

芝浦工業大学 機械機能工学科知能機械システム研究室 (担当:生田目祥吾,石田真一)

### ~~目次~~

#### 1.はじめに

1.1.	このRTCで出来ること2	2~3
1.2.	動作環境	- 4
1.3.	OpenRTM-aist インストール	- 4
1.4.	ネーミングサーバの起動	- 5
1.5.	RT System Editor の起動	- 6

#### 2. モータドライバ RTC

2.1.	モータドライバ RTC の概要 7
2. 2.	使用するハードウェア8
2.3.	設定用ソフトウェアのダウンロード9~11
2.4.	EPOS2 24/2の配線と初期設定12~15
2.5.	RT ミドルウェアを利用した動作確認 16~20

### 3. URG 計測 RTC

3.1.	URG 計測 RTC の概要	21
3. 2.	使用するハードウェア	22
3.3.	デバイスドライバのダウンロード	23~28
3.4.	RT ミドルウェアを利用した動作確認	29 <b>~</b> 32

### 4. 人追尾コントローラ RTC

4.1.	人追尾コントローラ RTC の概要	- 33
4. 2.	RTC ミドルウェアを利用した動作確認34	<b>↓~</b> 37

### 5. 人追尾カメラモジュール

5.1.	システム構成	38
5.2.	既存 RTC の再利用	39~41
5.3.	RTCの接続と実行	42
5.4.	さらに発展した利用法	42

再利用における注意事項 (ライセンス等)	- 43
参考文献・プログラム	- 43

## 1. はじめに

### 1.1. この RTC で出来ること

人追尾カメラモジュール制御 RTC は,以下に示す3種類の RT コンポーネントです.すべて 修正 BSD ライセンスを適用します.なお「モータドライバ RTC」ではマクソンジャパン(株) が,「URG 計測 RTC」では北陽電機(株)がそれぞれ公開しているサンプルプログラムの一 部とライブラリ利用しています.

モータドライバ RTC	LRF_Scan_Data URG_scan0	マクソンジャパン (株) で販売されているモー タドライバ EPOS2 24/2 を制御する RTC.
URG 計測 RTC	Target_Pos Now_Pos	北陽電機(株)で販売 されている測域センサ URG シリーズからデー タを取得する RTC.
人追尾コントローラ RTC	LRF_Scan_Data Target_Pos Target_Pos2 Tracking_Controller0	測域センサの計測値か ら人の存在箇所を推定 し,追尾動作指示をモ ータに出力する RTC.

これらの RTC を用いて, 搭載したカメラが人の動きに合わせて追尾して撮影するように 駆動するパンチルトカメラ雲台が実現可能です. 下図は作成した追尾カメラモジュールの 動作の様子ですが, フレームはホームセンターで購入した金属板を金鋸とハンドドリルで 加工したものですので, モータや測域センサさえあれば簡単に作成できます.



追尾カメラを使えば・・・・

追尾カメラは様々な用途で利用することが出来ます.環境に設置することで防犯カメラ として機能しますし、高度な顔認識を搭載することで受付システムとしても利用できるか もしれません.さらに、ロボットに搭載すれば人の存在や環境を認識するセンサとして利 用する事もできます.



追尾カメラ以外でも・・・・

「モータドライバ RTC」と「URG 計測 RTC」は市販の製品を制御するための RTC です. したがって,該当する製品さえ手に入れば誰でもある種の「アクチュエータ」と「センサ」が機能します. あとは既存の RTC と新しいアイディアを上手く組み合わせ,別の「コントローラ」に置き換えることで異なるシステムを構築することもできます. 測域センサの計測情報を利用して知的に動作するロボットを色々と考えてみましょう.



### 1.2. 動作環境

対応している OS は Windows のみです. 動作の確認には 64bit 版の Windows7 Home premium および Enterprise 上に C++版の OpenRTM-aist-1.1.0 をインストールした PC を使用しまし た. 開発環境は Visual C++ 2010 Professional を使用しています.



1.3. OpenRTM-aist のインストール

RT ミドルウェアに全く触ったこともないという人もいると思いますので, OpenRTM-aist を インストールする方法を説明します. 既にインストールされている場合は以下を読み飛ば してしまいましょう.

32bit Windows 環境は・・・・

OpenRTM-aist は各 RTC を動作させる上で必要不可欠ですので必ずインストールしなくては いけません. バージョンは OpenRTM-aist1.1.0 というものを利用します. ダウンロードは 以下の URL が示す OpenRTM-aist 公式 HP から行うことが出来ます. 32bit 版 Windows 環境の 場合は上記 HP に詳しい解説が載っていますので非常に簡単に設定ができます.

http://www.openrtm.org/openrtm/ja

- 64bit Windows環境は・・・・・

最近 64bit の 0S で動いている PC が随分と増えてきました. そこで 64bit 版 Windows7 環 境へのインストール手順をまとめた「インストールマニュアル」を別紙として用意しまし た. 64bit 環境を利用している場合は上記 HP と合わせてご覧いただければうれしいです.

### 1.4. ネーミングサーバの起動

 まずはネーミングサーバを立ち上げます.画面左下のウィンドウズボタンをクリックし, 「すべてのプログラム」「OpenRTM-aist 1.1(64-bit)」「C++」「tools」と辿って「Start Naming Service」をクリックして起動しましょう.



以下のようにネーミングサーバが立ち上がりますので、邪魔にならないように最少化しておきましょう。



## 1.5. RT System Editor の起動

 今度は RT System Editor を立ち上げます. 画面左下のウィンドウズボタンをクリック し、「すべてのプログラム」「OpenRTM-aist 1.1(64bit)」「C++」「tools」と辿って 「RTSystemEditor」をクリックして起動しましょう.

