

Stateflow® チュートリアル

MathWorks Japan
アプリケーションエンジニアリング部

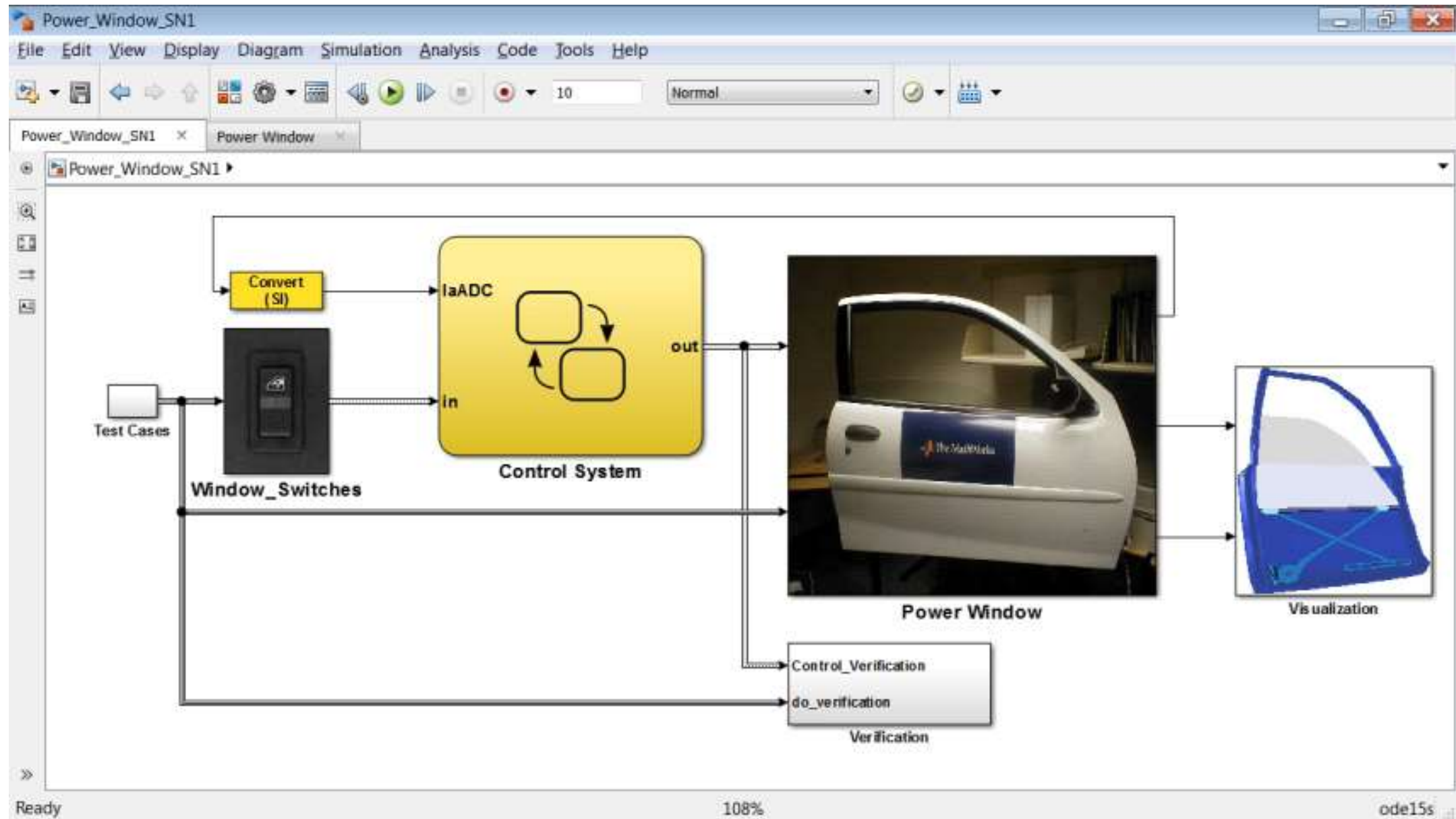
はじめに

- Stateflowの機能、基本操作について、実習を交えながら紹介します。



Let's try and feel Stateflow !

What is Stateflow?



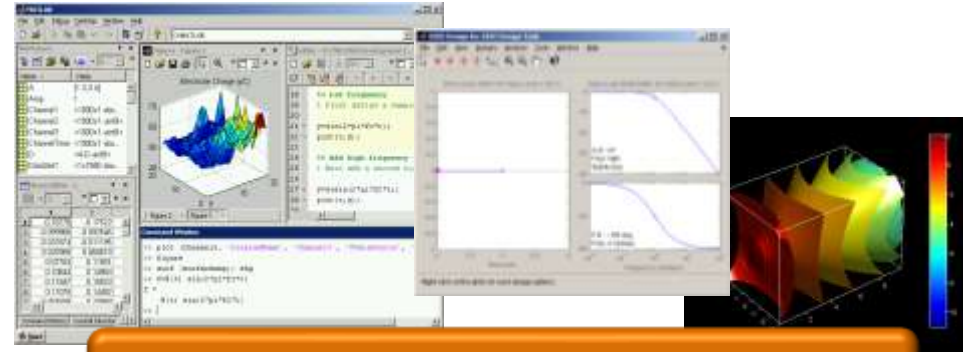
目次

- Stateflow 紹介
 - Stateflowとは
 - Stateflow基本環境
- Stateflow 演習
 - 例題1 フローチャート
 - 例題2 状態遷移図① 基本
 - 例題3 状態遷移図② 時相論理
 - 例題4 状態遷移図③ 階層化
 - 例題5 状態遷移図④ 並列状態
- 補足

MATLAB®とSimulink/Stateflowの関係は？

MATLAB

- 容易なデータ操作
- 簡潔なプログラミング言語
- 豊富な数学関数・ファイルI/O
- 2次元/3次元可視化機能



技術計算環境

Simulink

- ブロック線図モデリング
 - 豊富なブロックライブラリ
- ⇒MATLAB上で動作



モデルベースデザイン 環境

Stateflow®

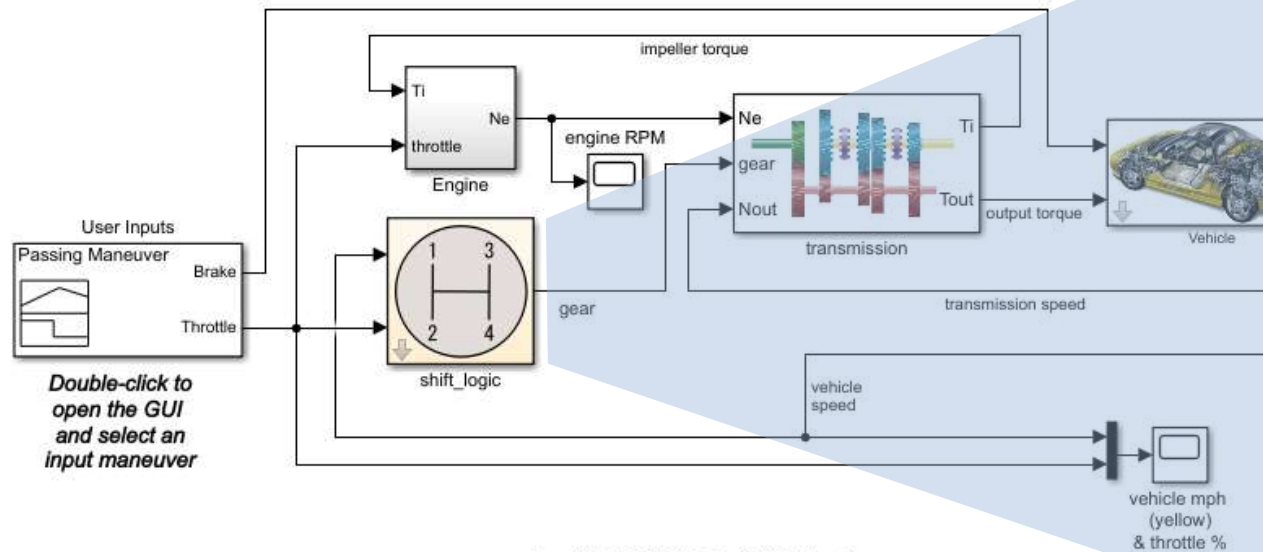
- フローチャート・状態遷移図を用いたモデリング環境
- ⇒Simulink上で動作



