RT システムビルダー 操作マニュアル

Ver.0.7.1

2015/08/02



Signal 株式会社 グローバルアシスト

1	概要	Į3
	1.1	ツール概要
	1.2	对象
	1.3	制限
2	イン	ストールと設定4
	2.1	RT システムビルダーのインストール4
	2.2	RT システムビルダーの初期設定6
	2.3	RT システムビルダー画面の起動
3	モデ	・・
	3.1	OPENRTM-AIST 事前定義型インポート機能7
	3.2	IDL ファイル定義型インポート機能
	3.3	RTC プロファイルインポート機能11
4	RT	コンポーネント設計用モデル作成機能12
	4.1	RT コンポーネントの指定方法12
	4.2	インポートした RTC ブロックの利用方法14
	4.2.	1 データポートのみ持つ RT コンポーネントの場合14
	4.2.	2 サービスポートを持つ RT コンポーネントの場合14
	4.3	RT コンポーネント本体の情報設定15
	4.4	コンフィギュレーションセット情報の設定17
	4.5	データポート情報の設定18
	4.6	サービスポート情報の設定19
	4.7	サービスインターフェース情報の設定
5	RT	C プロファイルの生成
6	デフ	/オルト値設定
	6.1	基本設定
	6.2	アクティビティ・ドキュメント設定
	6.3	ポート設定
7	改版	瓦履歴24

目 次

1 概要

1.1 ツール概要

RTシステムビルダーは、SysML モデルから RT コンポーネントのプロファイル(RtcProfile)を自動生成する ためのツールです。本ツールはモデリングツール Enterprise Architect のアドインとして実装されています。 本ツールを用いることで、上位工程で行ったモデリング結果から、RT コンポーネントの開発まで繋げること が可能となります。また、既存の RT コンポーネントのプロファイル情報や IDL で定義したデータ型情報をイ ンポートして、モデル構築時に利用する事も可能となります。

本ツールの動作に必要な環境は以下の通りです。

No.	環境	備考
1	モデリングツール Enterprise Architect (http://www.sparxsystems.jp/)	注意:バージョン 11.x 以降で動作確 認を行っております。
2	MDG Technology for SysML (http://www.sparxsystems.jp/products/MDG/mdgsysml.htm)	ご 使 用 に な ら れ る Enterprise Architect に合ったバージョンを利用 してください
3	.Net Framework Ver.4 (http://www.microsoft.com/ja-jp/default.aspx)	

また RT コンポーネントの開発を行うためには、以下の開発環境をインストールしてください。

No.	環境	
1	OpenRTM-aist-1.X.X (http://www.openrtm.org/)	RTコンポーネントが動作するために 必要な RT ミドルウェアです。お使い の環境に合ったバージョンをインス トールしてください。
2	RTCBuilder (http://www.openrtm.org/openrtm/ja/node/30)	RT コンポーネントのソースを自動生 成するツールです。

1.2 対象

本ドキュメントでは、作成した SysML モデルから、RT コンポーネント向けプロファイルを作成する方法を説明しています。SysML 自体の説明や、Enterprise Architect の操作方法に関する説明、RT ミドルウェア (OpenRTM-aist)の説明、RTCBuilder の説明などにつきましては、各ツールの操作マニュアルなどを参照してください。(なお、以降の説明中の各図は Windows7 のスクリーンショットです。)

1.3 制限

本ツールは、OpenRTM-aist を対象に開発されたものです。その他の RTC プラットフォームに対する動作 は想定しておりません。また、本ツールは Windows 向けに開発されたツールです。その他の OS 上での動作 確認は行っておりません。

2 インストールと設定

RT システムビルダーは、UML モデリングツール EnterpriseArchitect のアドインとして実装されています。 このため、事前に EnterpriseArchitect 本体および SysML アドイン(MDG Technology for SysML)をインストー ルする必要があります。(EnterpriseArchitect のインストール方法については、ツールに付属のマニュアルを 参照してください。)

2.1 RT システムビルダーのインストール

RT システムビルダーのインストールは、インストール用バッチファイル「RTSystemBuilderSetup.bat」を実行して行います。こちらのバッチファイルを実行後、EnterpriseArchitectを起動し、上部メニュー中の「アドイン・拡張」の部分に「RTSystemBuilder」が追加されていれば、インストールは完了です。

※インストール用バッチファイルは、レジストリの情報を変更するため、管理者権限のあるユーザが「管理者 として実行」で実行してください。

アドイン・拡張(X) 設定(S) ウィンドウ(M) GML ・ BPMN ・ CSVファイルからの入出力 ・ DFD ・	ヘルプ(H) 漆 译 ダ X … 留 。
IConixアドイン・ PFD・ RTSvstemBuilder・	設定)
SysML/ドインについて(A) SysML SysML >-ケンス回チェック メッセージクイック編集 名前重視チェック 接視スタイル変更 非同期メッセージの互換チェック モデルの読み込み モデルの出力 アドインサブウィンドウ(A) アドインの管理	 インボート スパークスSNATL 動画デモ ドキュメントライン 製品セミナー 無料アドイン 連携・間違製品 PDF-ゼロから始める 最新インストーラの メールニュース リリースメモ ユーザーフォーラL

図 2.1.1 RT システムビルダーのインストール

もしも、上記メニューが表示されていない場合は、上部メニュー中の「アドイン・拡張」中の「アドインの管理」を選択し、【アドインマネージャ】を起動してください。【アドインマネージャ】内の「有効なアドイン」リストの中に「RTSystemBuilder」が含まれている場合は、インストール自体は成功しています。右側の「起動時に読み込み」欄にチェックを入れて、Enterprise Architect 本体を再起動してください。

有効なアドイン	状態	起動時に読み込み
ysMLサポート	有効	~
ysML2BConvertor	エラー - アド	
ッセージクイック編集	有効	~
TAアドイン	有効	✓
and the set		
TSystemBuilder	有効	✓
Print State - L	19.50	
FDI评值	有効	✓
F互換チェック	有効	Y
(8月		
TSystemBuilder		

図 2.1.2 アドインマネージャの設定

インストールバッチを実行しても【アドインマネージャ】内に「RTSystemBuilder」の項目が追加されていない 場合には、ツールのインストールに失敗しています。以下の手順で手動インストールを行ってみてください。

