



SysML Simulation Feature Guide

by SparxSystems Japan

Enterprise Architect 日本語版

SysML パラメトリック図のシミュレーション 機能ガイド

(2016/11/16 最終更新)



1. はじめに

このドキュメントでは、**Enterprise Architect** で利用可能な **SysML** のシミュレーション機能について、最低限の構成を構築するための方法を説明します。

このドキュメントは、**Enterprise Architect 13.0** ビルド 1307 を利用して記述しています。過去のバージョン・ビルドについては利用できない機能や操作が異なる点がありますのでご注意ください。

Enterprise Architect で利用可能な **SysML** のシミュレーション機能には、次の 2 種類があります。

- **Enterprise Architect** 独自のシミュレーション機能
JavaScript や VBScript で制約ブロックの挙動を定義し、そのブロックを利用してパラメトリック図を作成します。シミュレーションの際には、パラメトリックモデルの内容から JavaScript や VBScript を自動生成、結果を CSV やグラフで出力できます。バージョン 13.0 よりも前のバージョンでは、この機能のみを提供していました。
- **OpenModelica** と連携するシミュレーション機能
ブロック図・内部ブロック図・パラメトリック図を作成し、その内容から **OpenModelica** のソースコードを自動生成し、実行・結果表示までを自動で行います。多数の実績と定評のある **OpenModelica** が利用できることで、従来の機能では実現できなかった複雑な内容や、制約を満たす条件の解決などが可能になりました。

これらの機能は、どちらかを事前に選択する必要があります。選択した内容に応じて作成するモデルの内容は異なり、一方の機能のために作成したモデルを他方の機能のモデルとして転用・流用することはできません。

2. 利用条件

SysML のシミュレーション機能を利用するためには、以下の条件を満たす必要があります。

2.1. Enterprise Architect独自のシミュレーション機能の場合

以下のいずれかのエディションが必要です。

- Enterprise Architect Suite システムエンジニアリング版
- Enterprise Architect Suite アルティメット版
- Enterprise Architect プロフェッショナル版あるいはコーポレート版と別売りのアドイン「MDG Technology for SysML」の組み合わせ

2.2. OpenModelicaと連携する場合

バージョン 13.0 以降の、以下のいずれかのエディションが必要です。

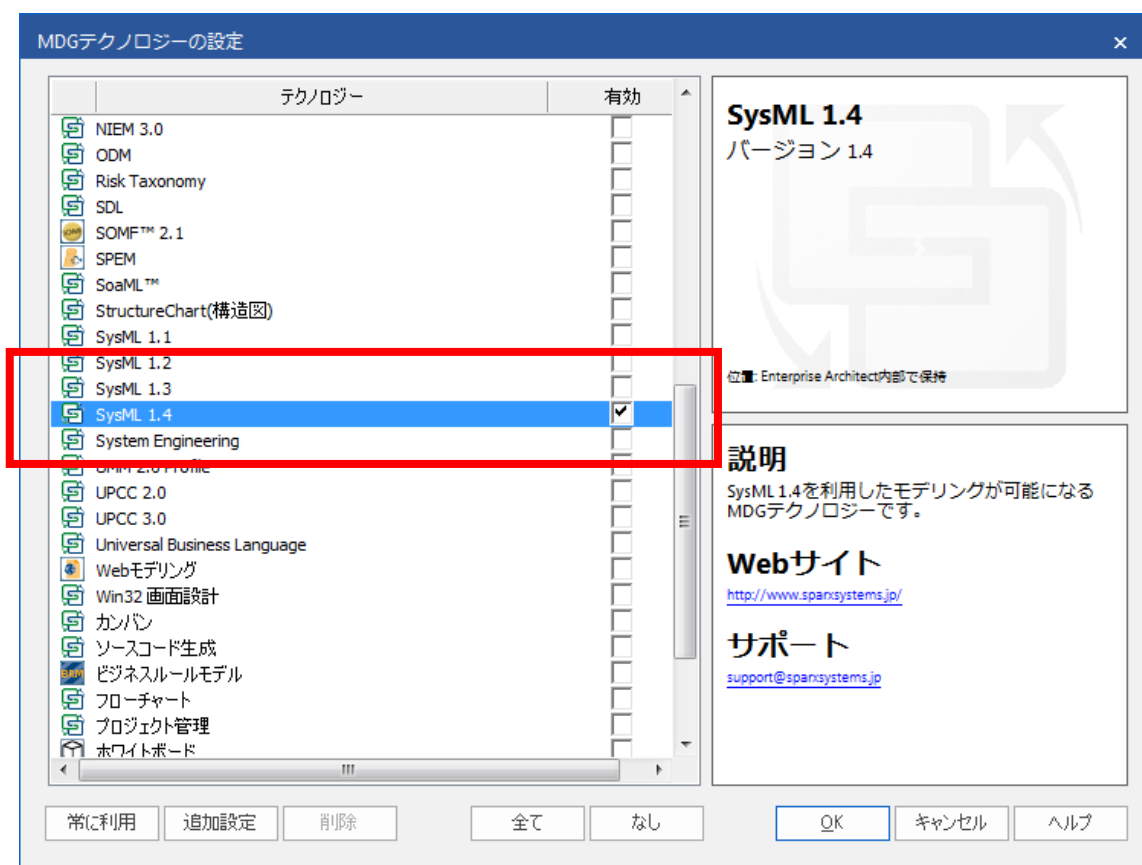
- Enterprise Architect Suite システムエンジニアリング版
- Enterprise Architect Suite アルティメット版

なお、このドキュメントでの説明用および動作確認用には、OpenModelica 1.9.6 を利用しています。OpenModelica は別途インストールしておいてください。

2.3. 共通の条件

このドキュメントで紹介するサンプルを利用するためには、「**SysML 1.4**」が有効である必要があります。Enterprise Architect を起動後、「アドイン・設定」リボン内の「MDG テクノロジー」パネルにある「設定」を実行して「MDG テクノロジーの設定」画面を表示し、「SysML 1.4」にチェックを入れて有効にし、「SysML 1.1」「SysML 1.2」「SysML 1.3」のチェックを外して下さい。

変更が完了したら、一度 Enterprise Architect を再起動して下さい。



3. Enterprise Architect独自の機能について

第3章では、Enterprise Architect 独自のパラメトリック図シミュレーション機能について説明します。

3.1. サンプルを利用して概要を確認する

Enterprise Architect では、SysML のパラメトリック図について実際に値を適用し、シミュレーションを行うことができます。この機能を利用することで、作成したモデルに実際の値を流し、その出力結果をモデル上で確認することができます。また、モデルという特性を生かすと、モデルの内容を変更して何度もシミュレーションを行い、その結果を基に最適な形を模索するという形式の設計作業も可能になります。