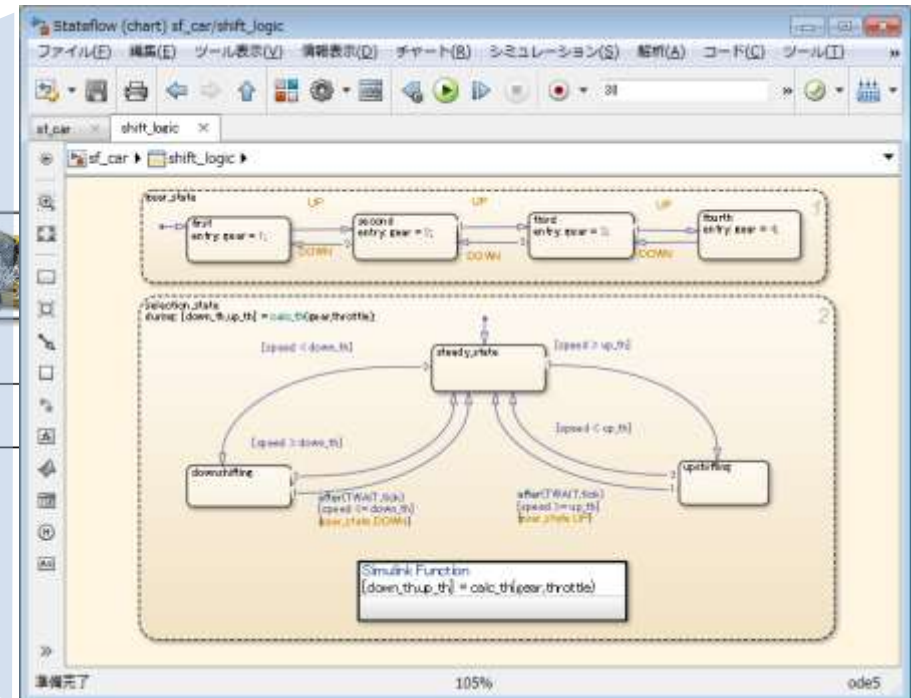
Stateflowとは

状態遷移図・状態遷移表・フローチャート・真理値表機能を提供

- Simulinkとシームレスに連携（Simulinkが必要）
- 階層化、並列化、関数化、時相論理に対応
- コード生成、モデル検証等各種オプション機能に対応



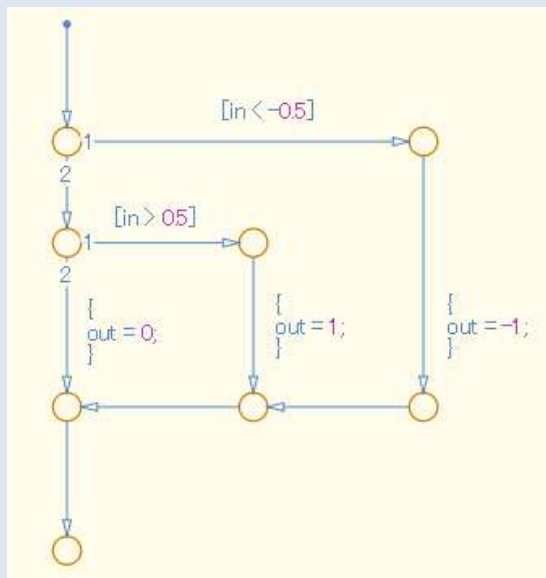
Copyright 1997-2016 The MathWorks, Inc.



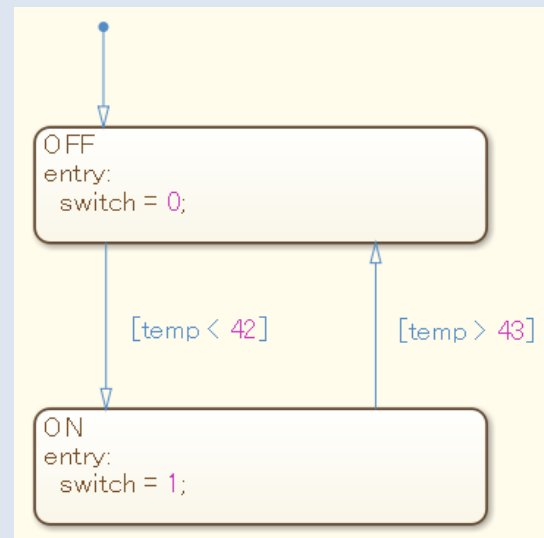
Stateflowによる色々なロジック表現

- ◆ 設計に応じて様々なデザインスタイルを活用できます

フローチャート



状態遷移図



状態遷移表

Block: untitled/State Transition Table*

ファイル(F) 編集(E) 表(T) ヘルプ(H)

ステート	遷移 (条件/アクション/遷移先ステート)
state1	if [in > 1] else-if (1) [in > 2]
state1	en: out = 1;
state1_1	en, du: cnt = 0;
state1_1	[flag]
state1_2	en: cnt = cnt + 1;
state2	[in < -1]
state3	[after(5, sec)]
state3	en: out = 3;
state3	state1
state3	state1
state3	state3
state3	% IGNORE %

真理値表

条件テーブル

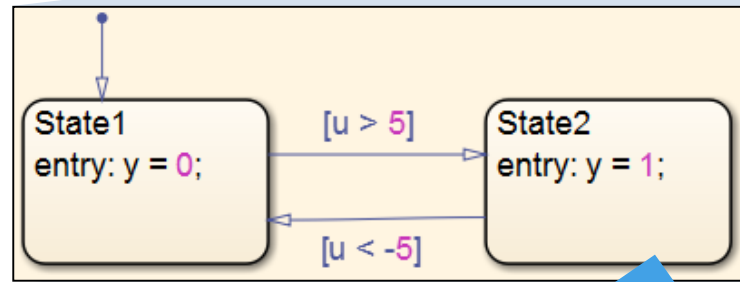
	説明	条件	D1	D2
1	Hot	t > T_thresh	T	T
2	Dry	h < H_thresh	T	-
	アクション: アクション テーブルから行を指定		CoolOn, HumidOn	CoolOn

アクション テーブル

#	説明	アクション
1	Turn On Cooling (This implicitly reduces humidity)	CoolOn: cooler = 1; heater = 0; humidifier = 0;
2	Turn On Heater (This implicitly reduces humidity)	HeatOn: heater = 1; cooler = 0; humidifier = 0;
3	Turn On Humidifier	HumidOn: humidifier = 1;