RT System Editor RCP						
e <u>W</u> indow <u>H</u> elp						
2 04L						
Name Ser 🕤 Repositor				- E	Properties	e 🗸 🗖
10 0 0 I I I I I I I I I I I I I I I I I	▽				Property	Value
RT localhost	-					
					-	
	Configuration View ZT Mana	ager Control View 📰 Com	posite Component View 🐹 Execution Context View 🐹 RT Lo	g View 🗢 t	5	
	Configuration View	ager Control View T Com	posite Component View 📷 Execution Context View 📷 RT Lo	g View		
	Configuration View Mana	ager Control View 🚮 Com ConfigurationSe	posite Component View ET Execution Context View ET RT Lo	g View 中間		
	Configuration View Mana ComponentName: active config	ager Control View 📷 Com ConfigurationSe name	posite Component View   22 Execution Context View   22 RT Lo tt:   Value	g View 中間		
	Configuration View Times Mane	ager Control View at Com ConfigurationSe	posite Component View at Execution Context View at RT Lo RT Lo Value	g View 中間		
	Configuration View E Mana ComponentName: active config	ager Control View To ConfigurationSe	posite Component View at Execution Context View at RT Lo R: Value	g View ロ [ 編集 通用 キャンセル		
	Configuration View and Mana ComponentName: active config	ager Control View 📷 Com ConfigurationSe	posite Component View at Execution Context View at RT Lo tc   Value	g View C ( 編集 道用 キャンセル		
	Configuration View and Mana ComponentName: active config	ager Control View at Com ConfigurationSe name	posite Component View all Execution Context View all RT Lo t: Value	g View 日 [ 編集 道用 キャンセル		
	Configuration View and Mana ComponentName:	ager Control View T Com ConfigurationSe name	posite Component View   11 Execution Context View   11 RT Lo tc   Value	g View 100 [ 100		
	Configuration View Mana ComponentName: active config	ager Control View TC Com ConfigurationSe name	posite Component View 🗱 Execution Context View 💐 RT Lo 4:   Value	g View C C C C C C C C C C C C C C C C C C C		
	Configuration View III Mann. Componenthame: active config	ager Control View at Com ConfigurationSe name	posite Component View 🗱 Execution Context View 🕷 RT Lo Rc   Value	g View で 1 構業 通用 キャンセル		
	Configuration View III Marie Componentihame: I active config	ager Control View all Com ConfigurationSe name	posite Component View at Execution Context View at Lo tr   Value	g View C ( 構成) (本中ンセル)		
	Configuration View Mann Componentiame:	ager Control View I II Com ConfigurationSe name	posite Component View   T Execution Context View   T La t:   Value	g View C ( 構築 通用 キャンセル 用語	5	

② ウィンドウ左上の「ON」マークをクリックして System Diagram を開きます.



③ System Diagram が立ち上がりました.

BT Curtom

Tu. 🔛 👹 🏭 🖓						
iame Ser 🌎 Repositor " 🗖 🕷 Syste	em Diagram 💱				Properties	1 ° °
🛅 (A) (A) 🔛 🎥 🏇 🤣 🗯 🍸					Property	Value
tr localhost					Maxon_motor0	
ROBO_CTRL09+PC host_ext					Path URI	local
					Instance Name	e Maxo
					Type Name	Maxo
					Description	Moto
					Version	1.0.0
					Vendor	Sho
					Category	Moto
					State	ACT
					properties	
					implementa	a Max
					hose energy	
					cype_name	e Max
Confi	nuration View 21 Manager Co	ntrol View III Composi	te Component View 21 Execution Context View 21 RT Log View		description	Max Mob
Confi	guration View T Manager Co	ntrol View 🔠 Composi	ite Component View 🔠 Execution Context View 🔠 RT Log View		description version	Mob 1.0.0
Confi	guration View 🐮 Manager Co	ConfigurationSet:	te Component View 🗱 Execution Context View 🗱 RT Log View	- C	description version vendor	Mob 1.0.0 Sho
Confis Compo active	iguration View T Manager Co inentName:	ConfigurationSet:	te Component View 🗱 Execution Context View 🧱 RT Log View		description version vendor category	Moto 1.0.0 Shop
Compo active	iguration View T Manager Co inentName: config	ntrol View T Composi ConfigurationSet:	te Component View all Execution Context View all RT Log View		description version vendor category activity_typ	Mob 1.0.0 Shop PER
Compo active	iguration View T Manager Co inentName: config	ConfigurationSet:	te Component View all Execution Context View all RT Log View	- 編集 通用 年ヤンセル	description version vendor category activity_typ max_instar	Mob 1.0.1 Sho Mob PER n 1
Compo active	iguration View T Manager Co nentName: config	ConfigurationSet:	te Component View   20 Execution Context View   20 RT Log View	ー 構集 適用 キャンセル	evention description version vendor category activity_ty max_instar language	Max Mob 1.0.1 Sho Mob PER n 1 C+4
Compo	iguration View at Manager Co inentName: config	ontrol View at Composition Set:	te Component View ( 截 Execution Context View ) 题 RT Log View ) Volue	編集 道用 年ヤンセル	etype_name description version vendor category activity_tyy max_instar language lang_type	Max Mob 1.0.1 Shoi Mob PER n 1 C++ com
Compo active	iguration View	ntrol View TC Composi ConfigurationSet:	te Component View   20 Execution Context View   20 RT Log View	ー 編集 適用 キャンセル	cype_shine description vendor category activity_tyj max_iestar langusge lang_type conf.default	Max Mob 1.0.1 Sho Mob P PER n 1 C++ com t 1
Compa active	iguration View 🐹 Manager Co onentName: config	ntrol View a Composi ConfigurationSet:	te Component View ( 题 Execution Context View ) 题 RT Log View	日日 	description version version category activity_by max_instar longuage lang_type conf.default	Max Mob 1.0.1 Sho Mob P PER n 1 C+4 com t 1 t 0
Compo active	iguration View 🔠 Manager Co onentName: config	ntrol View TComposi ConfigurationSet:	te Component View   20 Execution Context View   20 RT Log View	日日 	description version category activity_ty max_instar longuage lang_type conf.default conf.default	Maxi Moto 1.0.0 Shoq Moto P PERJ n 1 C++ com t 1 t 0 t 500
Comp schve	iguration Viewi 🗱 Manager Co onentName: 🗌	ntrol View TComposi ConfigurationSet:	te Component View ( 题 Execution Context View ) 题 RT Log View	- ロ 減気 ネヤンセル	description version version category max_instar language lang_bpe conf.default conf.default	Maxi Moto 1.0.0 Shog Moto p PERI n 1 C++ com t 1 t 0 t 5000
Compo active	guration View 証 Manager Co onentName: 「 config	ntrol View T Composi ConfigurationSet:	te Component View   20 Execution Context View   20 RT Log View	日本 「 「 「 「 「 「 「 「 「 」 「 」 「 」 「 」 「 」 「 」 「 」 「 」 「 」 「 」 「 」 」 」	vention version category activity_ty max_instan ianguoge iang_type conf.default conf.default conf.default	Maxi Moto Shog Moto P PERI n 1 C++ comp t 1 t 0 t 5000 t 5000
Compo schve	guration View III Manager Co mentName: config	ntrol View T Composition Set:	te Component View (語 Execution Context View) 語 RT Log View)	<ul> <li>「通用」</li> <li>「通用」</li> <li>(キャンセル)</li> </ul>	crype_same description vention vendor category max_instar language lang_type conf.default conf.default conf.default conf.default	Maxie Moto Shog Moto p PERI n 1 C++ comp t 1 t 0 t 5000 t 5000 t 0

# 2. モータドライバ RTC

## 2.1. モータドライバ RTC の概要

まずはモータドライバ RTC の概要を以下に示します. このコンポーネントはマクソンジャパン(株)が公開しているライブラリとサンプルプログラムの一部を利用しています.