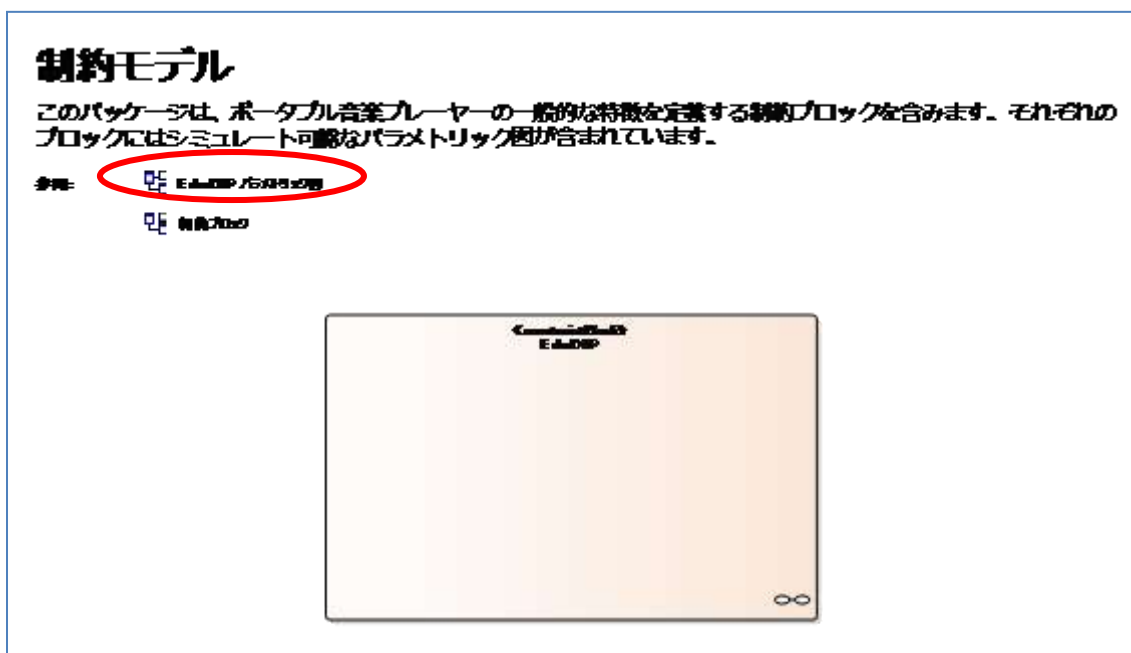
このシミュレーション機能を体験するには、サンプルを利用することをお勧めします。Enterprise Architect を起動し、「アドイン・拡張」リボン内の「アドインメニュー」パネルから「SysML」→「サンプルファイルを開く」を実行して下さい。