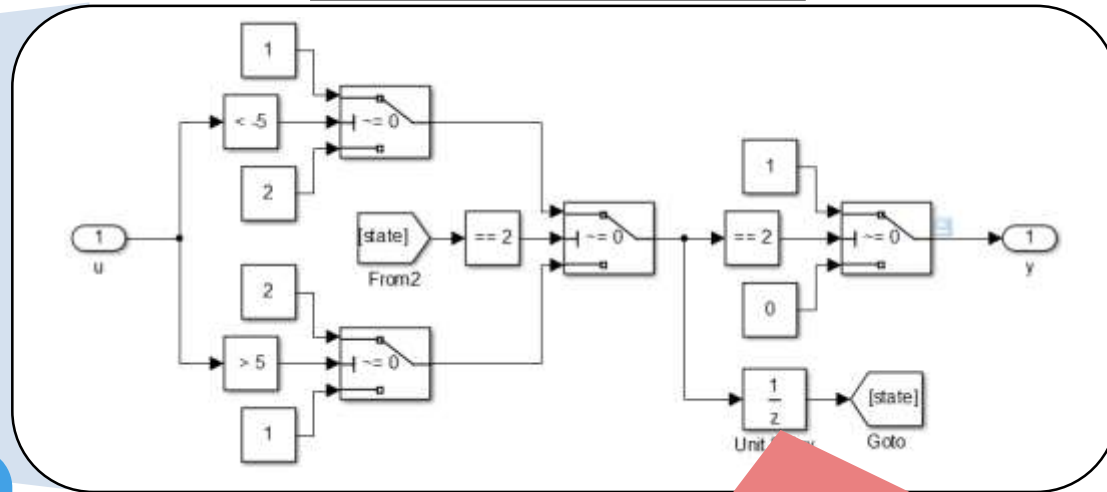
Stateflowの利点

Stateflow チャート

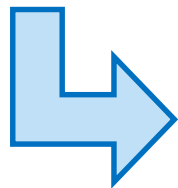


- ◆ ステート(状態)を持っている
- ◆ ステートや遷移に
処理内容を記述可能

Simulink ブロック



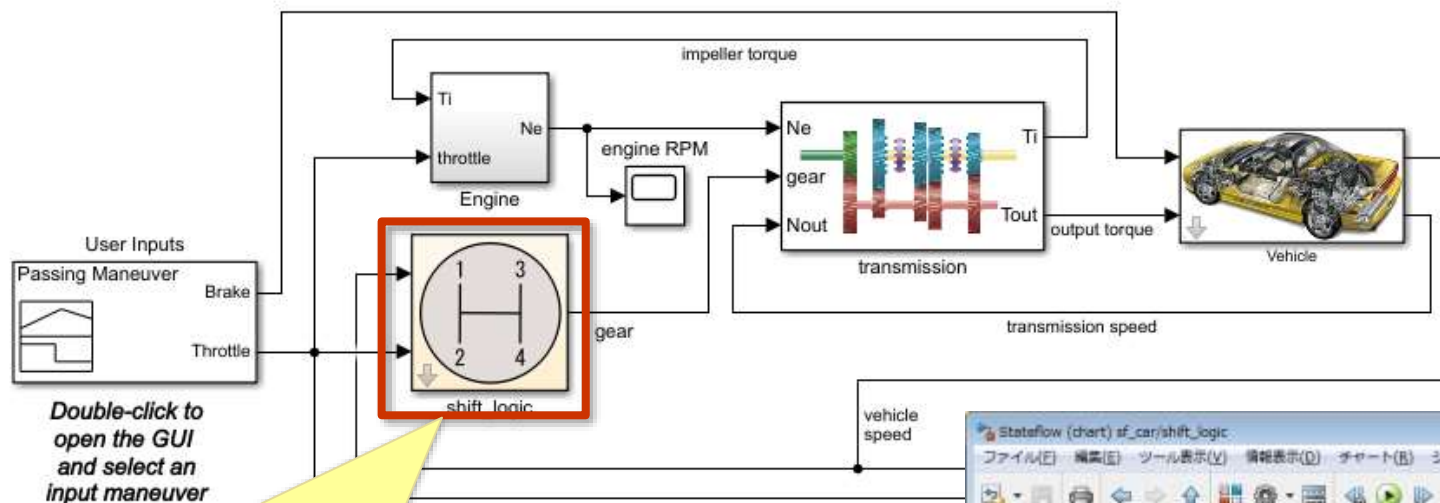
- ◆ 前回の状態を保持する設計必要
- ◆ Switchブロックなどを用いて
処理内容切り替える必要あり



Stateflow

- 状態遷移や条件分岐処理の表現が得意

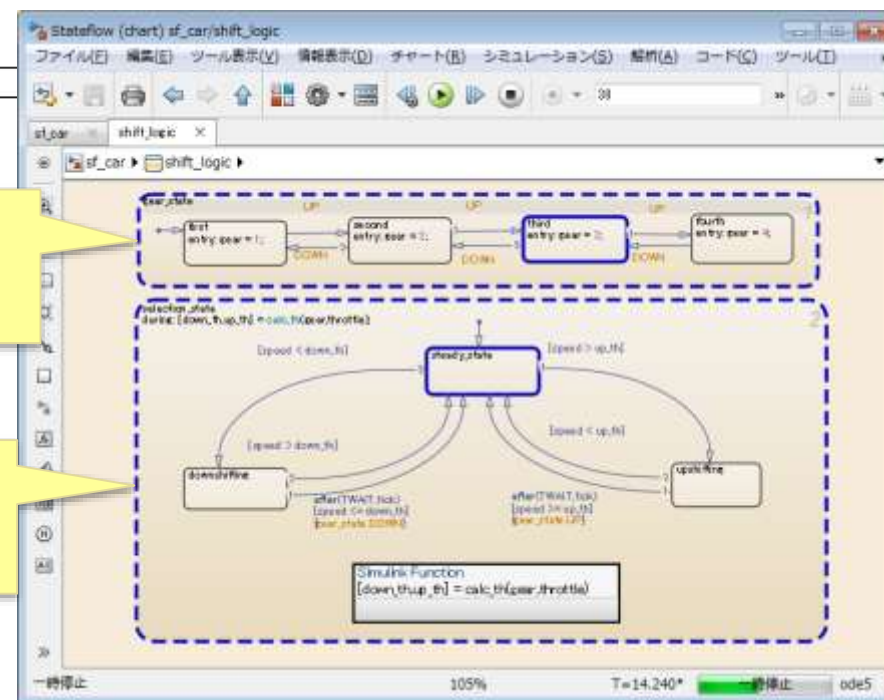
Stateflowデモ紹介： 4速AT車の変速制御



変速制御ロジックをStateflow
チャートブロックで表現

ギヤの状態遷移
(1速、2速、等)

シフトチェンジの判定処理
(シフトアップ・ダウン)

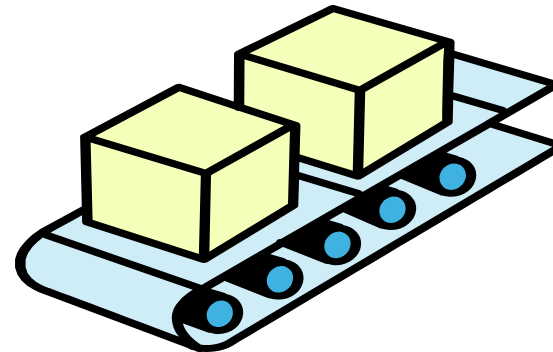
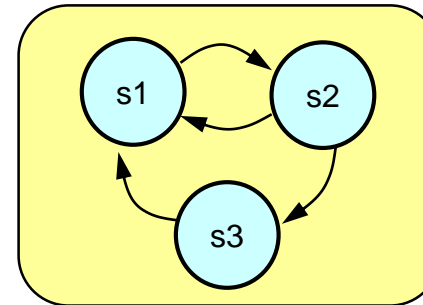


>> sf_car

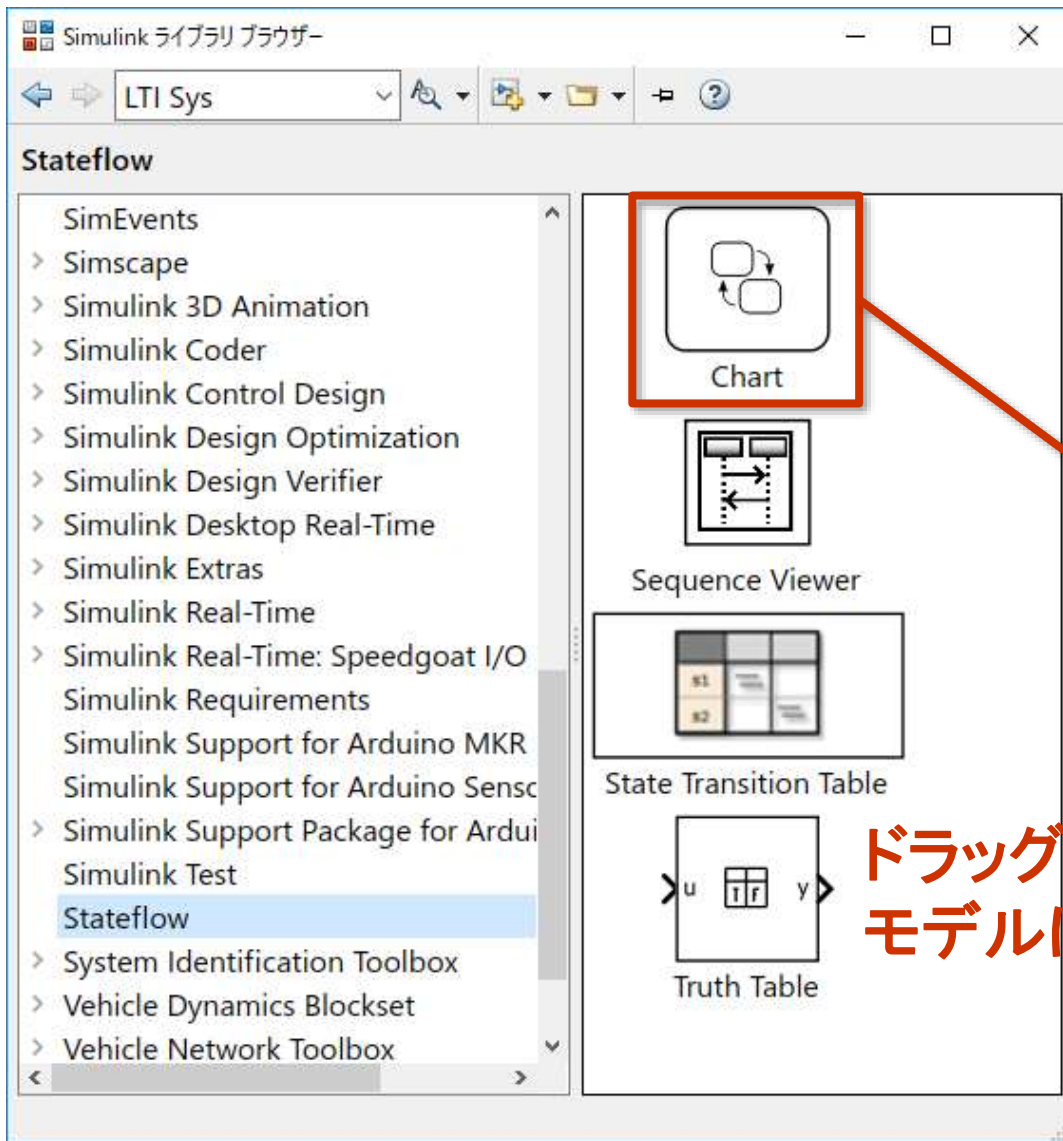
Stateflowの特長

下記のロジック記述やモデリングが得意です

- 複数の動作モードを持つ制御やプラント
- シーケンス制御、スケジューラ
- 監視・自動診断処理
- タイマ・イベントベース処理
- HMI（画面遷移）ロジック



