名	順運動学計算モジュール
ファイル/ヘッダ名	fk.cpp / fk.h
格納場所	本モジュールと同じフォルダ内
機能概要	本モジュール内でロボットアームの関節数、リンク長を設定し、その情報に基づきアームの順運動学計算を行う。

現在は本モジュールにおいてロボットアーム固有のパラメータ設定や、順運動学計算の記述がされている。本モジュールはロボットアーム毎に各自でファイルを用意するものとする。制御するロボットアームを変更する際は、適宜対応するファイルに差し替え、リコンパイルをする。

### 3. 特記事項

本モジュールをご利用される場合には、以下の記載事項・条件にご同意いただいたものとします。

- 本モジュールは独立行政法人 新エネルギー・産業技術総合開発機構の「次世代ロボット知能化技術開発プロジェクト」内実施者向けに評価を目的として提供するものであり、商用利用など他の目的で使用することを禁じます。
- ドキュメントに情報を掲載する際には万全を期していますが、それらの情報の正確性またはお客様にとっての有用性等については一切保証いたしません。
- 利用者が本モジュールを利用することにより生じたいかなる損害についても一切責任を負いません。
- 本モジュールの変更、削除等は、原則として利用者への予告なしに行います。また、止むを得ない事由により公開を中断あるいは中止させていただくことがあります。
- 本モジュールの情報の変更、削除、公開の中断、中止により、利用者に生じたいかなる損害についても一切責任を負いません。

#### 【連絡先】

RTC 再利用技術研究センター

〒101-0021 東京都千代田区外神田 1-18-13 秋葉原ダイビル 1303 号室

Tel/Fax : 03-3256-6353 E-Mail : [contact@rtc-center.jp](mailto:contact@rtc-center.jp)