

アカデミックスカラロボット制御 RTC

VS\_ASR\_RTC

解説マニュアル

(第 1.0.1 版)

埼玉大学 設計工学研究室

2015 年 11 月 2 日

**【改版履歴】**

日付	版番号	改版ページ	改版内容
2015.10.31	1.0	全ページ	新規作成
2015.11.2	1.0.1	pp.13-14	「6. ソースコード, ライブラリの引用・参照箇所」追加, 雑多な修正
2015.11.6	1.1.0	pp.4-6, p.11, p.13, p.15	OpenRTM-aist C++ 1.1.1-RELEASE へバージョンアップ, RTC の仕様を詳細化, RT System Editor 上での外観, および接続例の追加, 雑多な修正

## 【目次】

【改版履歴】 .....	1
1. はじめに .....	3
1.1 概略 .....	3
1.2 本書を読むに当たって .....	3
1.3 関連文書 .....	3
1.4 関連リンク .....	3
1.5 動作環境 .....	4
1.6 開発環境 .....	4
1.7 ライセンス .....	4
2. RTC の仕様 .....	5
2.1 データポート .....	5
2.1.1 InPort .....	5
2.1.2 OutPort .....	5
2.2 サービスポート .....	5
2.2.1 プロバイダ .....	5
2.2.2 コンシューマ .....	5
2.3 コンフィギュレーション .....	5
2.4 RT System Editor 上での外観 .....	6
2.5 RTC の接続例 .....	6
3. ロボットアーム共通 I/F (SI 単位系準拠 第 1.0 版) コマンド一覧 .....	7
3.1 低・中レベル共通インタフェース .....	7
3.2 中レベルモーションコマンドインタフェース .....	7
4. RTC の作成手順 .....	9
5. 操作手順 .....	14
6. ソースコード, ライブラリの引用・参照箇所 .....	14

## 1. はじめに

### 1.1 概略

本書では、ヴイストーン株式会社製アカデミックスカラロボット制御 RTC である VS\_ASR\_RTC の詳細について述べる。

### 1.2 本書を読むに当たって

本書は RT ミドルウェアに関する基礎知識を有した利用者を対象としている。

### 1.3 関連文書

本書に関連する文書を以下に示す。

No.	文書名	発行元	版数	備考
1	ロボットアーム制御機能共通インタフェース仕様書	JARA, 埼玉大学 設計工学研究室	SI 単位系準拠 1.0 版	NEDO で規定されたロボットアーム制御機能共通インタフェースの仕様を拡張したもの。

### 1.4 関連リンク

本書に関連するリンクを以下に示す。

No.	リンク名	著作元	URL
1	CP2110 評価キット	Silicon Labs, Inc.	<a href="http://jp.silabs.com/products/interface/Pages/CP2110EK.aspx">http://jp.silabs.com/products/interface/Pages/CP2110EK.aspx</a>
2	アカデミック スカラロボット	ヴイストーン株式会社	<a href="https://www.vstone.co.jp/products/scara_robot/download.html">https://www.vstone.co.jp/products/scara_robot/download.html</a>

## 1.5 動作環境

OS	Windows7 SP1
RT ミドルウェア	OpenRTM-aist-1.1.1-RELEASE
ランタイムライブラリ	Visual C++ 2010 ランタイム

## 1.6 開発環境

OS	Windows7 SP1
RT ミドルウェア	OpenRTM-aist-1.1.1-RELEASE
RTCBuilder	OpenRTP 1.1.0-RC5
開発言語	C++
コンパイラ	Visual C++ 2010 Professional

## 1.7 ライセンス

本書，並びに本 RTC は，MIT ライセンスのもとに提供される．

## 2. RTC の仕様

### 2.1 データポート

#### 2.1.1 InPort

ポート名	データ型	データ長	説明
-	-	-	InPort なし

#### 2.1.2 OutPort

ポート名	データ型	データ長	説明
-	-	-	OutPort なし

### 2.2 サービスポート

#### 2.2.1 プロバイダ

ポート名	インタフェース型	説明
ManipulatorCommon Interface_Common	JARA_ARM::Manipulator CommonInterface_Common	低・中レベル共通コマンドイン タフェース
ManipulatorCommon Interface_Middle	JARA_ARM::Manipulator CommonInterface_Middle	中レベル・モーションコマンド 共通インタフェース