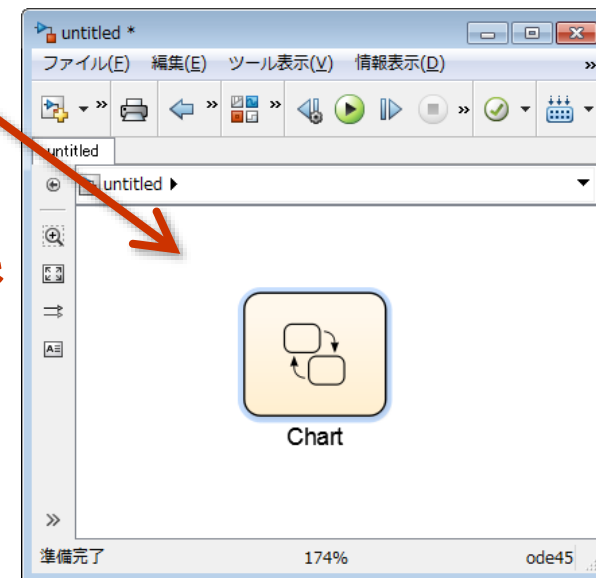
Stateflow ブロック



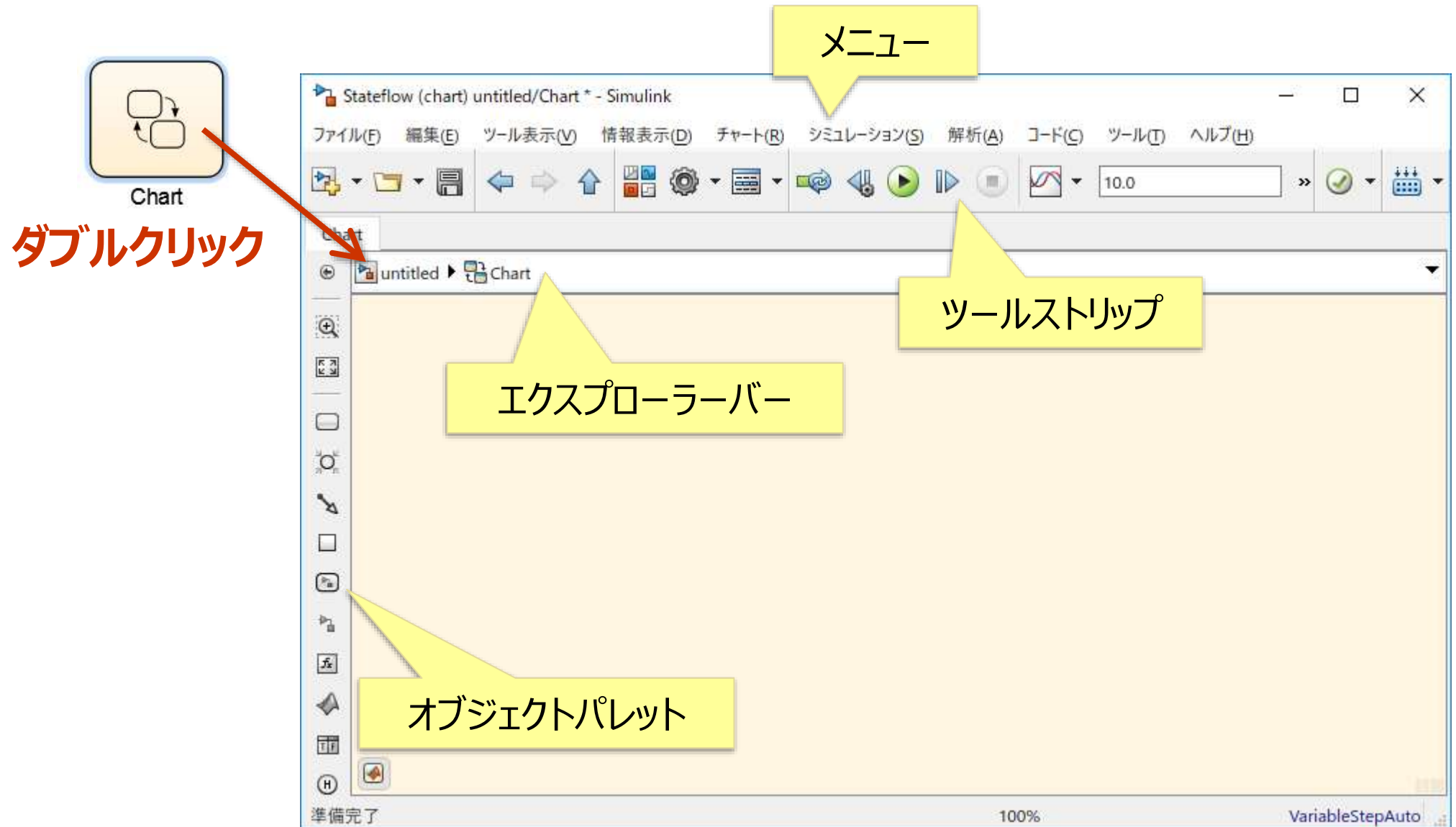
上から順に

- 状態遷移図
- シーケンス ビューア
- 状態遷移表
- 真理値表

ドラッグ&ドロップで
モデルに配置



Stateflow エディタ



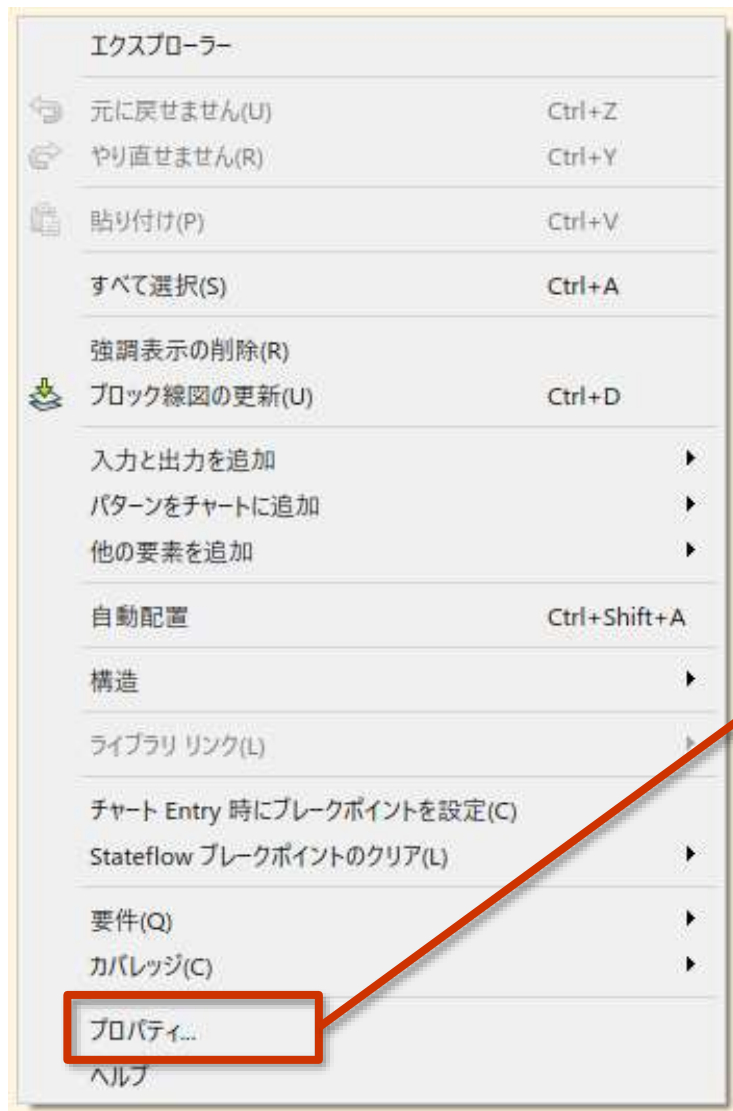
Stateflow エディタ オブジェクトパレット

上から順に

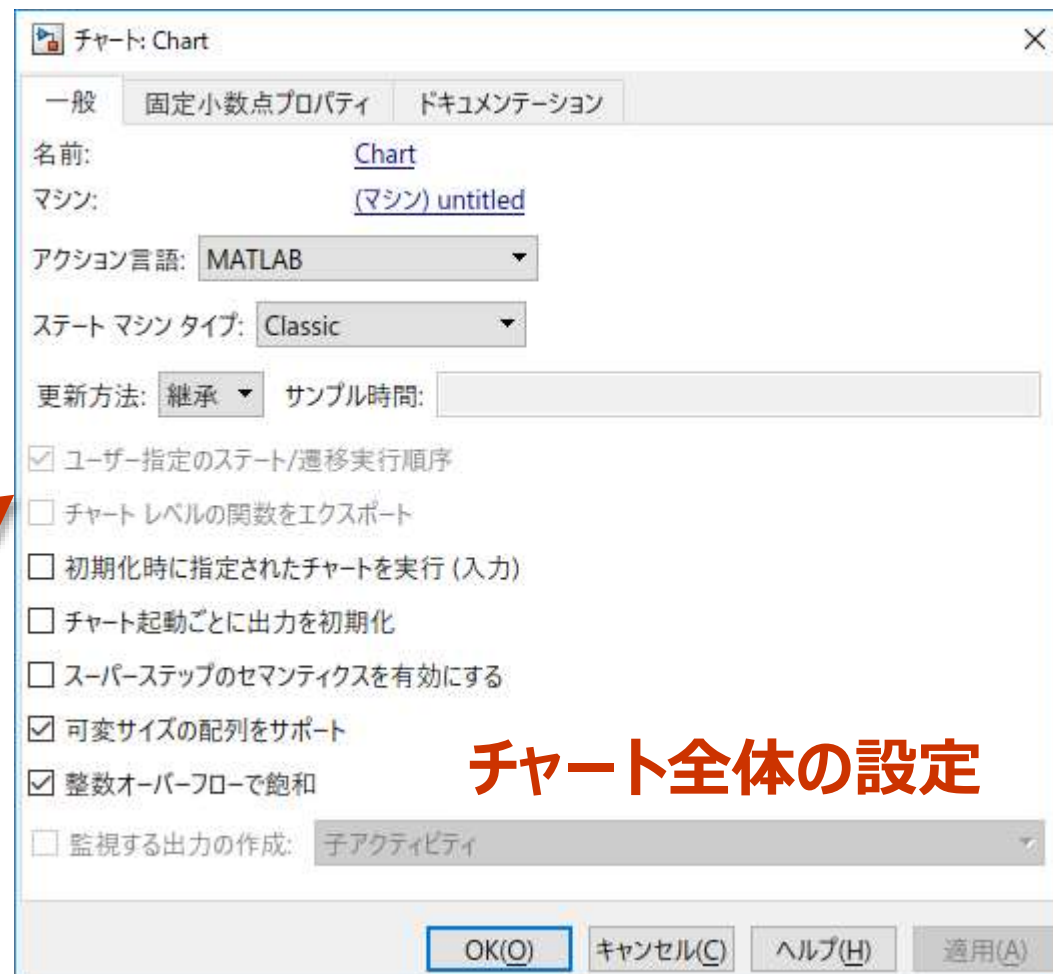


- ズーム
- 表示範囲のフィッティング
- 状態
- 分岐点（コネクティブジャンクション）
- 遷移線（トランジション）
- ボックス
- Simulink関数
- グラフィカル関数