RTC名	Maxon_motor								
	Target_Pos Now_Pos Maxon_motor0								
			Data Port						
	ポート種類	ポート名称	データ型	意味					
	InPort	Target_Pos	TimedLong	モ ー タ へ の 角 度 指 示 値 [deg]を受け取ります.					
1# <del>- *</del>	OutPort	Now_Pos	TimedLong	エンコーダの値から現在の 角度[deg]を送信します.					
備戍	Configuration Parameter (主要なもの)								
	名称	初期値	意味						
	abs_mode	1	角度の指示方法を選択できます. 0:現在角度からの相対角度指示 1:EPOS 起動時を原点とした絶対角度!						
	gear_para	90	各モータのギア比に応じて調整するパ ータです.						
	range_limit	90	異常動作を防 上の角度指示	ぐために, 絶対値がこの値以 は無視されます.					
	usNodeId	1	複数の EPOS を利用する場合に設定する ID です.						

### 2.2. 使用するハードウェア

この RTC で制御するハードウェアはマクソンジャパン(株)で販売されているモータドラ イバ EPOS2 24/2 です. 接続するモータは同社の DC モータである A-max モータを使用し, MR エンコーダとスパーギアヘッドを取り付けています. 使用したモータを構成する各要素の 名称と型番は以下の通りです.

	名称	型番	
DC モータ	A-max16	110072	
	GS16A-41:1	325066	Areuned uspen
	MR-256PPR	228182	
モータ ドライバ	EPOS 2	24/2	C C C C C C C C C C C C C C C C C C C

追尾カメラとして利用する場合は・・・・

EPOS 24/2 と PC は USB で接続します. EPOS 24/2 側は mini B 端子です. 追尾カメラモジュールではパン軸・チルト軸それぞれにこのモータとドライバを使用して いるので,同様のモジュールを作る場合は上記の商品が 2 組必要なので注意が必要です.

### 2.3. 設定用ソフトウェアのダウンロード

買ったばかりのモータを動かす為には少し準備が必要です. まずは下記の URL が示すマク ソンジャパン(株)の HP を表示しましょう.

http://www.maxonjapan.co.jp/index.htm

2012 年 11 月 15 日現在では以下の画面が表示されます.ページの構成が変化している場合は適宜読み替えてください.ページ上部のメニューから「製品」をクリックします.



下図のような画面になったら、左側の「maxon motor control(control electronics)」
 をクリックしてください.



③ 下図のような画面になったら「カタログデータ、各種マニュアルのダウンロードはこちらから」をクリックしてみましょう。



 ④ 現れたページの中央部「EPOS2」と書かれた周辺に以下のようなダウンロード表があり ます.以降はここから必要なプログラムをダウンロードしましょう.参照するのは 「EPOS2 24/2」の列です.

EPOSマニュアル、ソフトウェア									
タイトル	II兑 <sup>10</sup> 月	EPOS 24/1	EPOS2 24/2	EPOS 24/5	EPOS2 24/5	EPOS2 Module 36/2	EPOS2 50/5	EPOS 70/10	EPOS2 70/10
ゲッティング・スタート	はじめての運転時の設定・接続	Þ			LEb0.	\$2.24	/2		Z
ハードウェア・リファレンス	ハードウェアの説明	D	Z	ť	の列	52 24/ りを参!	熖	Þ	
ケーブル・スターティング・セット	コネクタと別売ケーブルの説明	Þ	Z	Z	A	-	因	Z	Z
Firmware Specification	ファームウェアの説明(英語)			EPOS	2	EPOS	2 🖬		
Communication Guide	通信プロトコル(CAN, RS232) の説明(英語)			EPOS		EPOS	2 🖾		
EPOS Studio とファームウェア	初期設定ソフト(886MB)				đ	1			
最新のファームウェア・ファイ ル	上記EPOS Studioに含まれてい ますが、 ここから個別にダウンロード可能 です。	ą	ą	ą	ą	ą	ą	ą	ą
VisualC++, Windows DLLと サンプル、解説	Microsoft Visual Studio Visual C++ 2005 and 6.0				ą	1			
VisualBasic, Windows DLLと サンプル、解説	Microsoft Visual Studio Visual Basic 2005 and 6.0				đ	2			
BolandC++, Windows DLLと サンプル、解説	C++ Builder 5.0 and Turbo C++10.0				ą	1			
Delphi, Windows DLLとサン プル、解説	Delphi 10.0 and 4.0				ų	1			
LabView, Windows DLLとサ ンプル、解説	LabVIEW 7.1 and 2010				ų	1			
LabWindows/CVI, Windows DLLとサンプル、解説	LabWindows/CVI 7.1 and 2010				đ	1			

⑤ 一覧から「ゲッティング・スタート」をダウンロードしてください. この pdf ファイルでは EPOS の初期設定方法について親切に説明されています. 基本的には指示通りに操作すれば設定できるので、以降の初期設定はこのファイルを参照してください. 次ページでは「ゲッティング・スタート」の補足説明をしてありますので合わせて参照することをお勧めします.

EPOSマニュアル、ソフトウェア									
タイトル	II 兑8月	EPOS 24/1	EPOS2 24/2	EPOS 24/5	EPOS2 24/5	EPOS2 Module 36/2	EPOS2 50/5	EPOS 70/10	EPOS2 70/10
ゲッティング・スタート	はじめての運転時の設定・接続	Ø	٦.	Ø	Z		Z	D	Z
ハードウェア・リファレンス	ハードウェアの説明	đ	đ		「ゲッ				
ケーブル・スターティング・セット	コネクタと別売ケーブルの説明	Þ	đ			をク	リック		
Firmware Specification	ファームウェアの説明(英語)			EPOO	_	2100	2 -		
Communication Guide	通信プロトコル〈CAN, RS232〉 の説明(英語)			EPOS		EPOS	2 🔁		
EPOS Studio とファームウェア	初期設定ソフト(886MB)				ą	]			
最新のファームウェア・ファイ ル	上記EPOS Studioに含まれてい ますが、 ここから個別にダウンロード可能 です。	ą	ą	ą	ą	ą	ą	ą	ą
VisualC++, Windows DLLと サンプル、解説	Microsoft Visual Studio Visual C++ 2005 and 6.0				ą	]			
VisualBasic, Windows DLLと サンプル、解説	Microsoft Visual Studio Visual Basic 2005 and 6.0				ų	]			
BolandC++, Windows DLLと サンプル、解説	C++ Builder 5.0 and Turbo C++10.0				ų	]			
Delphi, Windows DLLとサン プル、解説	Delphi 10.0 and 4.0				-	]			
LabView, Windows DLLとサ ンプル、解説	LabVIEW 7.1 and 2010				-	]			
LabWindows/CVI, Windows DLLとサンプル、解説	LabWindows/CVI 7.1 and 2010				ų	1			

### 2.4. EPOS2 24/2 の配線と初期設定

前述したように配線や初期設定に関しては「ゲッティング・スタート」の解説が詳しいの でそちらを参照してください.ゲッティング・スタートで指示されている手順がすべて完 了したら,このマニュアルを再び参照して以降の「2.5.RTCを利用した動作確認」に進み ましょう.以下ではゲッティング・スタートの概略と補足を述べます.

補足説明を加えた大まかな流れとしては

- 商品に付属の DVD-ROM を用いて PC にソフトウェアをインストールします.
   途中でインストールの種類を聞かれますが「Typical」で構わないでしょう.
- ② 配線を行います.