#### 2.2.2 コンシューマ

ポート名	インタフェース型	説明
-	-	コンシューマなし

### 2.3 コンフィギュレーション

名称	データ型	デフォルト値	説明
servoNum	int	5	ロボットの軸数 (5: ハンドタイプ, 3: ペンタイプ)

## 2.4 RT System Editor 上での外観

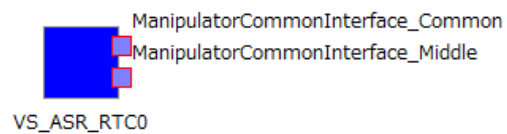


図 2.4.1 VS\_ASRTC

## 2.5 RTC の接続例

### 1) ScaraRobotControlRTC に接続

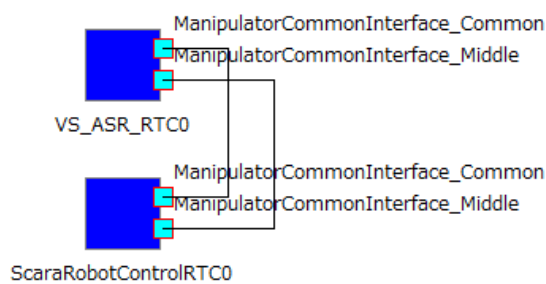


図 2.5.1 ScaraRobotControlRTC に接続した VS\_ASRTC

### 3. ロボットアーム共通 I/F (SI 単位系準拠 第 1.0 版) コマンド一覧

#### 3.1 低・中レベル共通インタフェース

No.	コマンド名	対応状況	説明
C1	clearAlarms	×	戻り値は NOT_IMPLEMENTED
C2	getActiveAlarm	×	戻り値は NOT_IMPLEMENTED
C3	getFeedbackPosJoint	○	
C4	getManipInfo	○	
C5	getSoftLimitJoint	○	RTC 内部で値を保持.
C6	getState	×	戻り値は NOT_IMPLEMENTED
C7	servoOFF	○	
C8	servoON	○	
C9	setSoftLimitJoint	○	RTC 内部に値を保持.

#### 3.2 中レベルモーションコマンドインタフェース

No.	コマンド名	対応状況	説明
M1	closeGripper	○	
M2	getBaseOffset	○	
M3	getFeedbackPosCartesian	○	
M4	getMaxSpeedCartesian	×	戻り値は NOT_IMPLEMENTED
M5	getMaxSpeedJoint	×	戻り値は NOT_IMPLEMENTED
M6	getMinAccelTimeCartesian	×	戻り値は NOT_IMPLEMENTED
M7	getMinAccelTimeJoint	×	戻り値は NOT_IMPLEMENTED
M8	getSoftLimitCartesian	○	RTC 内部で値を保持.
M9	moveGripper	○	
M10	moveLinearCartesianAbs	○	指令値はリミット値 (Cartesian) によってチェックされる.
M11	moveLinearCartesianRel	○	指令値はリミット値 (Cartesian) によってチェックされる.
M12	movePTPCartesianAbs	○	指令値はリミット値 (Cartesian) によってチェックされる.



No.	コマンド名	対応状況	説明
M13	movePTPCartesianRel	○	指令値はリミット値 (Cartesian) によってチェックされる.
M14	movePTPJointAbs	○	指令値はリミット値 (Joint) によってチェックされる.
M15	movePTPJointRel	○	指令値はリミット値 (Joint) によってチェックされる.
M16	openGripper	○	
M17	pause	×	戻り値は NOT_IMPLEMENTED
M18	resume	×	戻り値は NOT_IMPLEMENTED
M19	stop	×	戻り値は NOT_IMPLEMENTED
M20	setAccelTimeCartesian	×	戻り値は NOT_IMPLEMENTED
M21	setAccelTimeJoint	×	戻り値は NOT_IMPLEMENTED
M22	setBaseOffset	○	RTC 内部で値を保持.
M23	setControlPointOffset	×	戻り値は NOT_IMPLEMENTED
M24	setMaxSpeedCartesian	×	戻り値は NOT_IMPLEMENTED
M25	setMaxSpeedJoint	×	戻り値は NOT_IMPLEMENTED
M26	setMinAccelTimeCartesian	×	戻り値は NOT_IMPLEMENTED
M27	setMinAccelTimeJoint	×	戻り値は NOT_IMPLEMENTED
M28	setSoftLimitCartesian	○	RTC 内部で値を保持.
M29	setSpeedCartesian	○	
M30	setSpeedJoint	○	
M31	moveCircularCartesianAbs	×	戻り値は NOT_IMPLEMENTED
M32	moveCircularCartesianRel	×	戻り値は NOT_IMPLEMENTED
M33	setHome	○	RTC 内部で値を保持.
M34	getHome	○	RTC 内部で値を保持.
M35	goHome	○	原点復帰位置はリミット値 (Joint) によってチェックされる.