- MATLAB関数
- 真理値表
- ヒストリージャンクション
- 注釈
- イメージ

Stateflow エディタ コンテキストメニュー & プロパティ



**エディタ余白を
右クリックで表示**



チャート全体の設定

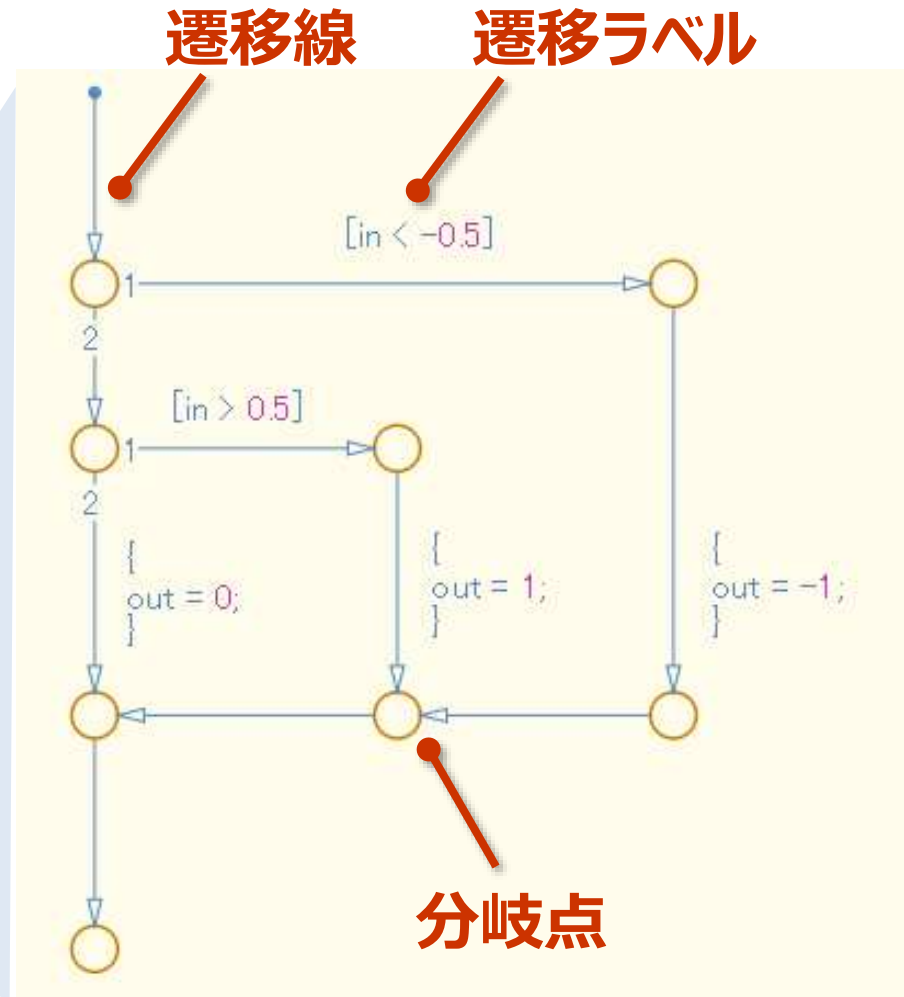
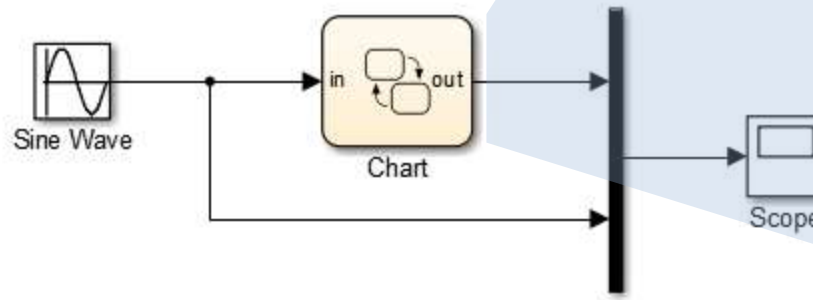
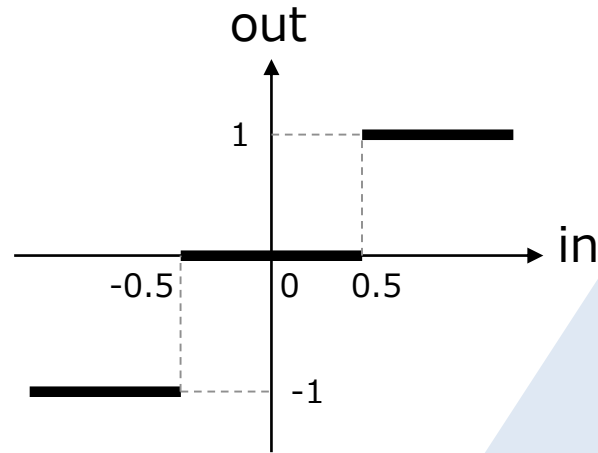
目次

- Stateflow 紹介
 - Stateflowとは
 - Stateflow基本環境
- Stateflow 演習
 - 例題1 フローチャート
 - 例題2 状態遷移図① 基本
 - 例題3 状態遷移図② 時相論理
 - 例題4 状態遷移図③ 階層化
 - 例題5 状態遷移図④ 並列状態
- 補足