以下のようなサンプルのトップページが表示されます。このサンプルは、携帯型音楽プレーヤを題材にしたサンプルです。

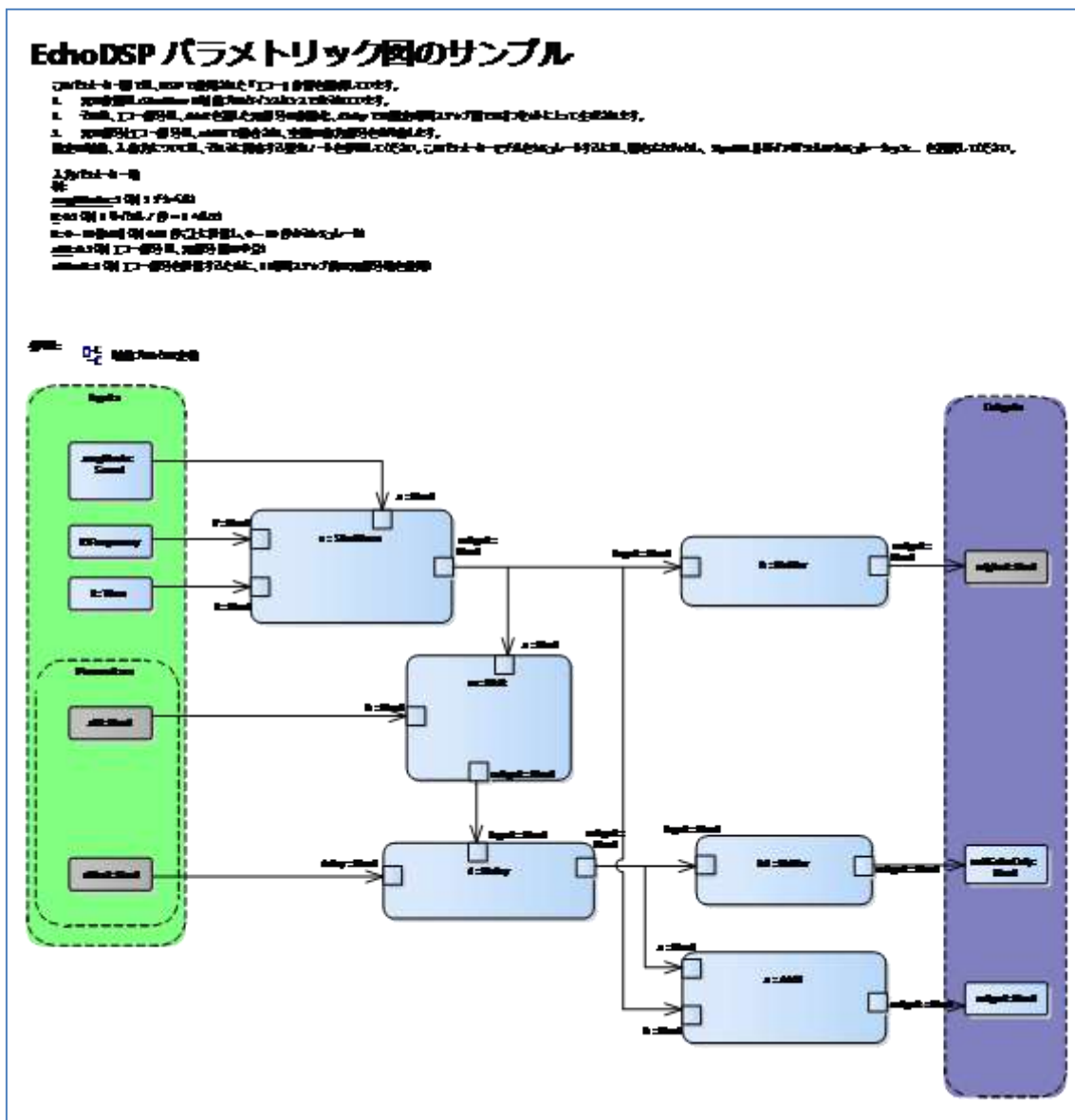
なお、このサンプルは、SysML1.4 で作成されています。



ここで、ダイアグラム内の「制約モデル」のリンクをダブルクリックして下さい。次のようなダイアグラムが表示されます。



ここでは、上記の赤丸の位置にある「EchoDSP パラメトリック図」のリンクをダブルクリックして下さい。次のようなパラメトリック図に移動します。



このダイアグラムの背景で右クリックし、「SysML」→「ダイアグラムのシミュレーション」を選択します。すると、次のような画面が表示されます。



この画面では、それぞれのパラメータを設定する必要がありますが、既定の状態を上記のように値が設定されています。

なお、入力パラメータの値は以下の通りです。

amplitude : 5

att : 0.5

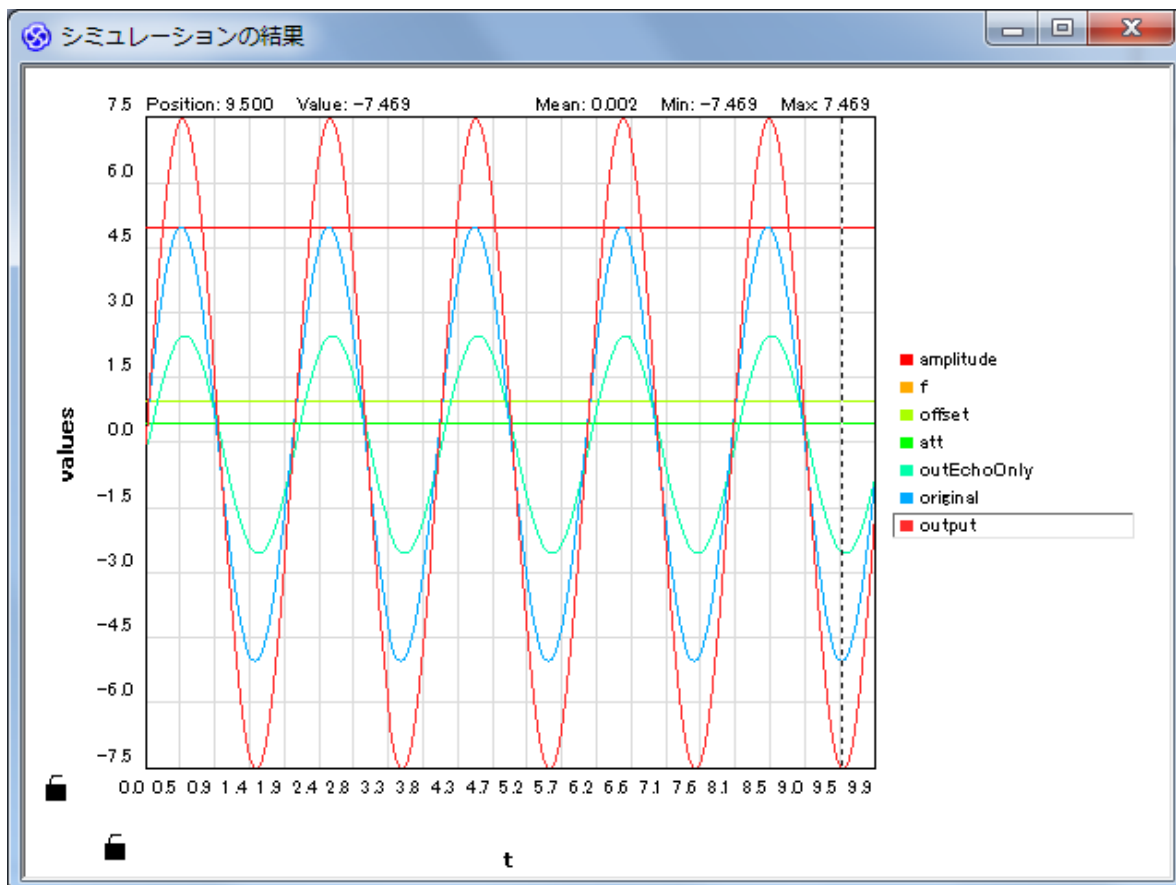
f : 0.5

offset : 1

t : 0 から 10 まで(増分 0.05)

「OK」ボタンを押すと、以下のような結果が表示されます。結果は CSV ファイルとして出力することも可能です。

(一部の環境では、この結果の画面が Enterprise Architect のウィンドウの裏に回ってしまう場合があります。)



このように、モデルの内容を元に計算を行い、結果を表示することができます。

3.2. シミュレーションに必要な情報

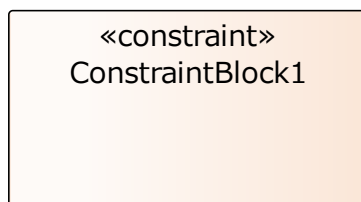
では、このようなシミュレーションを独自に定義し、行う方法について説明します。

このようなシミュレーションを行うのに必要な内容は、次の2つです。

- ・ 制約ブロック
- ・ パラメトリック図

3.2.1. 制約ブロックの作成

新規にブロック定義図を作成します。その後、ツールボックスに含まれる「制約ブロック」をダイアグラム内にドロップすることで、制約ブロックを作成することができます。



作成後、名前を変更して下さい。

この制約ブロックがどのような挙動を行うかを定義する方法には、次の 2 つの方法があります。

3.2.1.1. スクリプトで定義

Enterprise Architect では、以下のスクリプト言語を利用して、制約ブロックの挙動を定義することができます。

- JavaScript
- Jscript
- VBScript

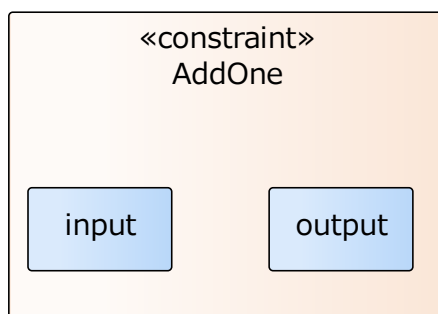
言語を一旦選択すると、定義内容を削除しない限り別の言語を選択することはできません。なお、利用する言語は、モデル内で 1 つのみに統一する必要があります。

スクリプトを定義するには、対象の制約ブロック要素を右クリックし、「SysML」→「要素のスクリプトの追加」→「(希望するスクリプト言語名)」を選択して下さい。Enterprise Architect のタブとしてエディタが起動し、内容を定義できます。

内容については自由に記述することができます。また必要に応じて Math ライブラリを利用することもできます。例えば、MathLib.IMath.cos のような関数です。(エディタ内で MathLib. と入力すると、インテリセンス(入力支援)機能が起動し、選択肢が表示されます。なお、この Enterprise Architect の Math ライブラリは Cephес の数学ライブラリとほぼ同じ関数を提供しています。個々の関数や定数については、この Cephес のライブラリの情報が参考になります。)

スクリプトにおける変数については、入力パラメータ・出力パラメータと同名の変数を利用することができます。例えば、以下のような制約ブロックの場合には、**input** および **output** という変数が利用できます。

(なお、言語仕様により、利用できない名前があります。プロパティ作成の際にはご注意ください。)



(上記は作成した制約ブロックの名前を **AddOne** に変更し、二つのパートプロパティを制約ブロックに配置し、**input** と **output** と名前を変更しました。)

一例として、今回は以下のような VBScript を定義しました。

A screenshot of a code editor window with a toolbar at the top. The code area contains a single line of VBScript: `1 output = input + 1`. The text is color-coded: '1' is blue, 'output' is black, '=' is black, 'input' is blue, and '+ 1' is black.