配線らしき配線は電源となる DC+12V と GND の 2 線をつけるだけです. 念のため, 以下
 に配線した EPOS の外観を示します. 電源には 12V 1.5A の AC アダプタを利用しました.



③ インストールした EPOS Studio でシステム設定をします.
 EPOS の電源を入れても、そのままでは PC 上でドライバが正常に認識されないという問題が起こりました.これに対しては次ページ以降の手順で解決できます.



④ 最後に、モータの制御ゲイン調整を行います。
 使用しているモータの使用値を参考にして指示通りに行えば問題ありません





😵 Wi	ndows セキュリティ	×
$\otimes$	ドライバー ソフトウェアの発行元を検証できません	
$\odot$	<ul> <li>              このドライバー ソフトウェアをインストールしない(N) お使いのデバイス用の、更新されたドライバー ソフトウェア どうか製造元の Web サイトで確認してください。      </li> <li>             このドライバー ソフトウェアをインストールします(I) 製造元の Web サイトまたはディスクから取得したドライバー アのみインストールしてください。その他のソースから取得 ソフトウェアは、コンピューターに危害を及ぼしたり、情報 可能性があります。      </li> <li>             ICのドライバーソフトウェアをインスト をクリック         </li> </ul>	小が存在するか - ソフトウェ した署名のない を盗んだりする くトールしま
以下の画	面が表示されればデバイスドライバが正常にインス	トールされま
J下の画 こる (C)」	面が表示されればデバイスドライバが正常にインス をクリックしたら、本来の手順に戻って設定を進め ) © ドライバー ソフトウェアの更新 - maxon motor EPOS2	トールされま うましょう.
以下の画 じる (C)」	<ul> <li>面が表示されればデバイスドライバが正常にインスをクリックしたら、本来の手順に戻って設定を進め</li> <li>『ドライバーソフトウェアの更新 - maxon motor EPOS2</li> <li>ドライバーソフトウェアが正常に更新されました。</li> </ul>	トールされま うましょう.
下の画 る (C) 」	<ul> <li>面が表示されればデバイスドライバが正常にインスをクリックしたら、本来の手順に戻って設定を進め</li> <li>『ドライバー ソフトウェアの更新 - maxon motor EPOS2</li> <li>ドライバー ソフトウェアが正常に更新されました。</li> <li>このデバイスのドライバー ソフトウェアのインストールを終了しました:</li> </ul>	トールされま うましょう.
以下の画 じる (C)」	<ul> <li>面が表示されればデバイスドライバが正常にインスをクリックしたら、本来の手順に戻って設定を進め</li> <li>『ドライバーソフトウェアの更新 - maxon motor EPOS2</li> <li>ドライバー ソフトウェアが正常に更新されました。</li> <li>このデバイスのドライバー ソフトウェアのインストールを終了しました:</li> <li>maxon motor EPOS2</li> </ul>	トールされま うましょう.
以下の画 こる (C)」	<ul> <li>面が表示されればデバイスドライバが正常にインスをクリックしたら、本来の手順に戻って設定を進め</li> <li>『ドライバーソフトウェアの更新・maxon motor EPOS2</li> <li>ドライバーソフトウェアの近常に更新されました。</li> <li>このデバイスのドライバー ソフトウェアのインストールを終了しました:</li> <li>maxon motor EPOS2</li> </ul>	トールされま

### 2.5. RTC を利用した動作確認

それでは、いよいよ RTC を利用した動作を確認してみましょう. ネーミングサービスと RT System Editor はあらかじめ起動しておいてください. (操作は 1.4. ~1.5 にて解説済)

 プロジェクトページからモータドライバRTCをダウンロードして任意のフォルダに保存 してください. 先頭のフォルダ名は「Maxon\_motor」です.

ル
<i>ν</i> μ
<i>ν</i> μ
ル
д 2 KB
0 KB

保存したフォルダ内を「build」「Release」と辿ると「Maxon\_motorComp. exe」という実行ファイルがあるのでダブルクリックして実行してください.

EposCmd.dll	2011/02/02 14:15	アプリケーショ	3,580 KB
EposCmd64.dll	2011/02/02 14:15	アプリケーショ	4,911 KB
Maxon_motor.dll		ケーショ	86 KB
🗐 Maxon_motor.exp	「Maxon_motorComp.exe	Library	1 KB
Maxon_motor.lib	をダブルクリック	マイル	2 KB
Maxon_motorComp.exe	2012/11/19 0:00	テランケーション	11 KB
rtc6196.log	2012/11/19 8:41	テキスト ドキュ	1 KB

③ 真っ黒なウィンドウが立ち上がりますので、そのままにしておきましょう.



④ Maxon\_motorComp. exe でウィンドウを立ち上げると RT System Editor 上に起動したコンポーネントが表示されるので System Diagram 上にドラック&ドロップして「All Activate」のを表す緑の三角マークをクリックしてみましょう.



⑤ System Diagram 上で RTC の表示が緑色に変化すれば接続成功です.この状態で degree 単位の角度指示値を与えると、モータは EPOS 起動時の位置を原点として指定の角度だ け動作します.エラーが出た場合は先ほど真っ黒だったコンソール上にエラーメッセー ジが現れますので参考にしてみてください.



 ⑥ モータドライバ RTC の InPort は TimedLong 型のデータポートですので OpenRTM-aist-1.1.0に付属のサンプルである「ConsoleIn」を利用して角度指示が可能 です. 実際に画面左下のウィンドウズボタンをクリックし、「すべてのプログラム」 「OpenRTM-aist 1.1(64bit)」「C++」「components」「examples」と辿って 「ConsoleInComp. exe」をクリックして起動し、試してみましょう.

ConsoleInComp.exe	
corba.id: omniORB exec_cxt.periodic.type: PeriodicExecutionContext exec_cxt.periodic.rate: 1000 exec_cxt.evdriven.type: EventDrivenExecutionContext logger.enable: N0	•
logger.log_level: INFU naming.enable: YES naming.type: corba naming.formats: %h.host_cxt/%n.rtc naming.names: ROBO_CTRLO9-PC.host_cxt/ConsoleIn0.rtc config_file: port.outport.dataport: port.outport.out:	
Port0 (name): ConsoleIn0.out - properties - port.port_type: DataOutPort	
dataport.data_type: IDL:RIC/TimedLong:T.0 dataport.subscription_type: flush,new,periodic dataport.dataflow_type: push,pull dataport.interface_type: corba_cdr	Ŧ

 ⑦「ConsoleInComp.exe」をSystem Diagram 上にドラッグ&ドロップし、先ほど起動した「Maxon\_motor」の InPort と接続してみましょう. 接続は ConsoleIn の OutPort から Maxon\_motor の InPort ヘドラッグすることで繋がります.



⑧ 再び「All Activate」を示す緑の三角マークをクリックしてみましょう.以下のように 両方のコンポーネントが緑色に変化すれば起動は成功です.



9 テストとして ConsoleIn のウィンドウ上で「80」と入力し Enter を押しましょう. Enter を押すと同時にモータが動作するはずです. 初期設定では 90°以上の値は受け付けない ようになっていますが、コンフィギュレーションパラメータを変化させることでこの制限も変更することができます.