例題1 フローチャート

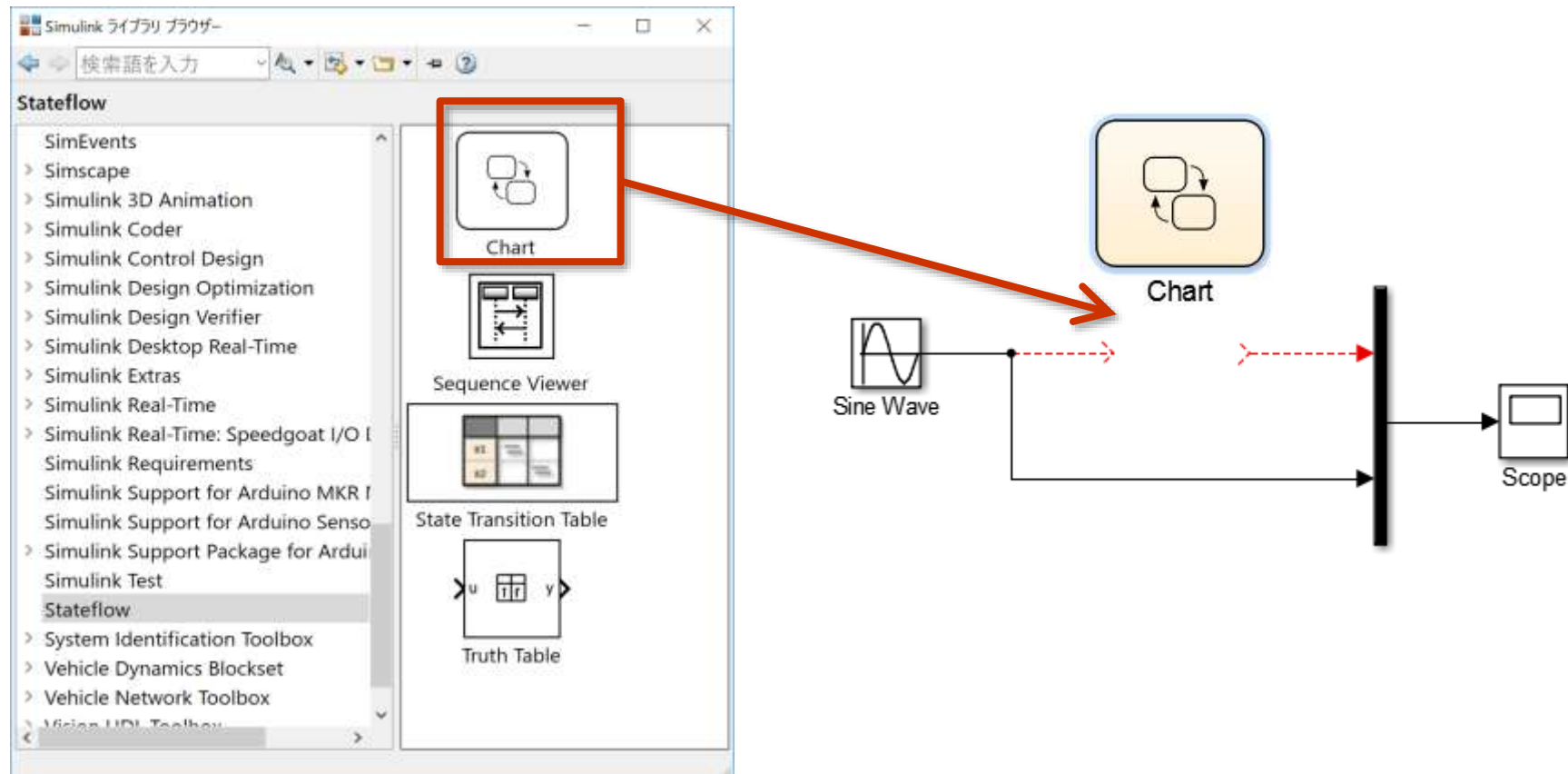
処理内容

- 入力値 : in、出力値 : out
- $\text{in} < -0.5$ なら $\text{out} = -1$
- $\text{in} > 0.5$ なら $\text{out} = 1$
- それ以外なら $\text{out} = 0$



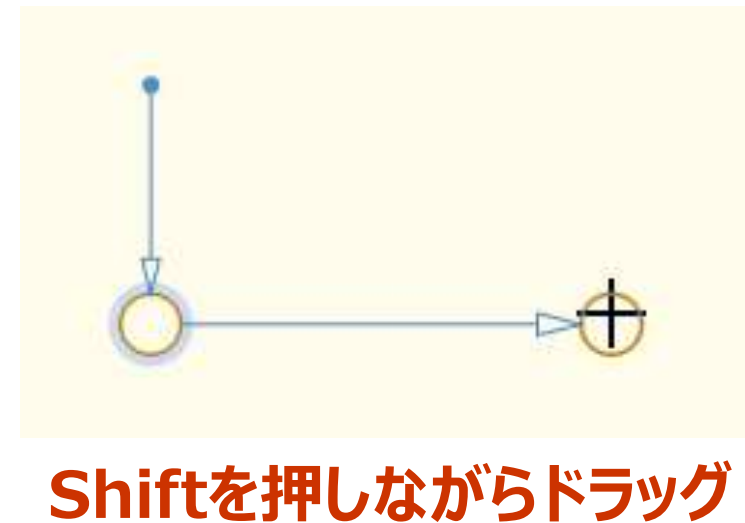
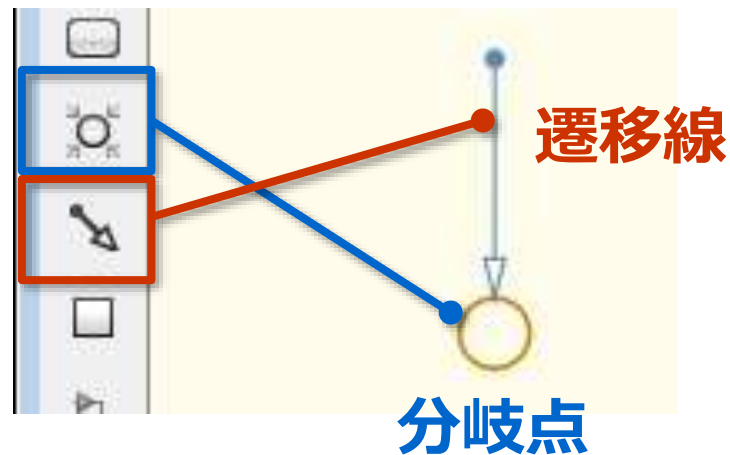
例題モデルオープン & Chartブロックの追加

1. ex1_flow_chart.slx を開きます。
2. StateflowライブラリからChartブロックをモデル内の適切な位置にドラッグ & ドロップします。



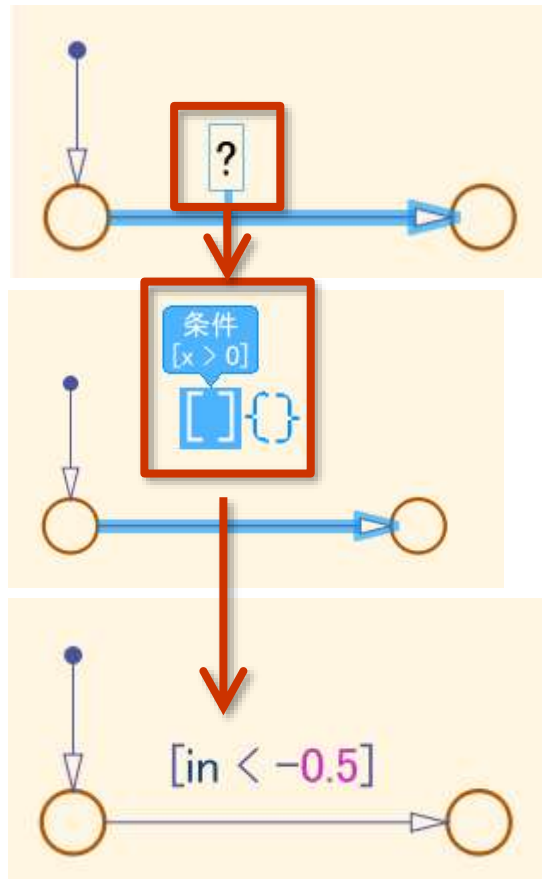
遷移線 & 分岐点の追加

1. オブジェクトパレットから分岐点を追加します。
2. 遷移線を追加し、分岐点に接続します。
3. Shiftを押しながら分岐点近傍で左クリック、そのままドラッグすると分岐点付きの遷移線を作成できます。