## 4. RTC の作成手順

RTC の一部に Silicon Labs 社から提供されるファイルを利用しているため、それらのファイルを除いたソースコードとバイナリファイルのみを公開している。そのため、本 RTC を改良するためには以下の手順に従ってソリューションファイルを生成し、インクルードファイルを追加する必要がある。

- 1) 以下の URL にアクセスし、CP2110 ソフトウェアのインストーラをダウンロードする。

<http://jp.silabs.com/products/interface/Pages/CP2110EK.aspx>

プラットフォーム	インストーラ	リリース・ノート
 Windows	<b>ダウンロード</b>	リリース・ノート
 Mac	ダウンロード	リリース・ノート
 Linux	ダウンロード	リリース・ノート

図 4.1 CP2110 ソフトウェアのインストーラのダウンロード

- 2) 任意のフォルダにダウンロードしたファイル”CP2110\_4\_Windows.exe”を実行し、CP2110 ソフトウェアをインストールする。

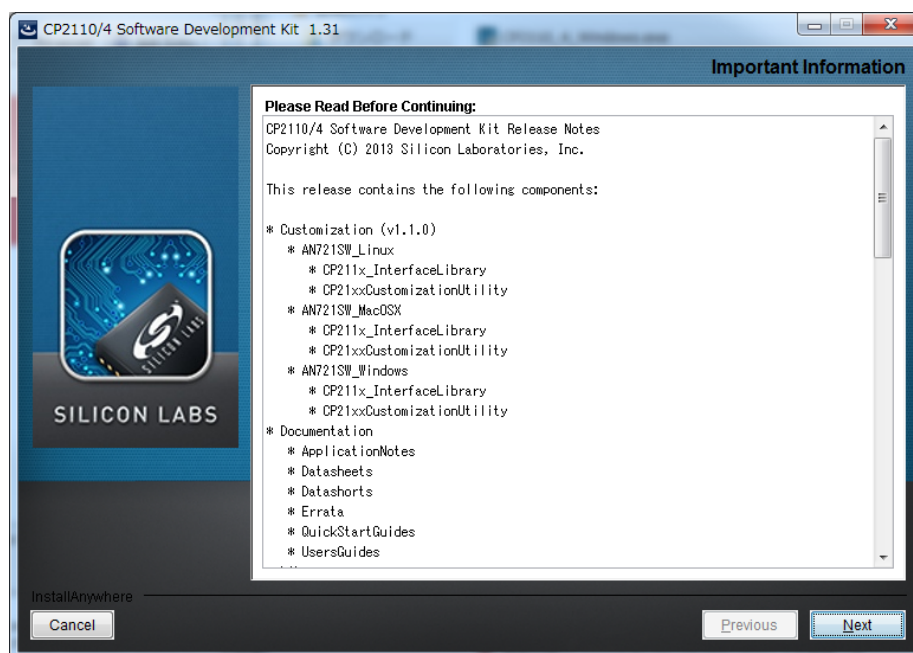


図 4.2 CP2110 ソフトウェアのインストール

- 3) インストールで展開されたヘッダファイル, ライブラリファイルに対してパスを通す.
- 3.1) 「コンピュータ」を右クリックし, 「プロパティ」を選択する.
  - 3.2) 左側に表示されるメニューから「システムの詳細設定」を選択する.
  - 3.3) 「環境変数」を選択する.
  - 3.4) 「システム環境変数」のうち, 「Path」を選択し, 「編集」を選択する.
  - 3.5) 「変数値」の末尾にインストールで展開されたヘッダファイル, ライブラリファイルがあるディレクトリのパスを追加する. インストール時のディレクトリが既定であれば, 以下の 2 箇所を追加すれば良い.