🖸 ConsoleInComp.exe	X J
312e30000000100000000000000064000000010102000e0000003137322e333302e3133312e32350 a2c80e000000fe2bd9a45000001e0000000000000000000000000000000	0 ^ 4
Please input number: 80	
Data Listener: ON_SEND Profile::name: ConsoleIn0.out_Maxon_motor0.Target_Pos Profile::id: e96db43e-65be-40a6-8d52-401e44f64f30 Data: 80 	
Data Listener: ON_RECEIVED Profile::name: ConsoleIn0.out_Maxon_motor0.Target_Pos Profile::id: e96db43e-65be-40a6-8d52-401e44f64f30 Data: 80	
Please input number:	Ŧ

 ① モータが動作したら、次はパラメータの調整を行います.先ほど「80」と入力したにも 関わらず、あまり回転しない、あるいは回転しすぎるといった場合はコンフィギュレー ションパラメータの「gear\_para」を調節することで解決できます.
 簡単な確認方法として、モータ先端に厚紙等を張り付け、分度器を当てて「80」と入力 しましょう.もし到達した位置が 80°以上であった場合は gear\_para の値を小さく、 80°以下であった場合は gear\_para の値を大きくすることで調整してください.



① モータの動作速度を変化させたい場合は以下のコンフィギュレーションパラメータで 変更可能です.なお変更した結果デバイスの接続時に適応されますので、速度や変える には一度 Deactivate してから再接続してください.

ulProfileAcceleration	加速度
ulProfileDeceleration	減速度
ulProfileVelocity	速度

以上でモータドライバ RTC の解説は終了です. 思い通りにモータを回転させることはできたでしょうか.

## 3. URG 計測 RTC

### 3.1. URG 計測 RTC の概要

まずは URG 計測 RTC の概要を以下に示します. このコンポーネントは北陽電機(株)が公 開しているライブラリとサンプルプログラムの一部を利用しています.



### 3.2. 使用するハードウェア

この RTC で制御するハードウェアは北陽電機(株)で販売されている測域センサである URG シリーズです.動作を確認できたのは「URG-04LX-UG01」と「URG-04LX」の2種類となって います. どちらか好きな方のセンサを利用してください。

URG-04LX -UG01	调明符用	20~5, 600mm	Aland
	測此配囲	240°	
	测压性在	60∼1,000mm ±30mm	
	測距有度	測距精度 1,000~4,095mm 距離の±3%	
URG-04LX	測明符用	60~4095mm	C C C C C C C C C C C C C C C C C C C
	沢正型団	240°	
	御史李	60~1000mm ±10mm	
	· 州此村/支	1000~4095mm 距離の±1%	

どちらの測域センサにすればいいか・・・・

とりあえず追尾カメラを動作させたいなら URG-04LX-UG01(画像:上)は相対的に測距精度 が劣るものの,追尾カメラを動作させる上では問題ない範囲です. 電源も USB からのバス パワーのみで動作するので手軽に利用できる点がお勧めです.

追尾カメラ以外にも精度を必要とする用途で使用する場合は URG-04LX (画像:下) が適し ているでしょう. ただし, DC5V を別に用意する必要があります.

### 3.3. デバイスドライバのダウンロード

URG を PC に接続する際のデバイスドライバをダウンロードします. まずは下記の URL が示 す北陽電機(株)の HP を表示しましょう.

http://www.hokuyo-aut.co.jp/02sensor/07scanner/download/index.html

 2012年11月15日現在では以下の画面が表示されます.ページの構成が変化している場合は適宜読み替えてください.ページ中央の「商品ラインナップ」から使用するURGの 写真をクリックしてください.



②「ダウンロード」という項目に「URG-04LX-UG01 ドライバ:Windows 用(32bit/64bit)」 あるいは「URG-04LX ドライバ:Windows 用(32bit/64bit)」があるのでクリックしてく ださい. どちらか片方をインストールすれば前述した2種類のURG がどちらも正常に動 作するので、ダウンロードするファイルの内容は同じようです.

<sup>北陽電機株式会社</sup> 測域センサ情報	潮域センサのデータ出力(LX)タイプ専 センサの遺定や開発に必要な情報を提(	用ページです。 共します。	► English 🕨 a	周い合わせ ・リンク		
HOME製品情報	比較表から探す	ダウンロード	よくあるご質問			
製品情報 PRODUCT	<u>≪一見へ戻る</u> S URG-04LX-UG01			-		
UTM-30LX-EW  UTM-30LX  UXM-30LX-EW		■ 通称 Sir • 電源電圧	nple URG 5V(USBから供給) m			
<u>URG-04LX-UG01</u> <u>URG-04LX</u> <u>UBG-04LX-F01</u>		・ 視野角 2 ・ インタフ	40° エース USB			/
Programming Guid		~~~ <i>~</i>	~ドライハ 7	∜:Windows をクリック	用(32	2bit/64bit)」
	ダウンロード ・ <u>URG-04LX-UG01 製品</u>	土壌豊				
	<ul> <li>URG-04LX-UG01 外形</li> <li>URG-04LX-UG01 運信(</li> <li>URG-04LX-UG01 ドラ-</li> </ul>	a <u>土様書</u> イバ: <u>Windows用 (32bit/6</u>	i4bit)			
	ニーク確認い、川の村内	20-58-20				

③ 以下のようなウィンドウが現れたら「保存(S)」あるいは「名前を付けて保存(A)」をク リックしてください. 解説では「保存(S)」をクリックしました.



 ④ ダウンロードしたディレクトリを確認すると「URG\_USB\_DRIVER\_Win.zip」というファイ ルがあるので右クリックして「すべて展開」をクリックしたら同じ場所に展開してくだ さい.



⑤ 展開したファイル「URG\_USB\_DRIVER\_Win」が現れました.これでドライバのダウンロードは完了です.

<pre>eclipse342_rtmtools110-rc3_win32_ja URG_USB_DRIVER_Win delipse342_rtmtools110-rc3_win32_ja.zip</pre>	2012/1 2012/1 2012/	「URG_USB_DRIVER_Win」 があれば完了	
🔀 OpenRTM-aist-1.1.0-RELEASE_vc10_x64.msi	2012/11/	Windows インス	139,356 KB
PyYAML-3.10.win32-py2.6.exe	2012/11/	アプリケーション	323 KB
URG_USB_DRIVER_Win.zip	2012/11/	圧縮 (zip 形式)	3 KB

⑥ URG と PC を USB で接続して、ダウンロードしたドライバが正常に動作するか確認しておきましょう.
 実際に URG と PC を USB で接続してください.
 最初に接続するときは以下のような表示が出てしまいます.



- ⑦ 正しくドライバを認識させるために、「コントロールパネル」「ハードウェアとサウンド」
   「デバイスマネージャ」という順に辿ってデバイスマネージャを起動させます。
- ⑧ 一覧のうち「ほかのデバイス」下に「URG-Series USB Driver」がありますので右クリ ックして「プロパティ」を選択します。

「URG-Series USB Driver」 を右クリックして
「ブロパティ」

「ドライバーの更新(U)」をクリックしてください.