- 01.RT システムビルダの DLL ファイル(RTSystemBuilder.dll)を、Enterprise Architect 本体のインストールディ レクトリにコピーしてください。
 - →デフォルト設定のまま Enterprise Architect をインストールしている場合、インストール先は以下のディ レクトリとなります。

C:¥Program Files (x86)¥SparxSystems Japan¥EA¥

- 02.コマンドプロンプトを起動し、EnterpriseArchitect本体のインストールディレクトリに移動してください。その後、アセンブリ登録ツール(RegAsm.exe)を実行してください。
 - →アセンブリ登録ツール(RegAsm.exe)は、通常、以下のディレクトリに存在します。

<u>「単化 6:372F プロンナト</u> Ev**P**Fogram Files (388) WScardSystems Japan/FAC:Windows/Microsoft.NETFFramework 40.30319(Modelsan.pee Systems Japan/FAC:Windows/Microsoft.NETFFramework Second Microsoft (60) AET Framework Assembly Registration Utility 4.0.30319.1 Covorinth (10) Microsoft Coronation 1989-2004, All rights reserved. 型(江家代に登録されました。 C:VProgram Files (388)VScardSystems Japan/FAC。

図 2.1.3 アセンブリ登録ツール

03.「スタート」メニューの「プログラムとファイルの検索」もしくは「プログラムを指定して実行」にて、"regedit"と 入力し、レジストリエディタを起動します。

04. 左側のウィンドウで、"HKEY_CURRENT_USER¥Software ¥SparxSystems¥EAAddins"を選択します

鳶 レジストリ エディター				
ファイル(E) 編集(E) 表示(⊻) お気に入り(<u>A</u>)	ヘルプ(出)			
😟 🔒 SolidWorks	▲ 名前	種類	データ	
		REG_SZ	(値の設定なし)	
😟 🕀 🔒 Sota				
B B SOURCENEXT				
📄 🔒 Sparx Systems				
😟 🍌 EA400				
😟 🍌 EAAddins				
SysMLSupport2				
😟 🎍 Sparx Systems Japan				
😟 🍌 SparxSystems Japan				
😟 🛄 srac	_			
testapp	그 -			
コンピューター¥HKEY_OURRENT_USER¥Software¥Sparx Systems¥EAAddins				

図 2.1.4 レジストリエディタ

05.上記「EAAddins」を右クリックして、新規にキーを作成し、アドインの名称を入力します。

→アドインの名称は任意です。DLL のファイル名や名前空間と一致している必要はありません。ここで付けた名称が EnterpriseArchitect のアドインマネージャに表示されます。

06.作成したキーに、「RTSystemBuilder.Connector」と設定します。

<u>ピレクスはります/0−</u> ファイル(F) 編集(E) 表示(x) お知(太り(x) ∧ x,7(h)	
0 Spec: System 1 6 m 1 m m 9 m 0 Spec: System CA18 (# - S) (# C - S)	Conta Conta
T/P'1-P-WHKEY CURRENT LISERNSoftwarehSparx Systematie Adding	T // 1 - P - WHEY CURRENT USERS (Inservices: Systematic Adding RTS) steen builder

図 2.1.5 レジストリの設定

以上の操作で、RT システムビルダーの情報が Enterprise Architect に登録されます。Enterprise Architectを再起動し、上部メニュー中の「アドイン・拡張」の部分に「RTSystemBuilder」が追加されていることを確認してください。

※上記の手動インストールを行っても、上部メニュー中の「アドイン・拡張」の部分に「RTSystemBuilder」が追加されない場合は、開発元までご連絡ください。

2.2 RT システムビルダーの初期設定

RT システムビルダーでは、各種ファイルを生成する際に、一時的な作業ディレクトリを使用します。このため、最初に作業ディレクトリの指定を行ってください。

RTシステムビルダー用作業ディレクトリの設定は、上部メニュー中の「アドイン・拡張」メニューから 「RTSystemBuilder」→「設定」→「作業ディレクトリ設定」を選択してください。ディレクトリ選択画面が表示され ますので、対象ディレクトリを選択してください。

_			フォルダーの参照	X
	アドイン・拡張(X) 設定(S) ウィンドウ(W)/Lプ(H)	佐養ご さちした地会しアノゼヤン	_
	GML	· 🐁 🕾 ダ 🗙 🗄 🖀 💷 🗈 A 📾 🏣 🚷 🦯	「F未ナイレントリを指定していたさい	
	BPMN			_
	CSVファイルからの入出力			
	DFD		田 (二) クリンフリ 田 (名) ホームグループ	
	FTA	•	🕀 🥦 GA	
	Iconixアドイン	•		
L	PFD		■ ♥■ ネットワーク	
	RTSystemBuilder	 設定 作業ディレクトリ設定 		
Г	SysMLアドインについて…(A)	インポート ・ デフォルト値設定 ・	🔒 old	
	SysML	■ 人バークスシステムスジャパン Webサイト		
	SysML		新しいフォルダーの作成(N) OK キャンセル	
	ミーケンフ図チェック	📃 ドキュメントライフラリ		//

図 2.2.1 作業ディレクトリの設定

2.3 RT システムビルダー画面の起動

RT システムビルダーでは、各種設定を行う画面をアドイン・サブウィンドウとして用意しています。アドイン・サブウィンドウが表示されていない場合、上部メニュー中の「アドイン・拡張」メニュー中の「アドイン・サブウィンドウ」のチェックを ON にしてください。「アドイン」というタブ名の画面がアドイン・サブウィンドウとして表示されます。