;C:\¥Silabs¥MCU¥CP2110\_4\_SDK¥Library¥Windows;C:\¥Silabs¥MCU¥CP2110\_4\_SDK¥Library¥Windows¥x86

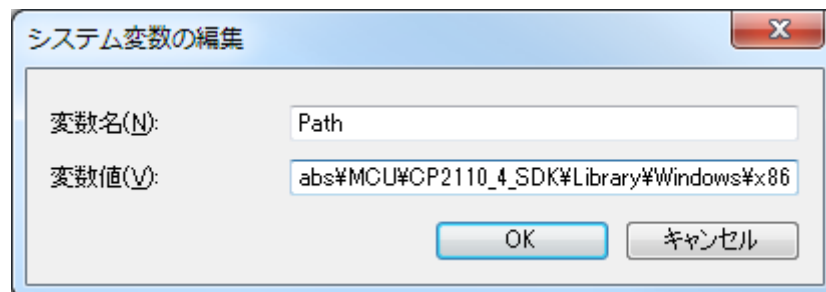


図 4.3 環境変数の設定

- 3.6) 「OK」を選択してウィンドウを閉じる.

- 4) 本パッケージにおけるソースファイルディレクトリ (..`VS_ASRTC`..`src`) を指定し、Cmake を用いてソリューションのビルドを行う。

