

RT ミドルウェア(RTM)の概要

規格統一の歴史

- 規格統一 1.0[機械] フート(20 or 30 or 50cm) → SI 単位
- 規格統一 2.0[電気] 送電(直 or 交)→ 交流送電(100V, 240V・・・50Hz,60Hz)
電池(直流) → 乾電池(単 1, 単 2, 単 3)
- 規格統一 3.0[情報] 各社ハード→ PC/AT 互換機, Mac
各社 OS→ Windows, MacOS
小規模ネットワーク→ インターネット
- 規格統一 4.0[R T] (物理空間と情報空間の結合)
混在→RT ミドルウェア(OMG という国際標準化団体で標準化)→ 未来
ROS(ユーザー数増加), IoT, インダストリー4.0



乾電池は、屋井先蔵が発明。
寸法・電圧などが、
国際電気標準会議で規格化。
単 3 乾電池など、
世界中で入手可能で便利。

RT ミドルウェア (RTM) とは

RT(ロボットテクノロジー)のための統一規格の一つ

RTM メリット

ソフトウェアを RT コンポーネントと呼ばれるモジュール単位で管理することで、
ソフトウェアの部品を組み立て、システム全体を構築することができる。



RT の統一規格として、
世界を便利にするため、
RTM を推進している。

OpenRTM-aist

産業技術総合研究所(AIST)で開発されたオープンソースの RTM
C++, Physon, JAVA に対応
Eclipse という統合環境上に開発環境を構築

統合環境

RT システムエディタ(RT システム構築のための環境 [部品の組立])
RTC ビルダー (RT コンポーネント(RTC)を作成するための環境 [部品の作成])

RT システム構築の手順 (RT システムエディタの使い方)

- 起動
 - 1.1 Start naming service(RTC 管理ソフトウェア)の起動
 - 1.2 RT システムエディタ(システムの構築環境)の起動
 - 1.3 RTC(システム構築の部品)の起動
- システム構築
 - 2.1 新しいシステムエディタを開く (Open New System Editor)
 - 2.2 RTC をシステムエディタ上に配置
 - 2.3 RTC を接続
- アクティベート(アクティブな状態にする。スイッチ ON)
 - 3.1 All Activate のボタンを押す。→ システム全体が動き始める。



RTC(RT コンポーネント)

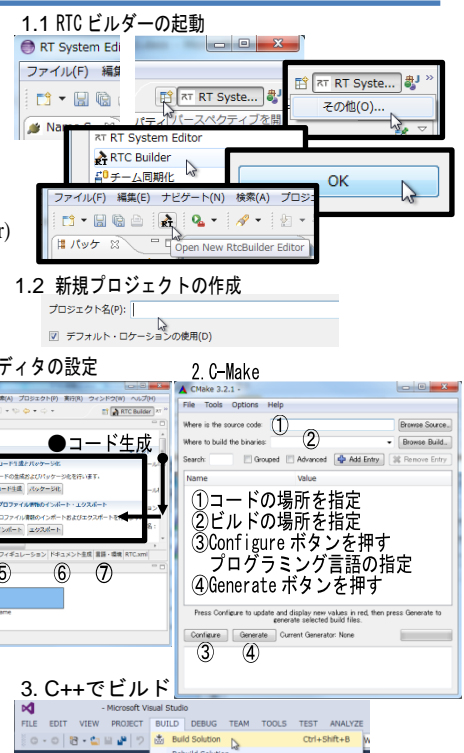
入力: ポート名 データ型 → Inactive(青色 OFF) ↔ Active(緑色 ON) → 出力: ポート名 データ型
Error(赤色) [内部状態]

RTC の作成 (RTC ビルダーの使い方)

- RTC ビルダー(RTC の基本構造の構築)
 - 1.1 RTC ビルダーを起動。(Eclipse. パースペクティブを開く・・・→その他→RTCBUILDER)
 - 1.2 新規プロジェクトの作成(プロジェクト名を決める)
 - 1.3 RTC プロファイルエディタの設定
 - ① 基本情報の設定(モジュール名, バージョン, バンダー名, モジュールカテゴリー)
 - ②～⑦ 詳細情報の設定
 - ② アクティビティ(onActivated→on, onDeactivated→on, onExecute→on 等)
 - ③ データポート(入力, 出力のポート数, ポート名, データ型を決める)
 - ④ サービスポート
 - ⑤ コンフィギュレーション(内部パラメータの設定)
 - ⑥ ドキュメント生成
 - ⑦ 言語・環境(使用する言語・環境の設定[C++, VB.NET, Python, Java, C#])
 - コード生成(①基本情報に戻り, コード生成ボタンを押す)
- CMake(プロジェクトファイル・Makefile 等の生成)
- 実装 (ロジックを作成し, ビルドする[実行プログラムの作成].)

RTC の再利用

OpenRTM-aist official website のプロジェクトページ,
NEDO RT コンポーネント, RT ミドルウェアコンテスト, 個人サイト。



充電式乾電池は、繰り返し使用でき便利です。RTM は RT のフレームワークを提供し、RTC の再利用を推進しています。



世界をもっと便利に