) P	ドイン・拡張(X) 設定(5) ウィンドウ(W)_ ヘルプ(H)	
	GML	・ 決 深 ダ × 1 部 回 回 /	
	BPMN		
81	CSVファイルからの入出力		
	DFD	•	
	FTA	•	
	Iconixアドイン		
	PFD		
	RTSystemBuilder	• D 115/2	
	SysMLアドインについて_(A)		- 1 x
	SysML	, 2 スパークスシステムズジャ	
	SysML	 ・ ・	20,741/k±06 DU291/k±06242.5mm-F DU291/k±06242.5mm-F D0201094002266224.5mm-F
	シーケンス図チェック	・ ビードキュメントライブラリ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	*t01~64
	メッセージクイック編集	 ・ ・	モジュール概要 ※2中 ※アフォルト値 Add
	名前重複チェック	 日 第科アドイン 	×パーラシン
	接続スタイル変更	 ● 連携・同連副語 ● DELETION 2 NO # 7 E - 1 	*<48
	非同期メッセージの互換チェック	 ・ ・ ・	xbrdy
	モデルの読み込み		IHARE
	モデルの出力	• 0 JU-3XE	間発言語 C G++ C Python C Java
¥	アドインサブウィンドウ(A)	🏊 ユーザーフォーラム(掲示!	出力先 Ref
	7141200BHE 7	ドインサブウィンドウ(4)	RTC計線情報設定 アクライビテイ体報設定 ドキュシント体制設定
_	121	自のアドインを作成・追加するこ	
	2	で、独自の情報の出力などに利用	RTC [DataPort] ServicePort] ServicePort
	Ent	terprise Architectを拡張する強力な	H イ + H \RIG (シーケンス回チェック / 居住と特下 /
	利	良です。	Q ダイアグラムの概要 ● ノート 圏ブロバティ ↓ タグ付きぎ アドイン

図 2.3.1 アドイン・サブウィンドウの表示

3 モデル作成サポート機能

本機能は、RT システムビルダーでRTコンポーネントの設計を行う際に、既存コンポーネントの情報や事前定義されたデータ型を再利用して、モデル作成をサポートする機能です。

3.1 OpenRTM-aist 事前定義型インポート機能

本機能は、OpenRTM-aistで事前定義されている各種型情報を、RTシステム設計時に利用するための機能です。各種標準仕様で定義されている型情報を SysML のブロック要素としてインポートします。

上部メニュー中の「アドイン・拡張」メニューから「RTSystemBuilder」→「インポート」→「OpenRTM-aist 定義 型インポート」を選択するか、RT システムビルダー用画面内の「OpenRTM-aist 型インポート」ボタンをクリッ クしてください。【インポート対象選択】画面が表示されますので、モデル内にインポートを行いたいデータ型 を選択し、「OK」をクリックしてください。

なお、本機能でインポート可能なデータ型は以下のとおりです。

•OpenRTM-aist 標準/拡張型

OpenRTM-aist がデフォルトで用意しているデータ型。BasicDataType.idl, ExtendedDataTypes.idl, InterfaceDataTypes.idl で定義されている内容が対象。「RTC」パッケージ配下にインポート。

・カメラ機能共通インターフェース

「カメラ機能共通インタフェース仕様書(第2.1版)」で定義されているデータ型。「Img」パッケージ配下にインポート。

・自律移動機能共通インターフェース

「自律移動機能共通インタフェース仕様書(第1.0版)」で定義されているデータ型。「IIS」パッケージ配下 にインポート。

・ロボットアーム制御機能共通インターフェース

「ロボットアーム制御機能共通インタフェス仕様書(SI単位系準拠 第1.0版)」で定義されているデータ型。 「JARA_ARM」パッケージ配下にインポート。

GML	2	🗑 ダ 🗙 🤅 😫	🖬 🗈 A 🕲 🔚 🚷 🖊	
(0)7= (1 + 0 + 0 + 1 + +				× • × 200
C3V / 7 1 / / / / / / / / / / /				<u>9</u> 2
000				8
FTA	,			+
Iconixアドイン	•			
PFD	•			
RTSystemBuilder	• 1	81座		
SysMLアドインについて…(A)		インボート	OpenRTM-aist定義型インボー	- 15
SvsML	. 📼	スパークスシス	2°	TM いいままま インパート
0.48	. 🧕	動画デモ	1017741	OpenKTM-astated - P
31/2 0.01				

プロファイル生成	RTCプロファイルインボート IDLファイル定義型インボー OpenRTM-aist定義型インボート
米ゼシュール名属 Contra モジュール名属 第100 メパーフェン 第100 ホベガ名 第201 ホカデュリ 100	uxifon 〒 米データ型 米デフタルト植 Add DEL IFH載な空
開発言語 C C++ C Python C Java 出力先 Ref RTC詳細細設定 アカライビライ情報設定 PA120-16構成定	
RT0 [DataPort] ServicePort] ServiceHerface]	
◎ タイアガラムの振奏 ● ノート ◎ ブロパティ ◎ タガオき待 アドイン	

 インボード対象変収

 データ名

 OpenRTM-aist構準型/拡展型

 カメラ機能共通インターフェース

 自律移動機能共通インターフェース

 ロボットアーム制御機能共通インターフェース

 OK
 全選択
 全解除
 キャンセル

図 3.1.1 OpenRTM-aist 定義型インポート

インポートを実行すると、作成中モデルのルートパッケージ配下にインポート対象パッケージが追加されます。そして追加した対象パッケージ配下に各種データ型用のブロックが追加されます。



図 3.1.2 OpenRTM-aist 定義型インポート後の状態

なお、ルートパッケージ配下にインポート対象パッケージと同一名のパッケージが存在する場合には、以下の警告画面が表示され、処理を継続するかを確認します。ここで「はい」を選択した場合、既存のパッケージとは別パッケージとしてインポート対象パッケージが追加され、データ型の追加が行われます。

※既存のパッケージ内にインポートされるのではなく、同一名称のパッケージを新たに追加してインポートを 行います。



図 3.1.3 OpenRTM-aist 定義型インポート処理の確認

本機能は、ユーザが独自に定義したデータ型情報を、RTシステム設計時に利用するための機能です。 IDL ファイルの中身をパースし、定義されている型情報を SysML のブロック要素としてインポートします。

上部メニュー中の「アドイン・拡張」メニューから「RTSystemBuilder」→「インポート」→「IDL ファイル」を選択 するか、RTシステムビルダー用画面内の「IDL ファイル定義型インポート」ボタンをクリックしてください。【ファ イル選択】画面が表示されますので、モデル内にインポートを行いたい IDL ファイルを選択し、「OK」をクリッ クしてください。