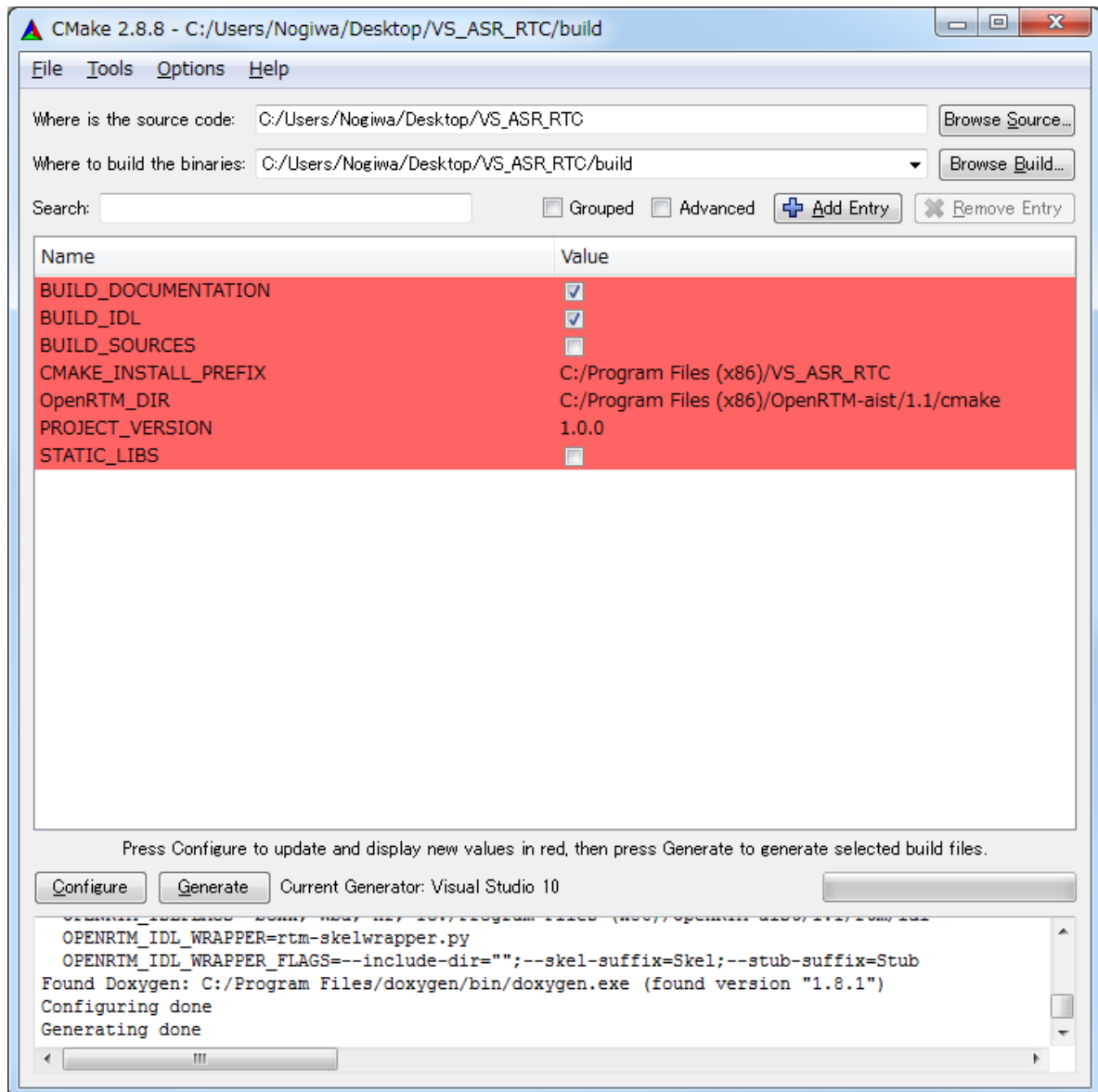


図 4.4 Cmake によるソリューションのビルド

- 5) 生成された sln ファイルからプロジェクトを開く。
- 6) ツールバーにおいて、「Debug」モードから「Release」モードへ切り替える。

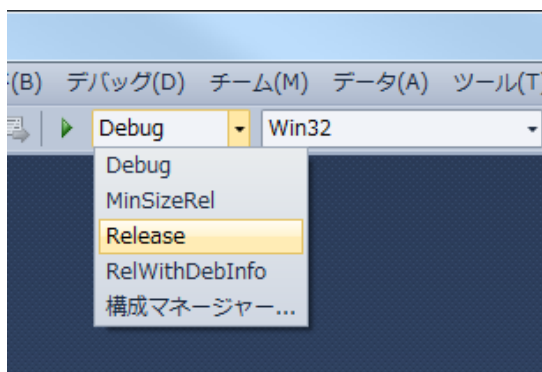


図 4.5 「Debug」モードと「Release」モードの切り替え

- 7) ソリューションエクスプローラーにおいて、プロジェクト「vs\_asr\_rtc」を右クリックし、「プロパティ」を選択する。

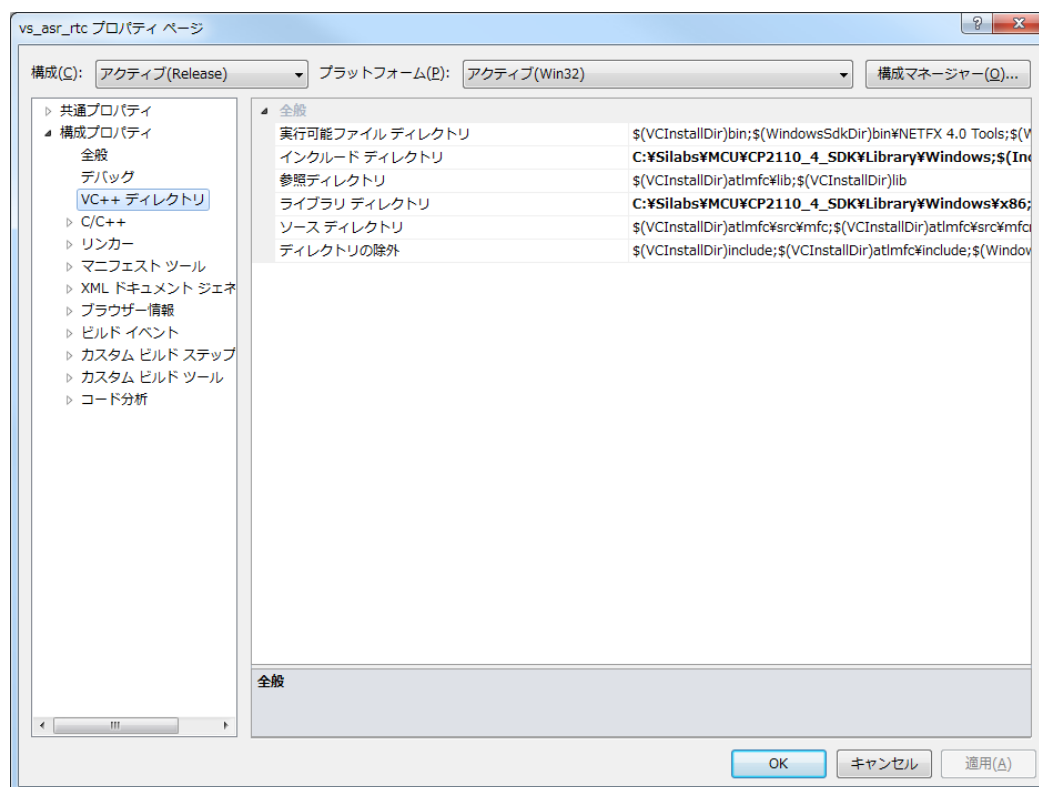


図 4.6 vs\_asr\_rtc のプロパティページ (設定後)

- 8) 「構成プロパティ」、「VC+ ディレクトリ」の順に選択する。
- 9) 「インクルードディレクトリ」を選択し、選択した際に表示される「▼」を選択し、さらに「編集」を選択する。

- 10) 「フォルダ」のアイコンを選択して新しい行を追加し、追加した際に表示される「...」のアイコンを選択する。
- 11) インクルードディレクトリを指定する。インストール時のディレクトリが既定であれば、以下に示すパスを追加すれば良い。

C:\Silabs\MCU\CP2110\_4\_SDK\Library\Windows

- 12) 「フォルダーの選択」、「OK」の順に選択する。
- 13) 「ライブラリディレクトリ」を選択し、選択した際に表示される「▼」を選択し、さらに「編集」を選択する。「フォルダ」のアイコンを選択して新しい行を追加し、追加した際に表示される「...」のアイコンを選択する。
- 14) ライブラリディレクトリを指定する。インストール時のディレクトリが既定であれば、以下に示すパスを追加すれば良い。

C:\Silabs\MCU\CP2110\_4\_SDK\Library\Windows\x86

- 15) 「フォルダーの選択」、「OK」の順に選択する。
- 16) 「OK」を選択し、vs\_asr\_rtc のプロパティページを閉じる。
- 17) ソリューションのビルドを行う。メニューにおいて、「ビルド」、「ソリューションのビルド」の順に選択する。
- 18) 正しくビルドされれば、..\RTC\VS\_ASR\_RTC\src\build\src\Release に以下に示す 4 つのファイルが生成される。

- ・ VS\_ASR\_RTC.dll
- ・ VS\_ASR\_RTC.exp