URG-Series	USB Driverのプロ	コパティ	×	
全般ド	ライバー 詳細			
1	URG-Series USB	Driver		
	デバイスの種類:	ほかのデバイス		
	製造元:	不明		
	場所:	Port_#0003.Hub_#0004		
ーデバイス Eのデ デバイ このデ さい。	Rの状態 バイスのドライバーがイ ス情報セットまたは要 バイス用のドライバーす	インストールされていません。(コ 素に選択されたドライバーがあり を検索するには、[ドライバーの更	「ドライバーの をクリッ	9更新 (U) 」 9 ク
		7۲ ) م	イバーの更新( <u>U)</u> K キャンセル	

10 「コンピューターを参照してドライバーソフトウェアを検索します(R)」をクリックしてください.



「参照(R)」をクリックしてドライバの場所を指定します.先ほどドライバをダウンロードしたディレクトリ上から「URG\_USB\_DRIVER\_Win」内の「URG\_USB\_Driver」を指定してください.指定したら「次へ」をクリックしてください.



② 以下のような警告が出ますが気にせず「このドライバーソフトウェアをインストールします(I)」をクリックしてください.

😵 Wir	idows セキュリティ
$\otimes$	ドライバー ソフトウェアの発行元を検証できません
	→ このドライバー ソフトウェアをインストールしない(N) お使いのデバイス用の、更新されたドライバー ソフトウェアが存在するか どうか製造元の Web サイトで確認してください。
	→ このドライバー ソフトウェアをインストールします(I) 製造元の Web サイトまたはディスクから取得したドライバー ソフトウェ アのみインストールしてください。その他のソースから取得した署名のない ソフトウェアは、コンピューターに危害を及ぼしたり、情報を盗んだりする 可能性があります。
	<sup>洋細の表示(</sup> 「このドライバーソフトウェアをインストールします(I)」 をクリック

③ 以下の画面が表示されればデバイスドライバが正常にインストールされました.「閉じる(C)」をクリックしてください.



 ・● 再びデバイスマネージャを確認すると以下のように表示されます. 括弧内にある COM ポート番号(ここでは COM3)をメモしておいてください. COM ポート番号はどの USB ポートを使用しているかによって異なります.



### 3.4. RTC ミドルウェアを利用した動作確認

それでは、いよいよ RTC を利用した動作を確認してみましょう. ネーミングサービスと RT System Editor はあらかじめ起動しておいてください. (操作は 1.4. ~1.5 にて解説済)

 プロジェクトページから URG 計測 RTC をダウンロードして任意のフォルダに保存してく ださい. 先頭のフォルダ名は「URG\_scan」です.

퉬 .metadata	2012/11/13 16:36	ファイル フォル	
퉬 Maxon_motor	2012/11/14 1:42	ファイル フォル…	
🌗 Tracking_Controller	2012/11/14 1:42	ファイル フォル…	
🐌 URG_scan	2012/11/14 3:15	ファイル フォル…	
systemeditor0.log	2012/11/15 22:57	テキスト ドキュ	7 KB

 保存したフォルダ内を「build」「Release」と辿ると「URG\_scanComp. exe」という実行 ファイルがあるのでダブルクリックして実行してください.



③ 真っ黒なウィンドウが立ち上がりますので、そのままにしておきましょう.



④ URG\_scanComp. exe でウィンドウを立ち上げると RT System Editor 上に起動したコンポ ーネントが表示されるので System Diagram 上にドラック&ドロップしましょう.

RT System Editor RCP	
<u>File W</u> indow <u>H</u> elp	
न्द्र 📴 📂 🖉 🔅	
🇯 Name Ser 👘 Repositor 🖵 🗖	🗟 *System Diagram 🔀
🟠 🗇 🔿 📑 🆆 🔗 🗯 🏹	
ד localhost	
ROBO_CTRL09-PC host_cxt	
DRG_scan0 rtc	
	Crite_stant

⑤ Activate する前に、System Diagram 上で「URG\_scan」を選択し、RT System Editor 下部に表示されているコンフィギュレーションパラメータ上でCOM番号を設定しましょう、デフォルトでは「COM1」ですが、先ほどデバイスマネージャ上で確認してメモを取った「COM3」に変更します。変更したら忘れずに「適用」をクリックしてください。

E	Config	ur	RT Ma	inager	KT Composi	<b>T</b> Executio	<b>RT</b> RT	Log V	
	Compor	entNa	ame: [	Configura	ationSet: defaul	t		編	ŧ
	active	confi	g	name		Value		) (a)	<b></b>
	۲	defa	ult	comNo		com3	/		
					_			キャン	セル
						「適用」			
						をクリック			
							)		

⑥ 「All Activate」のを表す緑の三角マークをクリックしてみましょう.



⑦ System Diagram 上で RTC の表示が緑色に変化すれば接続成功です.赤色に変化した場合 はエラーを意味します.先ほど真っ黒だったコンソール上にエラーメッセージが現れま すので参考にしてみてください.

💀 *System Diagram 🛛	
LIRC. scan0	
OKG_SCOND	

補足として・・・・・・・・・

エラーが発生した場合, 原因として最も怪しいのは COM ポート番号です. 再びデバイスマネージャで確認し, URG\_scan のコンフィギュレーションパラメータが一 致しているか確かめてください. ⑧ URG 計測 RTC が起動するとコンソール上に以下のような距離データが出力されます.表示される数値の単位はmmです.この表示は動作確認用に URG 前方の距離のみを出力しています. URG の前に何か物体を置いて移動させ,出力データが妥当な値であるか確認しましょう.

C:¥Users¥b09074¥workspace¥URG_scan¥build¥Release¥URG_scanComp.exe	- 0	×
Front Data =303 Front Data =304		^
Front Data =294		
Front Data =304 Front Data =297		
Front Data =301		
Front Data -507 Front Data =297		
Front Data =301 Front Data =307		
Front Data =294		
Front Data =302 Front Data =308		
Front Data =318 Front Data =205		
Front Data -280 Front Data =299		
Front Data =304 Front Data =301		
Front Data =295		
Front Data =308 Front Data =307		
Front Data =312 Front Data =317		
Front Data =302		
		Ŧ
	•	

以上で URG 計測 RTC の解説は終了です. URG で距離を計測することはできたでしょうか.

# 4. 人追尾コントローラ RTC

## 4.1. 人追尾コントローラ RTC の概要

まずは人追尾コントローラ RTC の概要を以下に示します.