) アドイン・拡張(X) 設定(S) ウィンドウ(W) ヘルプ(H)	
3 GML · 🗞 🐨 🔗 🗶 🖓 🖬 🖬 🗛 🖓 🗮 🙆 🦯	
BPMN	
CSVファイルからの入出力 ・	
DFD •	
FTA +	
Iconixアドイン・	
PFD	
RTSystemBuilder , 設定 ,	
SystMI アドイン(につし)て…(A) インボート OnenRTM-aid市美型インボート	
syunda SvcMi	
シューケンフロチェック 創 ドキュメントラ	
	▲ 廿 ×
<u></u>	TM-aist定義型インボート
※モジュール名 Configuration	
モジュール概要 「―――――――――――――――――――――――――――――――――	Add
*バーブョン	DEL
*~54	
※カテゴリ	The attent
開発言語 C++ C Python C Java	
出功先	
RTO詳細情報設定 アクティビラィ情報経設定 ドキュメント情報設定	
RTC DataPort ServicePort ServiceInterface	
IT T F F NINA / フラスロリエンス 小樹住ChTF /	

図 3.2.1 IDL ファイル定義型インポート

対象 IDL ファイルを選択し、インポートを実行すると、作成中モデルのルートパッケージ配下に IDL 内で定義されている各種データ型用のブロックが追加されます。なお、追加対象のデータ型は以下となります。

・構造体:struct で定義された型。内部メンバ情報とともにインポートされる。

・列挙型: enum で定義された型。内部メンバとともにインポートされる。

・型定義:typedef で定義された型。宣言元の型と宣言先の型がインポートされる。

また IDL ファイル内でモジュール(module)が定義されている場合、各モジュールに対応したパッケージ構造 も追加され、定義に従った形でブロックが追加されます。



図 3.2.2 IDL ファイル定義型インポート後の状態

なお、インポート先のモデル階層にインポート対象の型と同一名称、同一定義のブロックが既に存在する 場合には、何も処理は行いません。

しかし、インポート先のモデル階層にインポート対象の型と同一名称、異なる定義のブロックが既に存在 する場合、以下のような【IDL インポート】画面が表示され、対象の型が一覧表示されます。インポート元の IDL ファイル内の定義を優先する型に対してチェックボックスを ON にしてください。チェックボックスが ON に なった型情報に対応するブロックは、モデル内の定義が IDL ファイル内の定義に更新されます。(チェックボ ックスが OFF の型については、何も処理を行いません。)

IDLインボート
以下の型情報が重複しています インボート元とは異なる型が既にモデル内に存在しますが,型情報を更新しますか?
型名
📃 Info::Unit::AlarmType
Info::Unit::InfoMember

図 3.2.3 インポート対象選択画面

3.3 RTC プロファイルインポート機能

本機能は、既存の RT コンポーネントの情報を、RT システム設計時に利用するための機能です。RTC プロファイル(XML ファイル)の内容をパースし、定義されている RT コンポーネントを SysML のブロック要素としてインポートします。

上部メニュー中の「アドイン・拡張」メニューから「RTSystemBuilder」→「インポート」→「RTC プロファイル」を 選択するか、RT システムビルダー用画面内の「RTC プロファイルインポート」ボタンをクリックしてください。 【ファイル選択】画面が表示されますので、モデル内にインポートを行いたい RTC プロファイルを選択し、 「OK」をクリックしてください。

	アドイン・拡張(X) 設定(S) ウィ	ンドウ(W) ヘルプ(H)
	GML	• 🗞 🕾 🞸 🗙 🗄 💷 🗈 🗛 📾 🔚 🔍 🦯
	BPMN	
	CSVファイルからの入出力	,
	DFD	•
	FTA	,
	Iconixアドイン	•
	PFD	,
	RTSystemBuilder	設定 ・
	SysMLアドインについて…(A)	インポート OnenRTM-aist定美型インポート
	SysML	
	SysML	
	シーケンス図チェック	
	メッセージクイック編集	
アドイン	-	▼ 4 >
プロファイル生成		RTCブロファイルインボート IDLファイル定義型インボート OpenRTM-aist定義型インボート
※干型1~ 此 名		Configuration
モジュール概要		※名称 ※データ型 ※デフォルト値 Add
*バージョン		DEL
*心身名		
※カテゴリ		246028cm
FRANK STATES OF A		
開発言語 C C++ (C Python C Java	But [
開発言語 ○ C++ (出力先	C Python C Java	Ref
開発言語 C C++ (出力先 RTC詳細情報設定	C Python C Java アクティビティ情報	Ref 設定 ドキュ火ント体報設定
間発言語 C C++ (出力先 RTC詳細情報設定 RTC. DataPort ServiceP	C Python C Java アカティビティ情報 ort Service/Interface	Ref
開発言語 C C++ + 出力先 RTC詳細情報説定 RTC DataPort ServiceP	C Python C Java アクティビティ情報 ort] ServiceInterface] ノス団チェック / 周行と操作 /	Ref

図 3.3.1 RTC プロファイルインポート

対象プロファイルを選択し、インポートを実行すると、作成中モデルのルートパッケージ配下に「Imported」 というパッケージが追加されます(既に存在する場合には追加されません)。そして、対象 RTC プロファイル 内で定義されている「カテゴリ名」と同一名称のパッケージが追加され、その配下に RTC に対応するブロック が追加されます。

プロジェクトブラウザ
🙆 💁 😤 🔠 🐘 📄 • 🗐 • 🔒 🖡 🎯
🖂 🍋 プロジェクトルート
🖃 📃 RTCSample
🛨 🔁 🎞 S
🕀 🛅 Img
😑 🚞 Imported
🖃 🧰 Category
🖂 🔚 < <block>> Flip</block>
-🤄 flippedImage: CameraImage
📲 originalImage: CameraImage
ConfigurationSet>> flipMode
🕀 🗎 Info
📖 📼 🧰 тара арм

図 3.3.2 RTC プロファイルインポート後の状態

なお、インポートされたブロックには、選択した RTC プロファイルで定義されていた情報が自動的に格納されます。また、ポート情報、コンフィギュレーションセット情報が含まれる場合には、それらに対応した要素 (ポート、属性)も自動的に追加されます。更に、ポートに設定された型、コンフィギュレーションセットに設定された型が既存のモデルに存在しない場合には、型定義に対応するブロックも自動的に追加されます。