3.2.1.2. パラメトリック図で定義

制約ブロックの挙動を定義する場合に、パラメトリック図で定義することも可能です。この場合には、他の制約ブロックを利用することができます。つまり、制約ブロックを「入れ子」にして、挙動を定義することができます。

詳細手順は後ほど説明します。

3.2.2. パラメトリック図の作成

3.2.2.1. プロパティ要素の作成

パラメトリック図では、まずツールボックスにあるパートプロパティ要素を配置して、シミュレーション用のパラメータを定義します。

この例では、param を入力パラメータ・result を出力パラメータとしています。



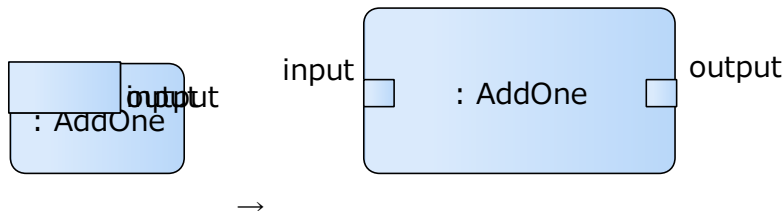
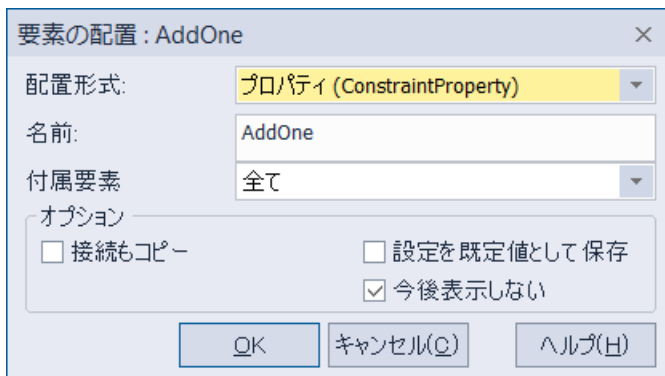
3.2.2.2. 定義済みの制約ブロックの配置

パラメトリック図で利用する、他の定義済みの制約ブロック要素をプロジェクトブラウザから配置します。配置しようとする、次のような「要素の配置」ダイアログが表示されます。(設定によっては表示されない場合があります。表示されない場合には、Ctrl キーを押しながら制約ブロック要素をドロップして下さい。)

このダイアログでは、以下のように

- ・ 「配置形式」では「プロパティ(ConstraintProperty)」
- ・ 「付属要素」では「全て」

を選択して下さい。OK ボタンを押すと、制約ブロック要素のインスタンスである制約プロパティ要素が自動的に生成され、配置されます。プロパティ要素が重なっている場合には、それぞれの要素の大きさや配置を調整して下さい。



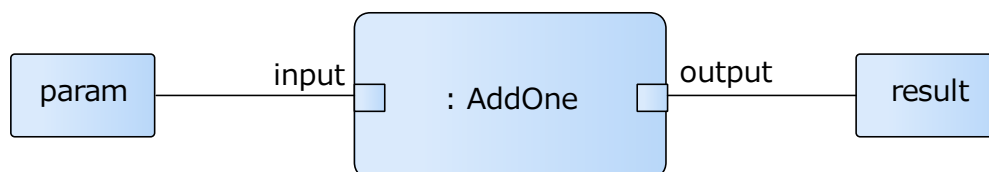
配置直後の例

レイアウトを調整した例

3.2.2.3. プロパティの接続

その後、作成したプロパティの入出力がつながるように、クイックリンク機能で「コネクタ」を選択してプロパティ要素間を接続します。

この例では、以下のように接続しました。なお、制約プロパティ要素(下の図の例では「:AddOne」と記載されている要素)にコネクタを接続しないように注意して下さい。



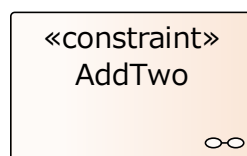
このパラメトリック図のダイアグラムでシミュレーション可能です。3.3章のシミュレーションの実行に進んでください。

3.2.3. パラメトリック図で制約ブロックの挙動を定義

制約ブロックに対する挙動をパラメトリック図で定義する場合には、制約ブロックにパラメトリック図を追加します。その後、プロパティや定義済みの制約ブロックを配置し、コネクタで接続します。

3.2.3.1. パラメトリック図の追加

対象の制約ブロック要素を右クリックし、「子ダイアグラムの追加」→「子ダイアグラムの作成」を実行して下さい。要素の子としてパラメトリック図が追加され、要素には∞のマークが表示されます。このマークが表示されている状態で制約ブロック要素をダブルクリックするとパラメトリック図に移動できます。



3.2.1.1 のスクリプトで定義の場合と違い、ここではパートプロパティを配置する必要は

ありません。

3.2.3.2. プロパティ要素の作成

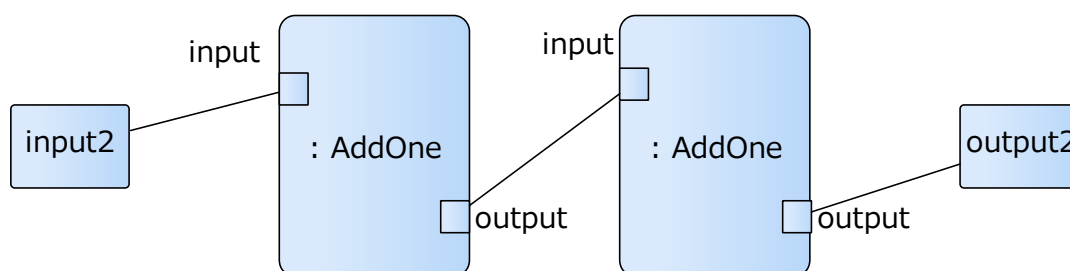
制約ブロック要素をダブルクリックして表示されるパラメトリック図に、ツールボックスにあるパートプロパティ要素を配置して、制約ブロックの入出力のパラメータを定義します。



この例では、**input2** を入力パラメータ・**output2** を出力パラメータとしています。この定義ではダイアグラムフレームは不要ですので、ダイアグラムフレームが表示されている場合は、ダイアグラムフレームを選択して **Delete** キーを押して削除するか、ダイアグラムの背景で右クリックし「ダイアグラムフレームの非表示」を選択して下さい。

3.2.3.3. 定義済みの制約ブロックの配置とプロパティの接続

3.2.2.2 と同じように、定義済みの制約ブロックを配置し、3.2.2.3 と同じようにプロパティを接続してください。今回は、3.2.1.1 で作成した **AddOne** 定義を2つ使い、下記のようにしました。



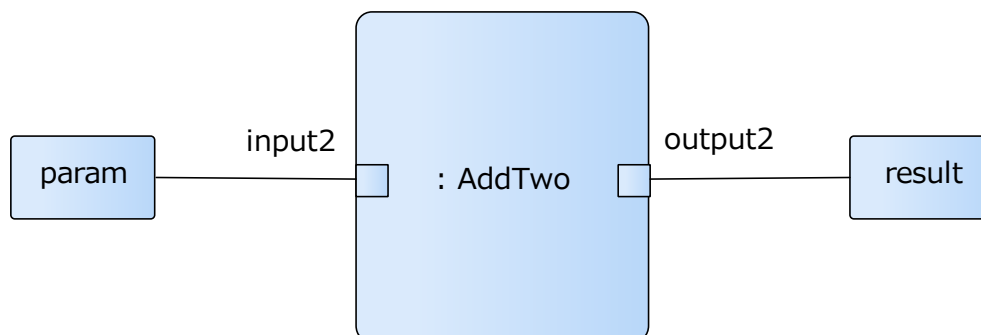
以上で定義は完了です。

3.2.3.4. シミュレーション用パラメトリック図作成

パッケージの新規パラメトリック図を作成し、3.2.2.2 などと同じようにプロジェクトブラウザから、パラメトリック図で挙動が定義された制約ブロックを配置します。この際、付属要素は「全て」ではなく「要素を選択」を選び、プロパティ要素にチェックをいれてください。