- ・ VS\_ASR\_RTC.lib
- ・ VS\_ASR\_RTCComp.exe (実行ファイル)
- ・ VS\_ASR\_RTCComp.exp
- ・ VS\_ASR\_RTCComp.lib

- 19) 実行ファイルが生成されたディレクトリに対し、次に示す 3 つのファイルを追加する。

- ・ rtc.conf
- ・ SLABHIDDevice.dll
- ・ SLABHIDtoUART.dll

“rtc.conf”は以下に示すディレクトリに存在する。

..¥RTC¥VS\_ASRTC¥src

”SLABHIDDevice.dll”および”SLABHIDtoUART.dll”はインストール時のディレクトリが既定であれば、以下に示すディレクトリに存在する。

C:¥Silabs¥MCU¥CP2110\_4\_SDK¥Library¥Windows¥x86

## 5. 操作手順

- (1) ネーミングサービスを起動する。
- (2) プロバイダ側 RTC である本 RTC (VS\_ASRTC), およびコンシューマ側 RTC (ScaraRobotControlRTC) の exe ファイルを実行する。
- (3) 本 RTC のコンフィギュレーションにロボットの軸数を設定する。
- (4) RT Syetem Editor を用いて RTC のサービスポート (ManipulatorCommon Interface\_Common, ManipulatorCommonInterface\_Middle) をそれぞれ接続する。
- (5) プロバイダ側 RTC, コンシューマ側 RTC の順で Activate する。

## 6. ソースコード, ライブラリの引用・参照箇所

VS\_ASRTC を作成するに当たって引用したソースコード, ライブラリを以下に示す。なお, 「scaraSample\_GetMotorAxis.cpp」の引用については, ヴイストーン株式会社様より許可を頂いている。

### ■ 新たに作成したソースコード内で引用・参照

「useSilabs.cpp」

From: 「scaraSample\_GetMotorAxis.cpp」 ([https://www.vstone.co.jp/products/scara\\_robot/download/scaraSample\\_GetMotorAxis.cpp](https://www.vstone.co.jp/products/scara_robot/download/scaraSample_GetMotorAxis.cpp))