※一部、RTC プロファイルの中でインポートできない情報も存在します。詳しくは第4章を参照してください。

4 RT コンポーネント設計用モデル作成機能

本機能は、SysML のブロック定義図および内部ブロック図の情報を用いて、RT コンポーネントのプロファ イルを生成するための機能です。

なお、作成したモデルの各種情報をデフォルト表示するため、事前に以下の設定を行っておくと便利です。 上部メニューの「ツール」→「オプション」を選択し、表示された【オプション】画面で「要素」を選択。「ポートとパ ートの型を常に表示」を選択。



図 4.1.1 ツールのオプション設定

4.1 RT コンポーネントの指定方法

RT コンポーネントは、システムを構成する要素の一部となりますので、SysML のブロックを用いて定義します。

作成したモデル中のどの要素をRTコンポーネントとするかを指定するためには、ブロック定義図内で該 当ブロックに、以下の手順でステレオタイプ「RTC」を追加してください。

01.該当ブロックをダブルクリックもしくは右クリックで表示されるコンテキストメニューから「プロパティ」を選択 02.表示された【プロパティ】画面で、「ステレオタイプ」右側の「…」ボタンをクリック

03.表示された【ステレオタイプ】画面で、「新規」をクリック

04.表示された【新規ステレオタイプ】画面で、「RTC」と入力し、「OK」ボタンをクリック

05.【ステレオタイプ】画面に追加された「RTC」のチェックボックスを ON にし、「OK」をクリック

※上記手順の 03,04 につきましては、初めてステレオタイプを追加する場合の手順となります。2回目以降 は、ステレオタイプ「RTC」は追加済みですので、こちらの手順は不要です。



図 4.1.1 ステレオタイプ「RTC」の追加

ブロック定義図でステレオタイプ「RTC」を追加した要素を親に持つ「プロパティ」を内部ブロック図内で選択 すると、RT システムビルダー用画面内に各種情報が表示されます。また、付属の「ポート」を選択した場合、 ポートの種類に応じて、データポートの情報、サービスポートの情報がそれぞれ表示されます。

プロファイル生成		RTCブロファイルインポート	IDLファイル定義型インボート OpenF	TM-aist定義型インボート	
	Configur	ation	//		
* モジュール・石 PEN王 PX モジュール概要	※名称	※データ型	※デフォルト値	Add	
※バージョン 10				DEL	
※ペダ名 GA					
*/ */ */ */ */ */ */ */ */ */ */ */ */ *				FX10F0 -	
開発言語 C C++ C Python C Java					
出力先	Ref				
RTC詳細情報設定 アクティビティ情報設定 ドキュメント情	報設定				
RTC DataPort ServicePort ServiceInterface					
RTC 人シーケンス図チェック 人層性と操作 /					
	2. 2時画像 2. 2 9 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5			<u>ب</u> د	
アドイン				▼ ‡ ×	
プロファイル生成		RTCプロファイルインボート	IDLファイル定義型インポート OpenF	RTM-aist定義型インボート	
*术	Documentation				
ポート型 深度画像	概要説明			_	
方向 jin 💌	データ型				
変数名 c	テータ数				
表示位置 LEFT 🔽	意味				
	単位				
	発生頻度·周期			*	
	创.理诗声度• 图图				
	~±./2/3C //4/991			<u>*</u>	
RTC DataPort ervicePort ServiceInterface					

図 4.1.2 RTC 情報、ポート情報の表示

SysMLモデル中と実際のRTコンポーネントでは各種名称を変更したい場合も多いため、RTコンポーネン ト本体、データポート、サービスポート、コンフィギュレーションセットの名称につきましては、「別名」が優先さ れます(もしも「別名」が設定されていない場合は、「名称」の内容を使用します)。モデルと実際の名称を変更 したい場合には、「別名」を利用してください。なお、SysMLのモデルエディタ画面では基本的に「名称」が表 示されます(設定によって変更可能)。RTシステムビルダー用画面内では、「別名」が設定されている場合に は「別名」が、「別名」が設定されていない場合には、「名称」が表示されます(RTC プロファイルに設定される 名称と同じ名称が表示されます)。

4.2 インポートした RTC ブロックの利用方法

RTC プロファイルインポート機能を用いて、既存のRTC をインポートした場合には、既にステレオタイプが 設定されていますので、生成されたブロックをドラッグアンドドロップ(DnD)するだけでRTC として使用可能で す。

4.2.1 データポートのみ持つ RT コンポーネントの場合

単純に対象ブロックを DnD するだけで利用可能です。DnD 時に表示される設定画面では、「動作の選択」 は「そのまま配置」、「付属要素」は「全て」を選択してください。

要素の貼り付け:Flip(クラス) 🔀	
動作の選択 そのまま記置 「付原要素 全て ▼	
	orteinslimere: Dementimere + top/doi: nt = 0

図 4.2.1 データポートのみの RTC の利用

※インポート RTC ブロックを利用した場合、内部ブロック図内で対象ポートのプロパティ画面を表示しても、 名称や型、方向などの詳細情報を変更することはできません(「OK」ボタンが選択できないようになってい ます)。もしもこれらの情報を修正したい場合には、ブロック定義図もしくは【プロジェクトブラウザ】内で該当 ポートのプロパティを変更してください。ただし、この方法を用いて変更した内容は、対象ポートを利用して いる全ての要素の影響しますのでご注意ください。



図 4.2.2 インポート RTC ブロックの属性変更

4.2.2 サービスポートを持つ RT コンポーネントの場合

上記の「データポートのみを持つ RT コンポーネントの場合」と同様な手順で、モデル内に追加します。しか し、この状態ではサービスインターフェースが表示されていませんので、サービスポートを右クリックして表 示されるコンテキストメニュー中から「付属要素」を選択してください。そして表示された【付属要素】画面右上 の「継承要素も表示」チェックボックスを ON にしてください。一覧画面にインターフェースが表示されますので、 「名前」列の右側のチェックボックスを ON にして「閉じる」ボタンを選択すると、モデル内にサービスインターフ ェースが表示されます。