そして、シミュレーション用のパラメータを加え・コネクタで接続し、以下のようにします。



このパラメトリック図のダイアグラムでシミュレーション可能です。

3.3. シミュレーションの実行

定義したパラメトリック図の挙動をシミュレーションする場合には、パラメトリック図の背景で右クリックし、「SysML」→「ダイアグラムのシミュレーション」を選択します。すると、3.1章で説明した「シミュレーションの設定」ダイアログが表示されます。

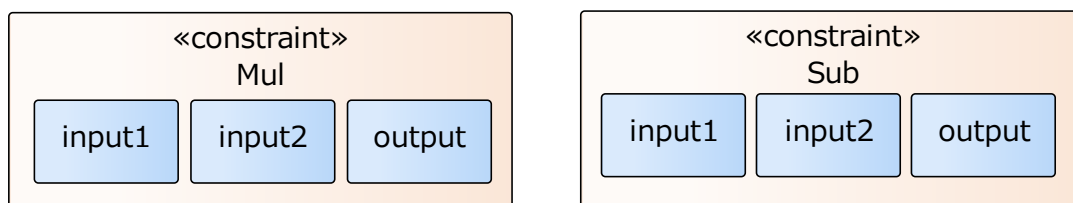
なお、この画面内の一覧にプロパティが表示されない場合や、シミュレーションの実行の際にエラーが表示される場合の多くは、プロパティの名前の付け方に問題があります。

また、プロパティの名前によっては、スクリプト言語での処理の都合で `element_000` のような接頭辞が追加される場合があります。この場合でも、シミュレーション結果には影響しません。

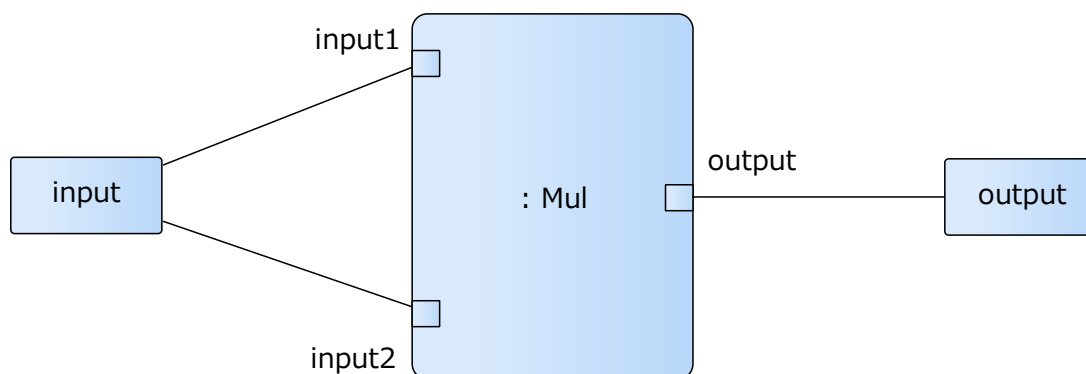
3.4. サンプル

最後に、実際に作成した例をご紹介します。複数の制約ブロックを組み合わせることで $y=ax^2-b$ を計算する制約ブロックを作成します。

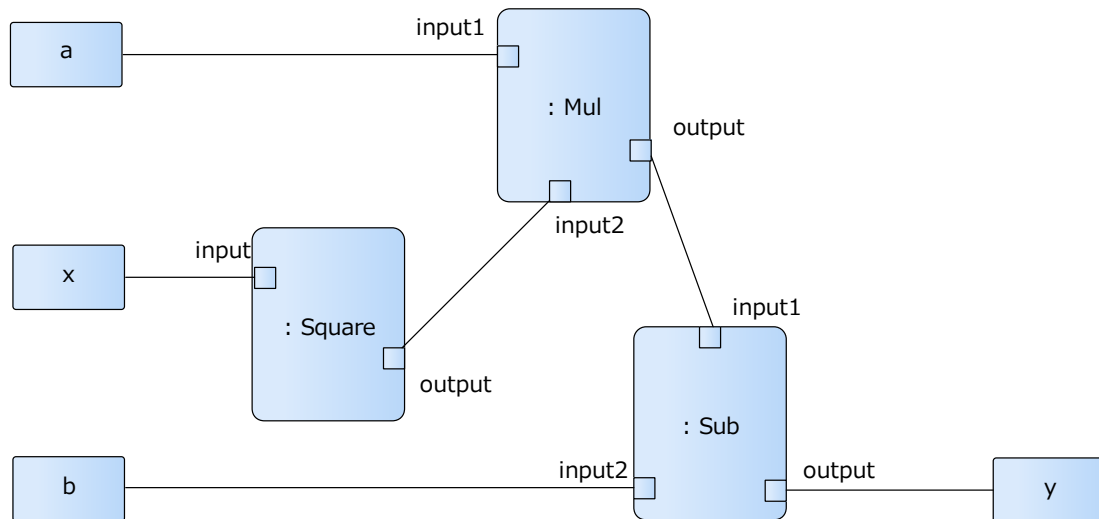
まず、基本的な計算の単位として、掛け算と引き算を実行する制約ブロックを作成します。これらの基本的な計算については、スクリプトで処理を定義します。



次に、掛け算の制約ブロックを利用して、2乗を計算する制約ブロックを作成してみましょう。以下のようなパラメトリック図になります。

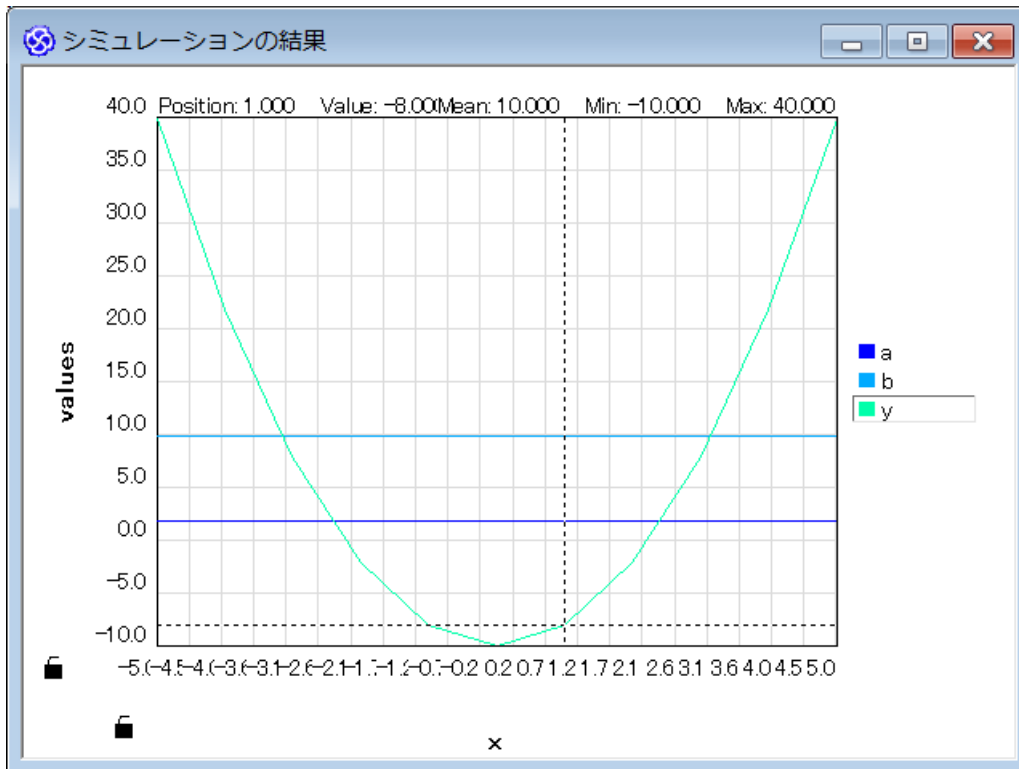


あとは、これらの制約ブロックを組み合わせて、当初の目的の関数を作成します。これは、以下ようになります。



そして結果は、次の通りです。

(a=2, b=10, x=-5~5)