RTC名	Tracking_Controller						
	LRF_Scan_Data Target_Pos Target_Pos2 Tracking_Controller0						
			Data Port				
	ポート種類	ポート名称	データ型	意味			
	InPort	LRF_Scan_Data	RangeData	URG の距離データを受信し ます.			
	OutPort	Target_Pos	TimedLong	パン軸モータへの角度指 示値[deg]を送信します.			
構成	OutPort	Target_Pos	TimedLong	チルト軸モータへの角度 指示値[deg]を送信します			
	Configuration Parameter(主要なもの)						
	名称	初期値	意味				
	max_dist	2000	計測距離の最大値を設定します. この値以上 のデータは全て「0」となります. 単位はmm.				
	object_rate	0. 8	背景距離データに対してこの割合以下の距離 が得られた箇所に物体があると認識します.				
	connect_size	100	隣接する検出物体同士の距離がこの値以下で ある場合は1つの物体とします.単位はmm.				
	lost_limit	50	人追尾中この値で示すスキャン回数以上だけ 連続して人の存在を見失った場合,追尾を中 止して再び人を探し始めます.				

### 4.2. RTC ミドルウェアを利用した動作確認

人追尾コントローラ RTC は単なるコントローラなので細かな設定は必要ありません. 早速 ですが動作確認に入ります. ネーミングサービスと RT System Editor はあらかじめ起動し ておいてください. (操作は 1.4. ~1.5 にて解説済)

このコントローラの動作確認として以下のようにモータ先端に付けた矢印が人の方向を指 し続けるというデモを行います.



 まずはハードウェアを構築します.といっても上図を参考にしてビニールテープ等で測 域センサの上にモータを固定し、モータの出力軸に矢印となるものを取り付けるだけで 構いません.矢印は EPOS 起動時に URG の前方を向くように取り付けてください. 注意点として、人を発見しやすくするために極力測域センサの計測範囲に細かい形状の 障害物が入らないようにしてください.URG の計測範囲は広いので、簡易的な対策とし て不必要な範囲をマスキングテープなどで覆ってもよいでしょう.テープ部分のデータ は「0」となります.



-34-

 ② 次にシステムを準備します. プロジェクトページから人追尾コントローラ RTC をダウン ロードして任意のフォルダに保存してください. 先頭のフォルダ名は 「Tracking\_Controller」です.

鷆 .metadata	2012/11/13 16:36	ファイル フォル…	
퉬 Maxon_motor	2012/11/14 1:42	ファイル フォル…	
Tracking_Controller	2012/11/14 1:42	ファイル フォル…	
퉬 URG_scan	2012/11/14 3:15	ファイル フォル…	
systemeditor0.log	2012/11/16 5:40	テキスト ドキュ	2 KB
systemeditor0.log.lck	2012/11/16 5:19	LCK ファイル	0 KB

保存したフォルダ内を「build」「Release」と辿ると「Tracking\_Controller. exe」という実行ファイルがあるのでダブルクリックして実行してください.

Tracking_Controller.dll		⊽∃	82 KB
Tracking_Controller.exp	「Tracking_Controller.exe」	rary	1 KB
Tracking_Controller.lib	をダブルクリック	/	2 KB
Tracking_ControllerComp.exe		ション	11 KB

④ 真っ黒なウィンドウが立ち上がりますので、そのままにしておきましょう.



⑤ Tracking\_Controller.exe でウィンドウを立ち上げると RT System Editor 上に起動したコンポーネントが表示されるので System Diagram 上にドラック&ドロップしましょう.



⑥ Tracking\_Controller はコントローラですのでActivate する前に、これまで設定して きたモータドライバ RTC と URG 計測 RTC を立ち上げて以下のように RTC 同士を接続して みましょう. 忘れがちですが、URG 計測 RTC のコンフィギュレーションパラメータで URG の COM ポート番号を再び設定しましょう.



⑦ 以上の準備が出来たら「All Activate」のを表す緑の三角マークをクリックしてみましょう. 人追尾コントローラ RTC は Activate されて最初に取得したデータを背景の環境 データとして登録します. したがって, Activate した瞬間から数秒の間は自分の体を含め, URG の測域に背景となる静止物体以外を置かないでください.



⑧ System Diagram 上で RTC の表示が緑色に変化すれば接続成功です.赤色に変化した場合はエラーを意味します.先ほど真っ黒だったコンソール上にエラーメッセージが現れますので参考にしてみてください.



⑨ Activateして数秒後にモータが動き出します. URG の前を歩きまわって矢印の向きが自分を追尾していることを確認してください.



以上で人追尾コントローラ RTC の解説は終了です. パン軸のみですが, 人追尾動作の実現 はできたでしょうか.

## 5. 人追尾カメラモジュール

### 5.1. システム構成

これまで解説した3種類の人追尾カメラモジュール制御 RTC と OpenRTM-aist 公式 HP 上で ダウンロード可能な既存 RTC および OpenRTM-aist-1.1.0に付属のサンプル RTC を活用して, 人追尾カメラモジュールを作成することが可能です.以下にその外観とシステム構成を示 します.





### 5.2. 既存 RTC の再利用

既存のRTCを再利用することで人追尾カメラはさらに高機能になります.今回はs-kurihara 様が公開している「顔検出コンポーネント」を再利用することで、取得した画像から人の 顔を検出してみましょう.ダウンロードは以下の URL に示す OpenRTM-aist 公式 HP から行 います.

http://www.openrtm.org/openrtm/ja/project/facedetect

動作確認も行いますので、ネーミングサービスと RT System Editor はあらかじめ起動して おいてください. (操作は 1.4. ~1.5 にて解説済)

 URL の先では以下のような画面が表示されますのでページ下部の「ホームページ」と書 かれている箇所をクリックします.



② 現れたページ上で「動作確認」と書かれているところが以下の画面となります.上にある「こちら」をクリックしてファイルを任意の場所にダウンロードしましょう.

動作確認		
使用するコンポーネント群		
DirectShowCamComp(OpenRTM-aistのサン 「スタート」 > 「すべてのプログラム」 > 「OpenRTM-a	ブルこ付属) ist 1,1」 > 「C++」 > 「components」 > 「opencv-rtcs」 と	と辿り、DirectShowCamComp.exeを実行します。
FlipComp(OpenRTM-aistのサンプルに付属) 「スタート」 >「すべてのプログラム」 > 「OpenRTM-a	ist 1.1」 > 「C++」 > 「components」 > 「opencv-rtcs」 と	cà当り、FlipComp.exeを実行します。
CameraViewerComp(OpenRTM-aistの 「スタート」 > 「すべてのプログラム」 > 「Oper	「こちら」をクリック	CameraViewerComp.exeを実行します。
FaceDetectComp		
<ol> <li>こちらからFaceDetect_vc10.zipをダウンロー VC9用はこちらからFaceDetect_vc9.zipをダ・</li> <li>展開したファオルダ内の"build" &gt; "Release"と</li> </ol>	ドし、適当なフォルダに展開します。 ランロードしてください。 辿り、FaceDetectComp.exeを実行します。	

③ ダウンロードしたディレクトリを確認すると「FaceDetect\_vc10. zip」というファイル があるので右クリックから「すべて展開」を選択し、同じ場所に展開してください.