図 4.2.3 サービスインターフェースの表示

4.3 RT コンポーネント本体の情報設定

ブロック定義図でステレオタイプ「RTC」を追加した要素を親に持つ「プロパティ」を内部ブロック図内で選択 すると、RT システムビルダー用画面内の「RTC」タブが自動的に選択され、RT コンポーネントの各種情報が 表示されます。

なお、本ツールで指定できる内容は、RTCBuilderと同等の内容となります。このため、各情報の詳細につきましては、RTCBuilderのマニュアルをご参照ください。

※RTCBuilder の「コンフィギュレーション・プロファイル」中の「RT-Component Configuration Parameter」およ び「ドキュメント情報」中の「バージョンアップログ」、「言語・環境」情報中の「環境」情報につきましては、本 ツールでは未対応です。もしもこれらの情報を設定したい場合には、RTCBuilder を用いて設定を行ってく ださい。

アトイン						<u></u>
プロファイル生成	× _		RTCプロファイルインポート	IDLファイル定義型インポート	OpenRTM-aist定	E義型インポート
※モジュール名	地図生成	Configura	tion			
モジュール概要		※名称	※データ型	※デフォ	サルト値	Add
※バージョン	10					DEL
*べ考名	GA					
*75_1						
						詳細設定
開発言語	C C++ C Python C Java					
出力先	Ref					
RTC詳細情	報設定 アクティビティ情報設定 ドキュメント情報設定	1				
		-				
RTC DataPor	t ServicePort ServiceInterface					

図 4.3.1 RT コンポーネントの情報

画面左側は、RT コンポーネントの基本情報となります。※印がついた項目は必須入力項目です。ただし、 「モジュール名」につきましては、割り当てられたブロックの名称が設定され、本画面で変更する事はできま せん。「モジュール名」を変更したい場合には、該当ブロックの名称を変更してください。また、その他の項目 については、必要に応じて設定を行ってください。

更に、本ツールでは複数 RT コンポーネントの情報を設定、プロファイル生成することができるため、RTC プロファイルの出力先を個別に指定する事が可能です。

RTコンポーネントの詳細情報を設定する場合には、画面左下の各ボタンを選択してください。

「RTC 詳細情報設定」ボタンを選択すると、以下のような【RTC 詳細情報】画面が表示されます。「コンポーネント種類」は必須項目となります。それ以外の項目はオプションとなりますので、必要に応じて設定を行ってください。

RTC詳細情報	
コンポーネント型	STATIC
アクティビティ型	PERIODIC
コンポーネント種類	🔽 DataFlow 🥅 FSM 🔲 MultiMode
最大インスタンス数	1
実行型	PeriodicExecutionContext
実行周期	1000
概要	
	<u> </u>
RTC Type	
ОК	キャンセル

図 4.3.2 RT コンポーネント詳細情報の設定

「ドキュメント情報設定」ボタンを選択すると、以下の【ドキュメント設定】画面が表示されます。こちらの項目も全てオプションとなっておりますので、必要に応じて設定を行ってください。

ドキュメント設定		
概要説明	1	* *
入出力		* *
アルゴリズムなど		4 ¥
作成者·連絡先	aaa	*
ライセンス,使用条件	bbb	4
参考文献		4
OK		キャンセル

図 4.3.3 ドキュメント情報の設定

「アクティビティ情報設定」ボタンを選択すると、以下の【アクティビティ設定】画面が表示されます。設定を 行いたいアクティビティに対応するボタンを選択すると【アクティビティ詳細設定】画面が表示されますので、 必要な情報を設定してください。

なお、【アクティビティ詳細設定】画面で「ON」に設定されたアクティビティは、【アクティビティ設定】画面でボ タンが青色となります。

アクティビティ設定			
「コンポーネントの初期化と終了タ	処理に関するアクション		
onInitialize	onFinalize		
実行コンテキストの起動と停止(に関するアクション		
onStartup	onShutdown		フカティバティディ書語会
Alive状態におけるコンポーネン	トのアクション		79)1()1計構設定
onActivated	onDeactivated	onAborting	アクティビティ名 onInitialize O ON O OFF
onError	onReset		Documentation 動作說兒別
ーDataFlow型コンポーネントのアク	1932		
onExecute	onStateUpdate	onRateChanged	
	/		
onAction			
 Mode型コンポーネントのアクショ	עע		
onModeChanged			
ОК		キャンセル	OK ++>>セル

図 4.3.4 アクティビティ情報の設定

4.4 コンフィギュレーションセット情報の設定

RT コンポーネントが持つコンフィギュレーションセットは、ブロックの属性として定義します。「RTC」タブ内の左側を利用して、定義することが可能です。

新規コンフィギュレーションセットを追加したい場合には、対象要素を内部ブロック図内で選択した後、 「Add」ボタンを選択してください。逆に、既存のコンフィギュレーションセットを削除したい場合には、対象要素を内部ブロック図内で選択した後、グリッド内で該当項目を選択し、「Delete」ボタンを選択してください。

-Configuration			
※名称	※データ型	※デフォルト値	Add
NewConfig_0	int		
			DEL
			=**4m=7.00

図 4.4.1 コンフィギュレーションセットの追加/削除

新規追加されたコンフィギュレーションセットは、各種設定がデフォルト設定のままとなっています。各コンフィギュレーションセットの「名称」「デフォルト値」につきましては、グリッド内で直接編集が可能です。それ以外の項目を変更したい場合には、該当項目を選択し、「詳細設定」ボタンを選択してください。以下の 【ConfigurationSet 詳細設定】画面が表示され、各項目の設定を行うことができます。