遷移ラベルの編集

1. 遷移したい遷移線を左クリックします。
2. ? が表示されたら左クリックしてラベルを編集します。
3. ラベルを左クリックして位置を変更できます。

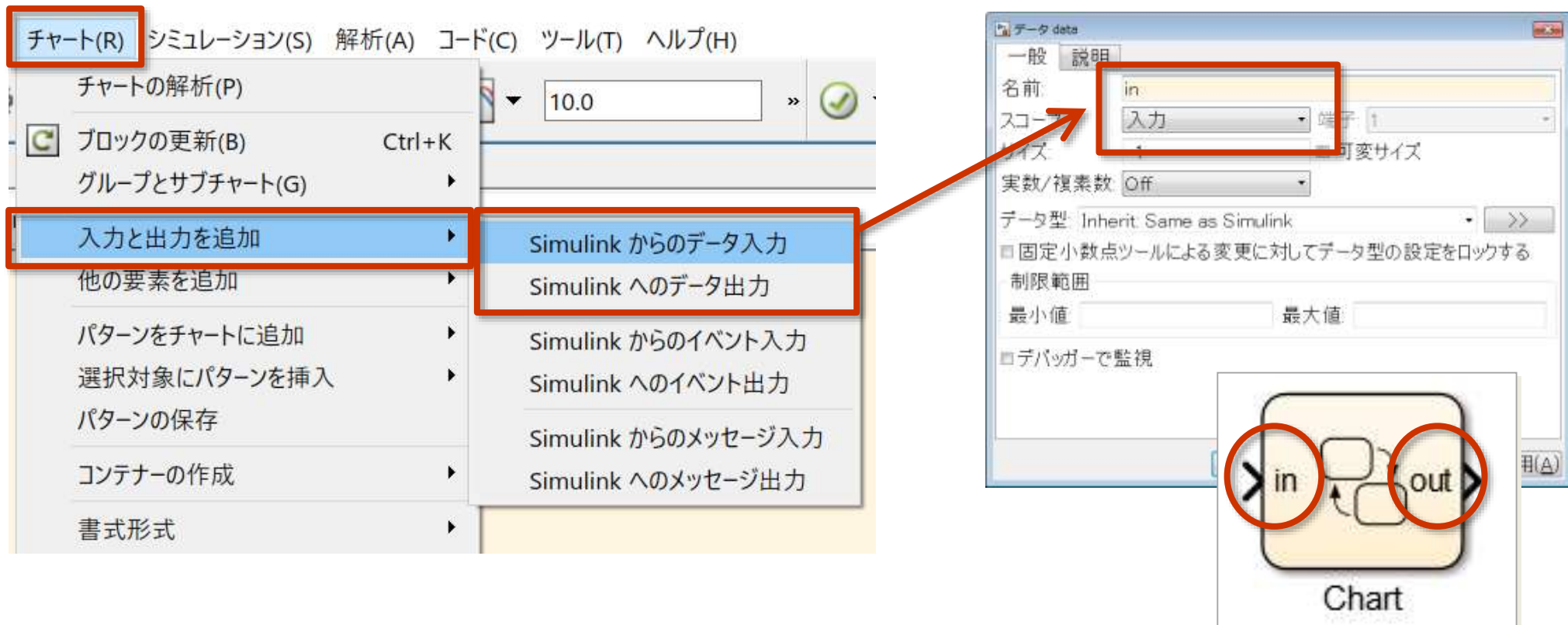


遷移ラベル表記法

- **[条件] { 条件アクション }**
同一ラベルに両方書くこともできますが、お勧めしません。
- C言語で記述
一部記述が異なる部分アリ（剰余演算子等）
- ラベルが無いと無条件で遷移
- 処理を改行して継続する際は ...（ピリオド3つ）
(例) **in1 == 1 && ...**
in2 == 1

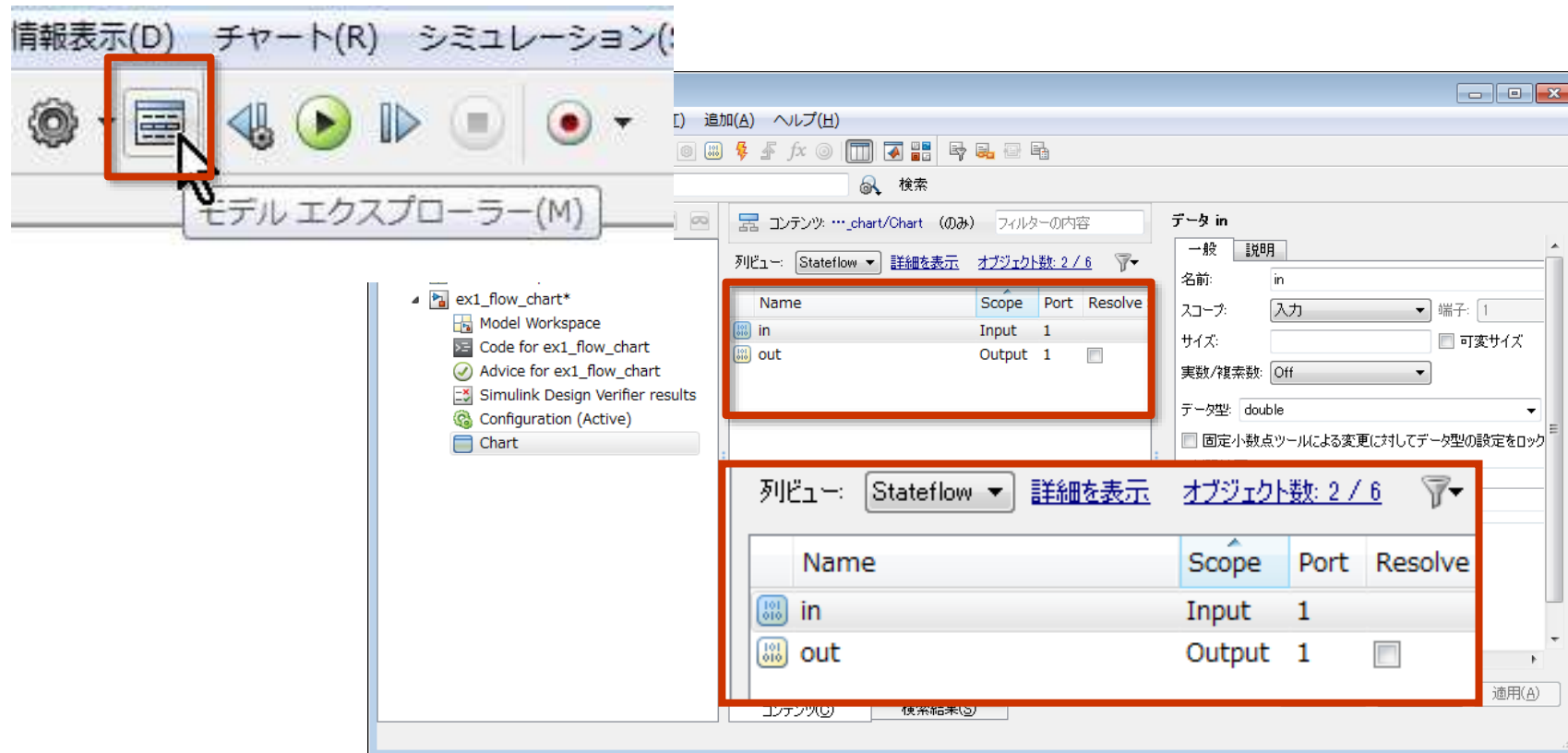
データの追加

1. 下記メニューから入力データ in、出力データ out を追加します。
2. 入出力データを作成すると、Chartブロックに入出力端子が表示されます。



定義したデータの確認 & 削除

1. モデルエクスプローラから定義したデータを確認 & 削除できます。
(データを追加することもできます)
2. データプロパティ (入出力順、スコープ、データ型等) を変更できます。