- ・ int RSTorqueOnOff( HID\_UART\_DEVICE dev, short sMode ,BYTE id,int num)
- ・ int RSGetAngle( HID\_UART\_DEVICE dev ,BYTE id,short \*getParam)
- ・ int RSMove( HID\_UART\_DEVICE dev , short \*sPoss, unsigned short sTime ,BYTE id,int num)
- ・ int ReadLocalEcho(HID\_UART\_DEVICE dev ,unsigned char \*sendbuf,DWORD data\_len)

「useSilabs.h」

From: 「scaraSample\_GetMotorAxis.cpp」 ([https://www.vstone.co.jp/products/scara\\_robot/download/scaraSample\\_GetMotorAxis.cpp](https://www.vstone.co.jp/products/scara_robot/download/scaraSample_GetMotorAxis.cpp))

- ・ #define VID (0x10C4)
- ・ #define PID (0xEA80)
- ・ #define AXISLEN\_A (80.0)
- ・ #define AXISLEN\_B (80.0)

「VS\_ASRTC」

From: 「scaraSample\_GetMotorAxis.cpp」 ([https://www.vstone.co.jp/products/scara\\_robot/download/scaraSample\\_GetMotorAxis.cpp](https://www.vstone.co.jp/products/scara_robot/download/scaraSample_GetMotorAxis.cpp))

- ・ int servoNum = 0;
- ・ DWORD numDevice=0;
- ・ HID\_UART\_DEVICE dev=0;
- ・ HidUart\_GetNumDevices(&numDevice,VID,PID);
- ・ if(numDevice==0) return;
- ・ if(HidUart\_Open(&dev,0,VID,PID)!=HID\_UART\_SUCCESS){};
- ・ HidUart\_Close(dev);

■ ソースコード・ライブラリそのものを引用・参照

- ・ 「SLABCP2110.h」 (<http://jp.silabs.com/products/interface/Pages/CP2110EK.aspx>)
- ・ 「SLABHIDtoUART.h」 (<http://jp.silabs.com/products/interface/Pages/CP2110EK.aspx>)
- ・ 「CP2114\_Common.h」 (<http://jp.silabs.com/products/interface/Pages/CP2110EK.aspx>)
- ・ 「SLABHIDtoUART.LIB」 (<http://jp.silabs.com/products/interface/Pages/CP2110EK.aspx>)
- ・ 「winmm.LIB」 (<http://jp.silabs.com/products/interface/Pages/CP2110EK.aspx>)
- ・ 「ManipulatorCommonInterface\_DataTypes.idl」 (「ロボットアーム制御機能共通インタフェース仕様書\_20120224.pdf」 [http://openrtm.org/openrtm/sites/default/files/RobotArm\\_Interface1.0.zip](http://openrtm.org/openrtm/sites/default/files/RobotArm_Interface1.0.zip))  
7.1 ManipulatorCommonInterface\_DataTypes.idl (p.19)
- ・ 「ManipulatorCommonInterface\_Common.idl」 (「ロボットアーム制御機能共通インタフェース仕様書\_20120224.pdf」 [http://openrtm.org/openrtm/sites/default/files/RobotArm\\_Interface1.0.zip](http://openrtm.org/openrtm/sites/default/files/RobotArm_Interface1.0.zip), )  
7.2 ManipulatorCommonInterface\_Common.idl (pp.20-21)
- ・ 「ManipulatorCommonInterface\_MiddleLevel.idl」 (「ロボットアーム制御機能共通インタフェース仕様書\_20120224.pdf」 [http://openrtm.org/openrtm/sites/default/files/RobotArm\\_Interface1.0.zip](http://openrtm.org/openrtm/sites/default/files/RobotArm_Interface1.0.zip))  
7.3 ManipulatorCommonInterface\_MiddleLevel.idl (pp.21-22))

■ DLL ファイル等

- ・ 「SLABHIDDevice.dll」 (<http://jp.silabs.com/products/interface/Pages/CP2110EK.aspx>)
- ・ 「SLABHIDtoUART.dll」 (<http://jp.silabs.com/products/interface/Pages/CP2110EK.aspx>)

■ その他

None