🛃 Configura	tionSet詳細設定	
パラメータ名	NewConfig_0	
データ型	int	_
デフォルト値	,	_
変数名	1	_
単位		_
制約条件 -	,	
		-1
指定なし:空 範囲指定:< 列挙型指定 配列型指令 ハッシュ型指	注日 指定信(即間値(ex:000) -> (<-> (<-> (<->(<->(<->(<->(<->(<->(<->(<->(<->(<->)
-Widget		
	▼ Step	
text(テキスト slider(スライ spin(スピナ) radio(ラジオ: checkbox(チ ordered_list	ボックス):刺除で行着に値。範囲指定が可能。 約:刺除でな何望の範囲指定が必要。別は一緒をstepで指定可能 本タン:刺除で列撃型発症にしばたか。 方、中かで利擎型発症しばないない。 たマクホックス):刺れで列撃型指定しばないかって指定が必要 (0.2人):刺れて列撃型指定が必要	能
Document -		
データ名		_
デフォルト値		-
概要説明		
		-
単位		
データ範囲		
制約条件		*
		Ŧ
OK	キャンセノ	k

図 4.4.2 コンフィギュレーションセットの編集

なお、コンフィギュレーションセットの情報は、対象プロパティに割り当てられたブロックの属性として定義 されています。このため、上記の手順で変更を行った場合には、この親ブロックの属性が変更されますので ご注意ください。

また、すでにブロックに定義済みの属性をコンフィギュレーションセットとして使用した場合には、属性のステレオタイプに「ConfigurationSet」を追加してください。

4.5 データポート情報の設定

RT コンポーネントのデータポートは、フロープロパティを持つポートとして定義します。内部ブロック図内で 該当要素を選択すると、RT システムビルダー用画面内の「DataPort」タブが自動的に選択されデータポート の各種情報が表示されます。

※ステレオタイプ「RTC」を追加した要素を親に持つ「プロパティ」内に定義されたポートが対象となります。

アドイン				▼ # ×
プロファイル生成		RTCプロファイルインボート IDLファイル定	議型インポート	OpenRTM-aist定義型インポート
※煮-卜名	Documentation			
ポート型 深度画像	概要説明			×
方向in	データ型			
変数名 [c	データ数			
表示位置 LEFT 🔽	意味			
	単位			
	発生頻度·周期			
	処理速度·周期			
RTC DataPort ServicePort ServiceInterface				

図 4.5.1 データポート情報の設定

「ポート名」「ポート型」「方向」につきましては、対象ポートの情報が表示され、本画面で変更する事はできません。これらの情報を変更したい場合には、該当ポートの情報を変更してください。

ボート:入力画像	×
名前	入力画像
型	深度画像
	☑ 型名を表示
向き (Direction)	in 💌
共役 (IsConjugated)	false
種類	Port 💌
その他のプロパティ	OK キャンセル

図 4.5.2 ポート情報の編集

また、その他の各種情報につきましては、どれもオプション扱いとなっておりますので、必要に応じて各種 項目を設定してください。

4.6 サービスポート情報の設定

RT コンポーネントのサービスポートは、フロープロパティを持たない通常のポートとして定義します。内部 ブロック図内で該当要素を選択すると、RT システムビルダー用画面内の「ServicePort」タブが自動的に選択 されサービスポートの各種情報が表示されます。

アドイン			→ # ×
プロファイル生成	RTCプロファイルインボート	IDLファイル定義型インポート	OpenRTM-aist定義型インポート
※ボート名 Test 表示位置 LEFT			
Documentation			
概要說明			
レド概要説的			
RTC DataPo ServicePort ServiceInterface			

図 4.6.1 サービスポート情報の設定

「ポート名」につきましては、対象ポートの情報が表示され、本画面で変更する事はできません。「ポート 名」を変更したい場合には、該当ポートの情報を変更してください。

ポート:Test	×
名前	Test
型	
	□ 型名を表示
向き (Direction)	none
共役 (IsConjugated)	false
種類	Port 💌
その他のプロパティ	OK キャンセル

図 4.6.2 ポート情報の編集

また、その他の各種情報につきましては、どれもオプション扱いとなっておりますので、必要に応じて各種 項目を設定してください。

4.7 サービスインターフェース情報の設定

RT コンポーネントのサービスインターフェースは、フロープロパティを持たない通常のポートに追加された インターフェースとして定義します。内部ブロック図内で該当要素を選択すると、RT システムビルダー用画 面内の「ServiceInterface」タブが自動的に選択されサービスインターフェースの各種情報が表示されます。

アドイン					▼ ₽ :
プロファイル生成			RTCプロファイルインボート	IDLファイル定義型インボート	OpenRTM-aist定義型インポート
*//ንቃ-71-አ名	RequiredInterface 1	Documer	itation		
方向	Required	概要説明			*
インスタンス名	· [引援	τ		A
変数名		戻り値	ī		~
※IDLファイル	Ref				
※インターフェース型		例外	×		
IDLパス	Ref	事前条件			
		事後条件			
			<u> </u>		<u>*</u>
RTC DataPort Se	rvicePort ServiceInterface				
DIC / S	()				

図 4.7.1 サービスインターフェース情報の設定

なお、「インターフェース型」を選択する際には、事前に「IDL ファイル」を設定してください。対象 IDL 内で定 義されているインターフェース情報がドロップダウンリスト内に表示されますので、使用したいインターフェー スを選択してください。

また、モデル中でサービスインターフェースを追加したい場合には、対象ポートを右クリックして表示され るコンテキストメニュー内から「子要素の追加」を選択し、「要求インターフェース」もしくは「提供インターフェー ス」を選択してください。

→間	A ₩ 1	All - 🍬 - 🔜 - 1 🗦 🎸 🎤 66' - 🍥		
移動推		アドイン・拡張(X)	•	
	***	プロパティ(P) Alt+E	nter	
		追加設定 	•	
		子要素の追加	•	要求インターフェース
移動指令		付属要素		提供インターフェース
市り伊		関係する要素の追加		ノートの追加
	°131	イート・イントの作成(K) Ctrl+4	Alt+D	制約の追加
	-	ドキュメントの生成		根拠の追加
				問題の心治力

図 4.7.2 サービスインターフェースの追加

「ServiceInterface」タブの「インターフェース名」「方向」につきましては、対象インターフェースの情報が表示され、本画面で変更する事はできません。これらの情報を変更したい場合には、該当インターフェースの 情報を変更してください。

公開インターフェース	X
インターフェース: <mark>RequiredInterface2</mark>	
· 種類 ○ 提供 ○ 要求	I
<u>QK</u> ++>tel/(C)	