	British an Inc	and have			x
Coover the Robo_CT	RL09 • ダウンロード •		<ul> <li>✓</li> <li>✓</li></ul>	ドの検索	Q
整理 ▼ ライブラリに追	加▼ 共有▼ 新しいフォルダー				0
☆ お気に入り	名前	更新日時	種類	サイズ	
🗘 Dropbox	퉬 eclipse342_rtmtools110-rc3_win32_ja	2012/11/	ファイル フォル		
ダウンロード	URG_USB_DRIVER_Win	2012/11/	ファイル フォル…		
📃 デスクトップ	🚹 eclipse342_rtmtools110-rc3_win32_ja.zip	2012/11/	圧縮 (zip 形式)	168,236 KB	
◎ 最近表示した場所	FaceDetect_vc10.zip	2012/11/	圧縮 (zip 形式)	2,121 KB	
	🔀 OpenRTM-aist-1.1.0-RELEASE_vc10_x64.msi	2012/11/	Windows インス	139,356 KB	
🗪 – /-ť i i	PyYAML-3.10.win32-py2.6.exe	2012/11/	アプリケーション	323 KB	
	URG_USB_DRIVER_Win.zip	2012/11/	圧縮 (zip 形式)	3 KB	
■ ピクチャ 「					
😸 ビデオ					
🎝 ミュージック					
№ コンピューター					
🏭 OS (C:)					
👝 DATA (D:)					
Apple iPhone					
😘 ネットワーク 🔹					
7個の項目					

④ 展開したファイル「FaceDetect\_vc10」が現れますので、そのファイル内を「FaceDetect」
 「build」「Release」と辿って「FaceDetectComp. exe」を起動しましょう.

10-1. m	a Photo a	-	· ····		x
CO V ROBO_CT	RL09 🔸 ダウンロード 🔸		<ul> <li>✓ ✓</li> <li>✓ ✓</li></ul>	ドの検索	 ٩
整理 ▼ 🔭 開く ラ	イブラリに追加 ▼ 共有 ▼ 新しいフォルダー				0
☆ お気に入り	名前	更新日時	種類	サイズ	
🗘 Dropbox	퉬 eclipse342_rtmtools110-rc3_win32_ja	2012/11/	ファイル フォル…		
🚺 ダウンロード	FaceDetect_vc10	2012/11/	ファイル フォル…		
📃 デスクトップ	URG_USB_DRIVER_Win	2012/11/	ファイル フォル…		
🧐 最近表示した場所	eclipse342_rtmtools110-rc3_win32_ja.zip	2012/11/	圧縮 (zip 形式)	168,236 KB	
	FaceDetect_vc10.zip	2012/11/	圧縮 (zip 形式)	2,121 KB	
😑 = Z T = 11	OpenRTM-aist-1.1.0-RELEASE_vc10_x64.msi	2012/11/	Windows インス	139,356 KB	
	PyYAML-3.10.win32-py2.6.exe	2012/11/	アプリケーション	323 KB	
	URG_USB_DRIVER_Win.zip	2012/11/	圧縮 (zip 形式)	3 KB	
■ ピクチャ 「					
📑 ビデオ					
👌 ミュージック					
▲ コンピューター					
DATA (D:)					
Apple iPhone					
📭 ネットワーク 🔹					
FaceDetect_vo ファイル フォルタ	:10 更新日時: 2012/11/16 10:44 ダー				



⑤ 真っ黒なウィンドウが立ち上がりますので、そのままにしておきましょう.

⑥ サンプルとして付属している「CameraViewerComp」と「OpenCVCamerComp」を起動して
 RT System Editor 上で以下のように接続したら All Activate を実行してください.



⑦ 全てのコンポーネントが正常に動作したら「CameraViewerComp」で立ち上がったウィンドウを確認しましょう.以下のようにリアルタイムで顔検出が行われています. 赤枠で囲まれた箇所が顔と判断された部分です.



### 5.3. RTC の接続と実行

最後に、このマニュアルでインストールして動作確認をして解説を行ったコンポーネント を総結集してカメラモジュールを動かします.以下のように各 RTC を起動し接続してくだ さい.準備が出来たら All activate で実行します.



このシステムではモータがパン軸・チルト軸にそれぞれについているので,設定したおよ その顔高さとURGによって得られた距離を基にチルト軸も制御しています.したがって, モジュールに対して前後に移動した場合でも人の顔位置が上下に見切れることなく撮影が 可能です.これにより様々な位置に移動した場合でも顔検出が可能であることが確認でき ました.

### 5.4. さらに発展した利用法

チルト軸の補正

顔検出を搭載したことによってモータに搭載されたエンコーダの値と取得した画像上での 顔位置から顔の高さが計測可能となるはずです.顔の高さが計測できるということは,個 人の身長に合わせてチルト軸の位置補正を行い,より顔検出を容易にすることが期待でき るかもしれません.

個人の同定

顔検出をさらに発展させて顔画像から個人を同定することができれば、追尾カメラモジュ ールの用途はさらに広がります.防犯システムや受付などが考えられますが、これを実現 するためには人の顔を明確に映す必要があります.本モジュールでは人の顔付近を撮影す ることが可能なので個人の同定もしやすいのではないでしょうか.

### 再利用における注意事項(ライセンス等)

解説の冒頭でも述べましたが、人追尾カメラモジュール制御 RTC は修正 BSD ライセンスを 適応します.また、「モータドライバ RTC」と「URG 計測 RTC」はそれぞれマクソンジャパン (株)と北陽電機(株)が公開しているサンプルプログラムの一部とライブラリ利用してい ますので研究用途でのみ利用してください.最後に、開発者および知能機械システム研究 室は本コンポーネントおよびユーザーマニュアルの利用によって生じたいかなる損害にも 責任を負いかねますのでご了承ください.

### 参考文献・プログラム

人追尾カメラモジュール制御 RTC では以下で公開されているライブラリやサンプルプログ ラムの一部を利用しています. 再利用させていただいた s-kurihara 様の顔検出 RTC のダウ ンロードページと合わせて以下に示します.

- [1] Maxon Motor: 製品概要 サーボアンプ/位置制御(last accessed 2012/11/19) http://www.maxonjapan.co.jp/product mmc.htm
- [2] 北陽電機: URG プログラミングガイド(last accessed 2012/11/19) http://www.hokuyo-aut.co.jp/02sensor/07scanner/download/urg programs/
- [3] RT-Middleware: 顔認識コンポーネント(last accessed 2012/11/19) http://www.openrtm.org/openrtm/ja/node/4629

RT ミドルウェアの基礎を学習する上で、以下のサイトや文献を参考にしました。一覧には ありませんが、産業技術総合研究所で 2012 年に行われた「RT ミドルウェアサマーキャンプ 2012」に参加することで RT コンポーネント開発の一通りの流れをみっちりと教えていただ きました。

[4] ysuga.net : RT ミドルウエア(last accessed 2012/11/19) http://ysuga.net/robot/rtm

[5] 長瀬 雅之, 中本 啓之, 池添 明宏.「はじめてのコンポーネント指向ロボットアプリ ケーション開発 RT ミドルウェア超入門」. 毎日コミュニケーションズ. 2008 年.