このように、制約ブロックという「部品」を作成し、部品同士を組み合わせて別の部品

を作ることができます。ぜひお試しください。

4. OpenModelicaとの連携について

次に、Enterprise Architect 13.0 から利用できるようになった OpenModelica との連携について説明します。

4.1. サンプルを利用して概要を確認する

まず、サンプルを利用しながら、OpenModelica の設定や基本的な流れを確認します。

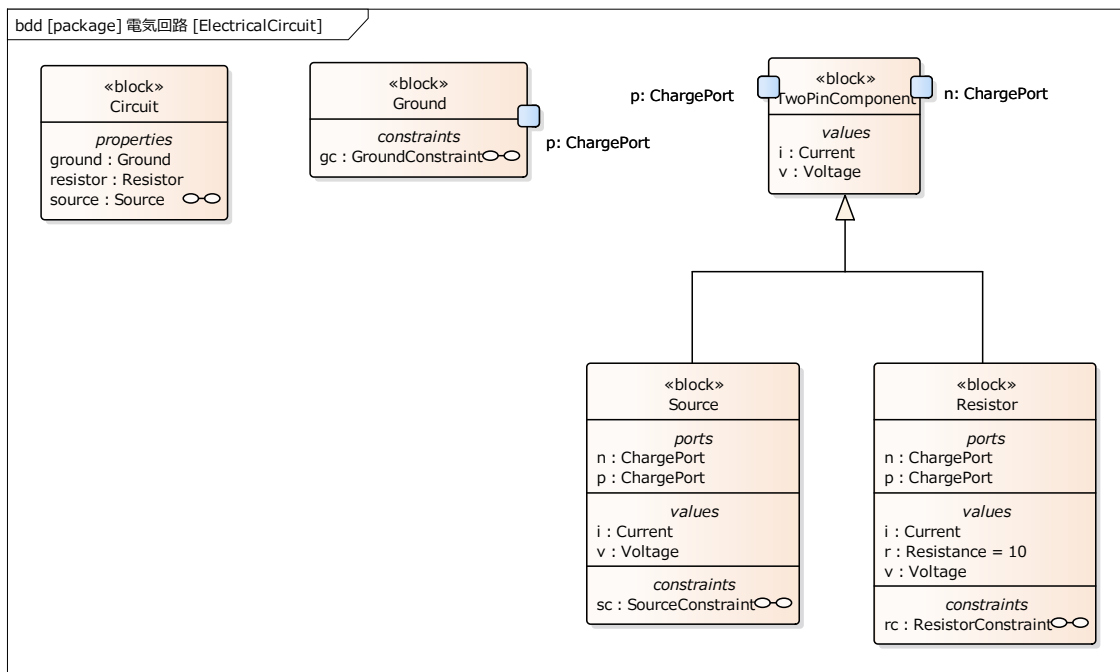
サンプルファイルは、Enterprise Architect のインストールディレクトリにある「SysMLSim_Sample.eap」になります。インストールディレクトリは Windows のアクセス権の関係でファイルを開くことができませんので、このファイルをデスクトップなどファイルを編集可能なディレクトリにコピーし、Enterprise Architect で開いてください。

このファイルには 3 つのサンプル(「電気回路」「バネ質量系ダンパー」「貯水タンク」)が含まれています。それぞれの内容の詳細については、Enterprise Architect のヘルプに概要を記載しています。

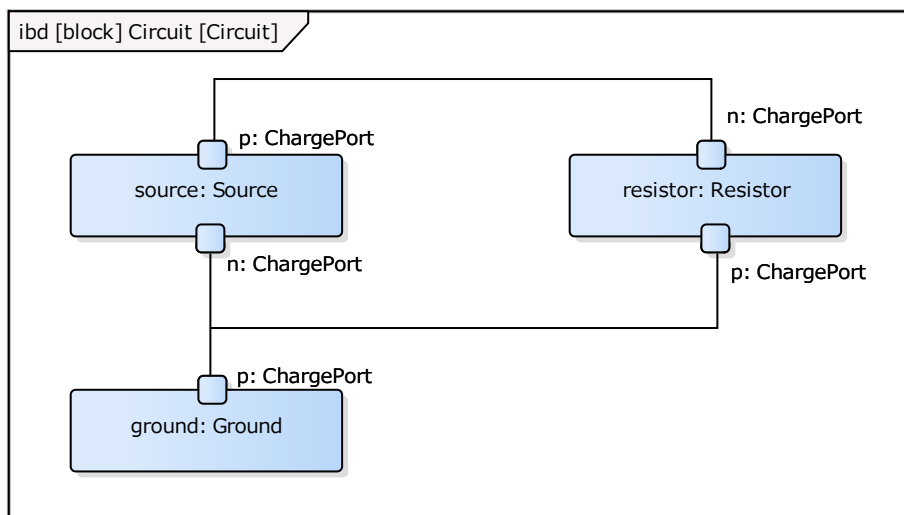
今回は、「電気回路」の例を利用して説明します。まずは、サンプルに含まれるそれぞれのモデルを確認します。

OpenModelica と連携するシミュレーション機能では、パラメトリック図以外に、ブロック図・内部ブロック図が必要です。そして、ブロック要素に対して OpenModelica のソースコードを自動生成し、実行することになります。