図 4.7.3 サービスインターフェース情報の編集

5 RTC プロファイルの生成

本機能は、モデルに設定した情報から RTC プロファイルを生成するための機能です。モデル内に複数の RT コンポーネントが定義されている場合、全ての RT コンポーネントのプロファイルを生成します。

モデル内の全ての RT コンポーネントへの情報設定が完了後、RT システムビルダー用画面左上の「プロ ファイル生成」ボタンを選択してください。モデル内に含まれる RT コンポーネントが一覧表示されます。

]-I	1~下生成对象選択		
	コンポーネント名	出力先	15-
	測域センサ制御	I¥work	
Г	距離画像センサ制御		開発言語が指定されていません
	障害物回避		開発言語が指定されていません
	地図生成		開発言語が指定されていません
	自己位置推定		開発言語が指定されていません
	経路生成		開発言語が指定されていません
	移動制御		開発言語が指定されていません
	ОК	全選択	全解除キャンセル

図 5.1.1 RTC プロファイルの生成

RTC プロファイルを生成する項目のチェックボックスを ON にして「OK」を選択すると、対象 RT コンポーネントのプロファイルが指定したディレクトリ内に出力されます。

なお、RTコンポーネントに必要な必須項目が指定されていなかったり、設定した情報に不備が存在する場合には、一覧表中でグレー表示され、チェックボックスは選択できないようになります。エラー内容が一番右の列に表示されますので、表示内容に従って設定情報の修正を行ってください。

生成した RTC プロファイルから、RT コンポーネントのコードを生成する方法、作成したコンポーネントを実行する方法などにつきましては、RTCBuilder のマニュアルを参照してください。

また、SysMLのポートの場合、「方向」として「inoout(入出力)」を指定することができますが、OpenRTM-aist ではこのような方向のデータポートを作成することができません。このため、モデル中のポートとして「inout」 を指定した場合には、自動的に2つのポートが生成されます。この際、ポートの「ポート名」および「変数名」 につきましては、元の名称に「In」「Out」の接尾語が追加されます(OpenRTM-aist では単一コンポーネント内 に同一名称のポート、もしくは同一変数名のポートを生成することができないためです)。この場合、生成し た RTC プロファイルを再度読み込んでも、元のモデルとは定義が異なってインポートされますのでご注意く ださい。

6 デフォルト値設定

RTシステムビルダーでは、RTコンポーネントの各種情報について、デフォルト値を予め設定することが可能です。設定したデフォルト値は、新規 RTコンポーネントの設計を行う際に、自動的に設定されます。

各種設定は、上部メニュー中の「アドイン・拡張」メニューから「RTSystemBuilder」→「設定」→「デフォルト 値設定」を選択してください。

Pドイン・拡張(X) 設定(S) ウィンドウ(W) GML →	ヘルプ(H) 漆 留 ダ X : 留 圖 副 A 國 ☷ & /
BPMN ・	
DFD •	
FTA +	
Iconixアドイン・	
PFD •	
RTSystemBuilder •	設定 作業ディレクトリ設定
SysMLアドインについて…(A)	インポート・ デフォルト値設定・・ 基本設定
SysML •	マンスパークスシステムスジャパン Webサイト アクティビティ・ドキュメント
SysML •	2 動画デモ ポート
シーケンス図チェック・	 ジードキュメントライブラリ ジー製品やミナー



6.1 基本設定

【基本設定デフォルト値設定】画面では、RTコンポーネント本体に関する情報およびコンフィギュレーション セットの情報に関するデフォルト値を設定する事が出来ます。ここで設定した値は、「RTC」タブ内、【RTC 詳 細情報】画面および【ConfigurationSet 詳細設定】画面で使用されます。

基本設定デフォルト値設定	オプション
Amage 2017 アメリートレート Basic モジュールボデゴリ インダ名 モジュールガデゴリ コンポーネント型 マ アクティビティ型 マ コンポーネント種類 マ 最大インスなンス数 1 実行周期 1000 -Configuration 名称	
データ型	- 29 Java
変数名	
デフォルト値	WB.Net
制約条件	
単位	
ОК ++>тел	開いる ヘルプ他

図 6.1.2 基本設定デフォルト値設定

なお、「モジュール名」につきましては、ブロックの名称を利用しているため、デフォルト値を設定する事はで きません。また、「バージョン番号」のデフォルト値を変更したい場合には、上部メニューの「ツール」→「オプシ ョン」を選択し、表示された【オプション】画面で「要素」を選択し、「バージョン」欄の値を修正してください。

6.2 アクティビティ・ドキュメント設定

【アクティビティ・ドキュメントデフォルト値設定】画面では、RT コンポーネント本体のアクティビティ情報、ドキ ュメント情報のデフォルト値を設定することが可能です。ここで設定した値は、【アクティビティ設定】画面、【ア クティビティ詳細設定画面】および【ドキュメント設定】画面で使用されます。

アクティビティ・ドキュメントデフォルト値設定	
- Activity	
• ON C OFF	
	et
● ON C OFF C ON ● OFF ● ON C OFF C ON ● OFF C OF	OFF
onExecuteonStateUpdateonRateChanged	
CON OFF CON OFF CON OFF	
onAction	
© ON ○ OFF	
-onModeChanged	
© ON ⊙ OFF	
Documentation	
概要説明	^
	<u> </u>
	*
アルゴリズムなど	
	-
作成者・連絡先	<u>^</u>
ライセンス,使用条件	<u>^</u>
S A BA	*
	44500

図 6.2.1 アクティビティ・ドキュメントデフォルト値設定

6.3 ポート設定

【ポートデフォルト値設定】画面では、データポート、サービスポートおよびサービスインターフェースのデフ オルト値を設定することが可能です。ここで設定した値は、「DataPort」タブおよび「ServiceInterface」タブで使 用されます。

ボートデフォルト値設定	
- データポート	
サービスインターフェース	
インスタンス名	
変数名	
ОК	キャンセル

図 6.3.1 ポートデフォルト値設定

<u>7 改版履歴</u>

日付	バージョン	内容
2015/03/09	0.7.0	β版向け新規作成
2015/08/02	0.7.1	プロファイル生成機能修正