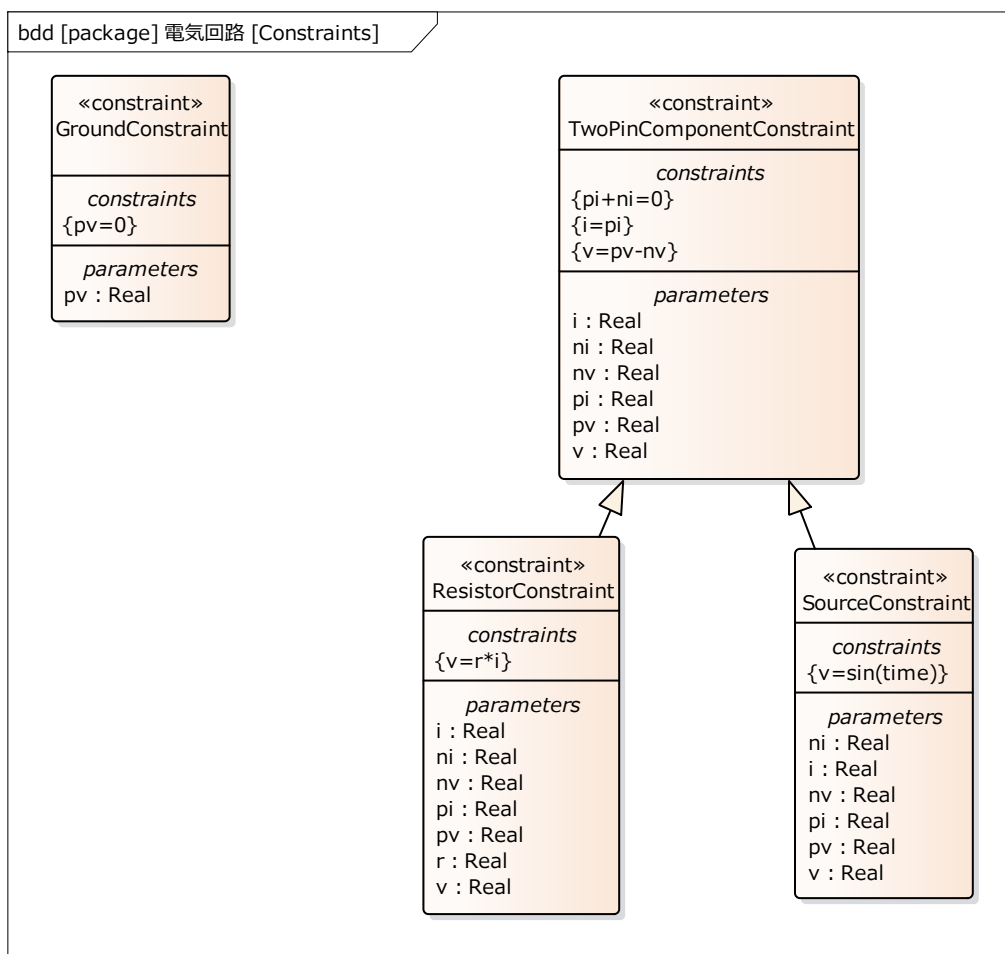
ElectricalCircuit のブロック定義図を開くと、以下のようなブロックが定義されています。



ここで定義されるブロックのほとんどには内部ブロック図が定義されていますので、ブロック要素をダブルクリックして参照できます。例えば、**Circuit** ブロックをダブルクリックすると、以下のような内部ブロック図が定義されていることがわかります。



また、ブロック要素には制約も定義されています。例えば、**Source** ブロックには「sc : SourceConstraint」として成約と結びつけられています。この制約の定義は、Constraints という名前のブロック定義図で定義されています。



制約ブロックの **constraints** 区画には、制約の内容が記述されています。第 3 章で説明した Enterprise Architect 独自の機能では、この制約の内容を JavaScript など記述する必要がありましたが、OpenModelica との連携の場合は、パラメータを利用して等式で記述します。

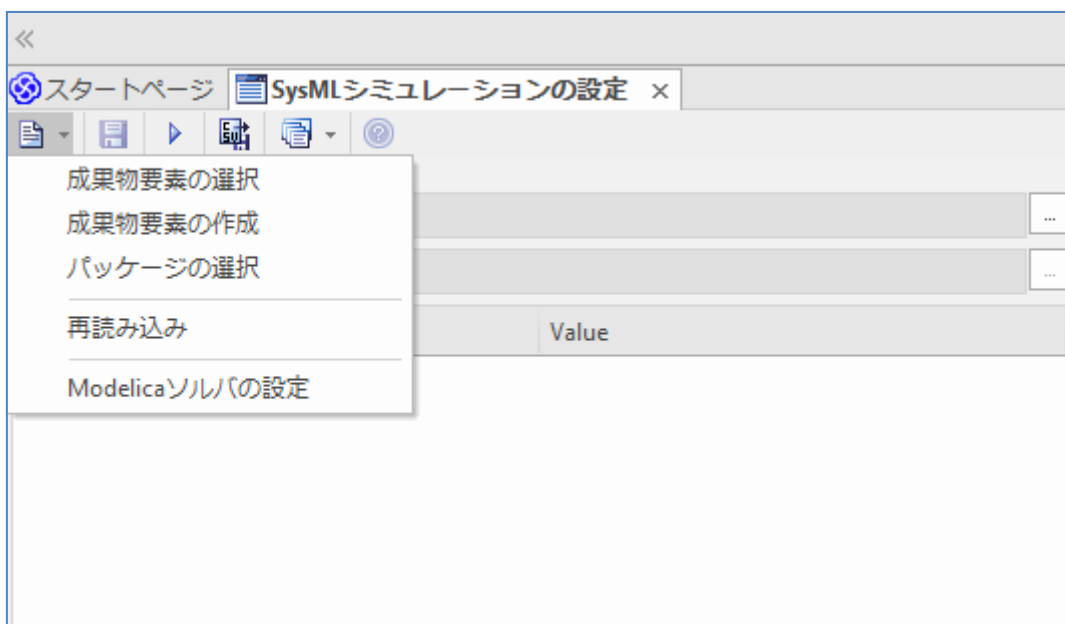
OpenModelica を利用したシミュレーションでは、このような SysML のモデルに加えて、シミュレーションのための要素が必要です。ツールボックス内の「拡張要素」グループに含まれる「SysML シミュレーションの設定」要素がその要素です。今回のサンプルでは、この要素も既に用意されています。

では、具体的にシミュレーションを実行する手順について説明します。

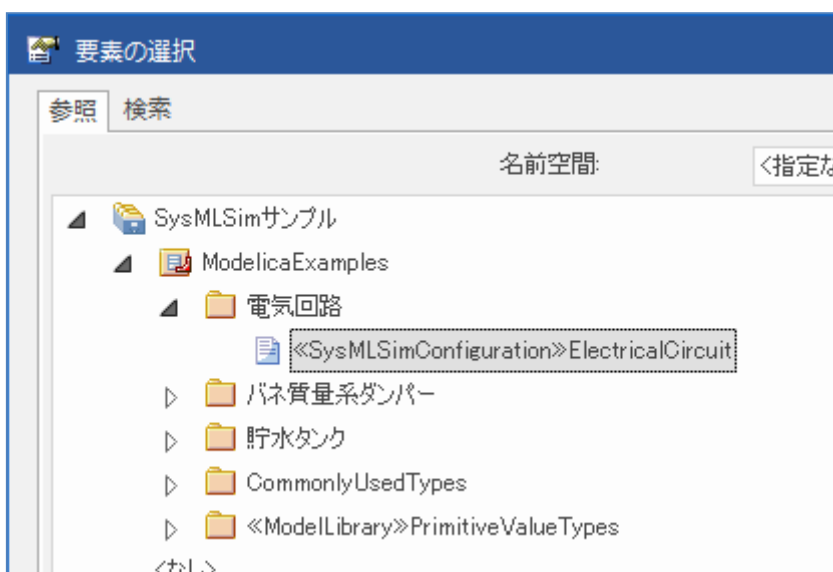
まず、「シミュレーション」リボン内の「SysMLSim」パネルにある「設定」ボタンを押してください。「SysML シミュレーションの設定」タブが表示されます。



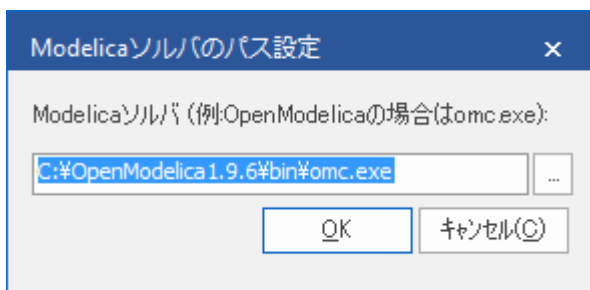
今回は定義済みの設定を読み込みますので、このタブ内にあるツールバーの左端のボタンを押すと表示されるメニューから「成果物要素の選択」を実行します。



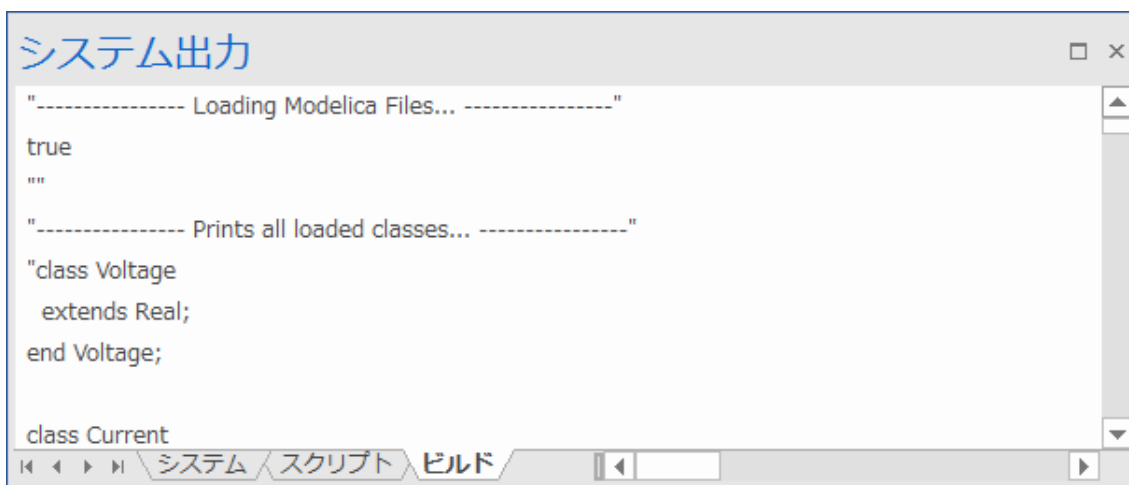
「電気回路」パッケージ内にある要素を選択します。



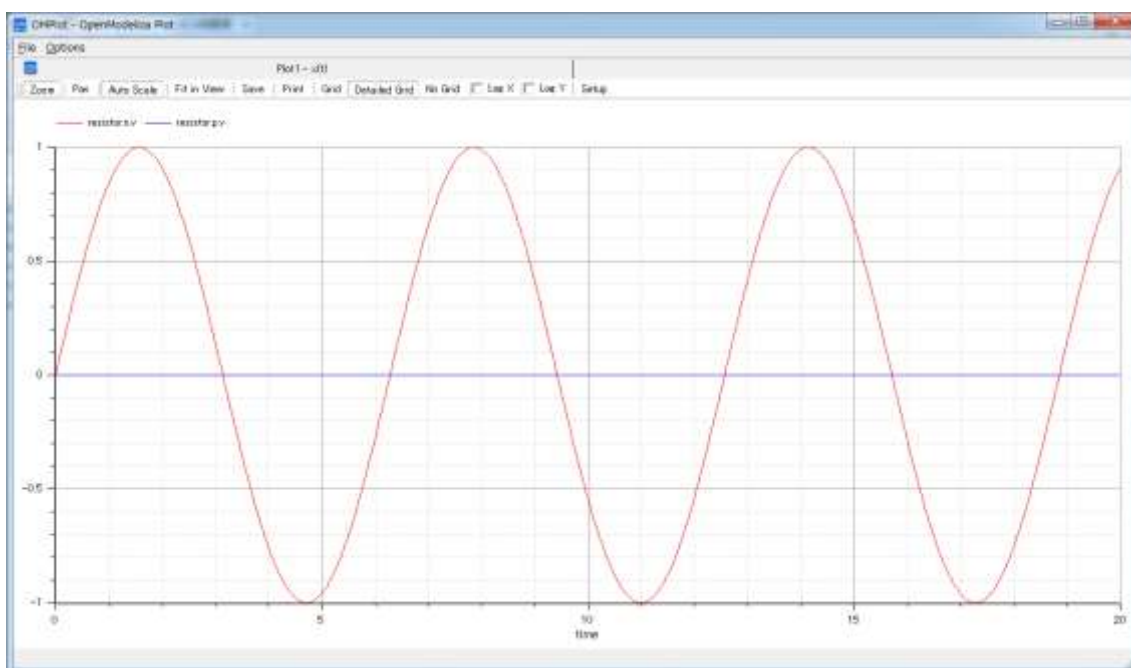
初回に限り、OpenModelica の設定が必要です。ツールバーで先ほどと同じボタンで表示されるメニューから「Modelica ソルバの設定」を実行し、インストール済みの Modelica ソルバ(omc.exe)の位置を指定する必要があります。



今回のサンプルではそれ以外の設定は全て完了していますので、「SysML シミュレーションの設定」タブの右側にある「解析」ボタンを押すと、内部で OpenModelica のソースコードを生成し、実行した結果を表示します。その際に、「出力サブウィンドウ」が表示され、実行状況などの詳細が表示されます。自分で作成したモデルで問題がある場合には、この内容が参考になります。



このサンプルでの実行結果として、以下のようなグラフが自動的に表示されます。



この **OpenModelica** を利用したシミュレーションでは、パラメトリックモデルに与えるデータを「データセット」として複数定義し、同じモデルに対して異なる条件でシミュレーションを実行することができます。第3章で説明した、**Enterprise Architect** 独自の機能よりも高機能であり、詳細なシミュレーションが可能です。
(ただし、その分設定も詳細です。)

他の 2 つのサンプルとヘルプの説明もご覧になると、概要が把握しやすいです。ぜひご覧ください。

○ 改版履歴

2009/09/03 初版

2010/04/16 Enterprise Architect 8.0 のリリースに伴い、内容を更新。

2011/05/18 Enterprise Architect 9.0 のリリースに伴い、内容を更新。

2013/07/18 SysML シミュレーション機能の利用条件について、分かりやすく記述。また、内容について最新の情報に更新。

2014/04/22 Enterprise Architect 11.0 のリリースに伴い、内容を更新。

2014/06/24 4 章の内容を更新。

2014/06/25 細かい表現の修正。

2016/03/17 サンプルファイルの内容の変更に伴い、内容を変更。その他、Enterprise Architect12.1 の画面・画像に差し替え。

2016/10/07 Enterprise Architect 13.0 のリリースに伴い、内容を大幅に更新。既存の機能の説明を第 3 章としてまとめ、第 4 章を追加。

2016/11/16 OpenModelica との連携機能が利用可能なエディションに誤りがあった点を修正。