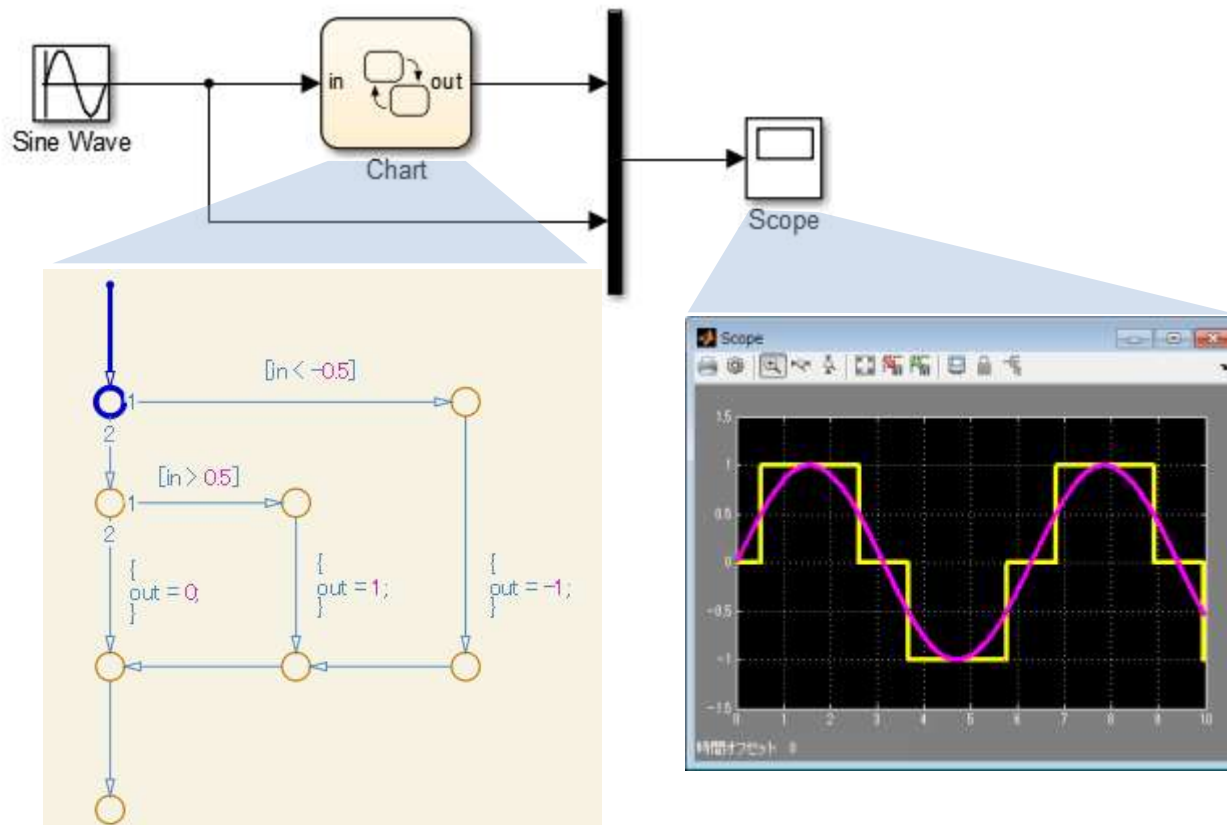


Stateflow データスコープ

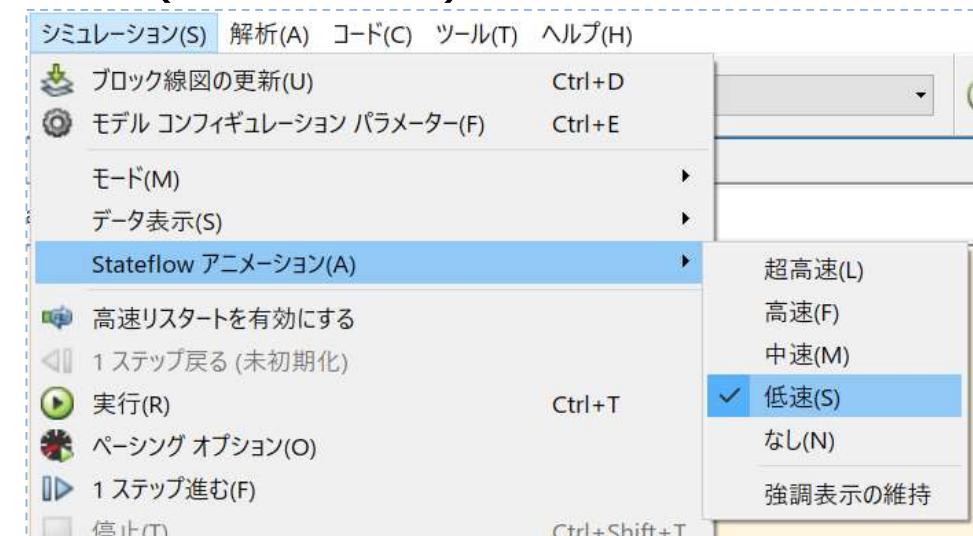
スコープ	特徴
入力	Simulinkからの入力
出力	Simulinkへの出力
ローカル	チャート内ローカル変数 階層上位のローカル変数は下位階層でも使用可能
パラメータ	同名のMATLABベースワークスペース値、またはマスクパラメータ値を参照
定数	定数値
データストアメモリ	Data Store Memory

入出力結線 & シミュレーション

1. in、outを接続してシミュレーションを実行します。
2. シミュレーション中にアクティブなパスがハイライト表示されます。

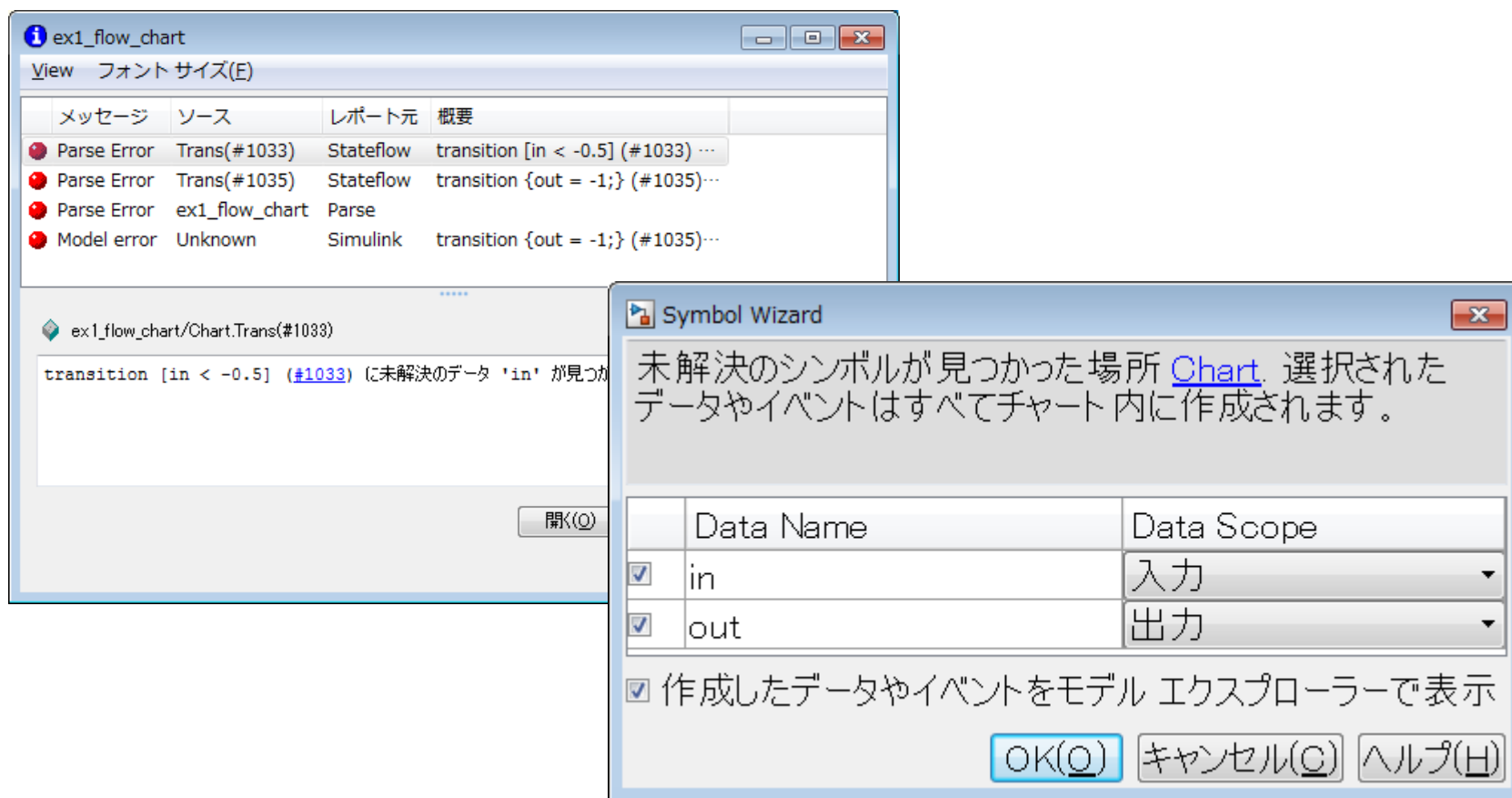


[アニメーション速度の選択]
シミュレーション→Stateflowアニメーション→
低速 (※任意選択)



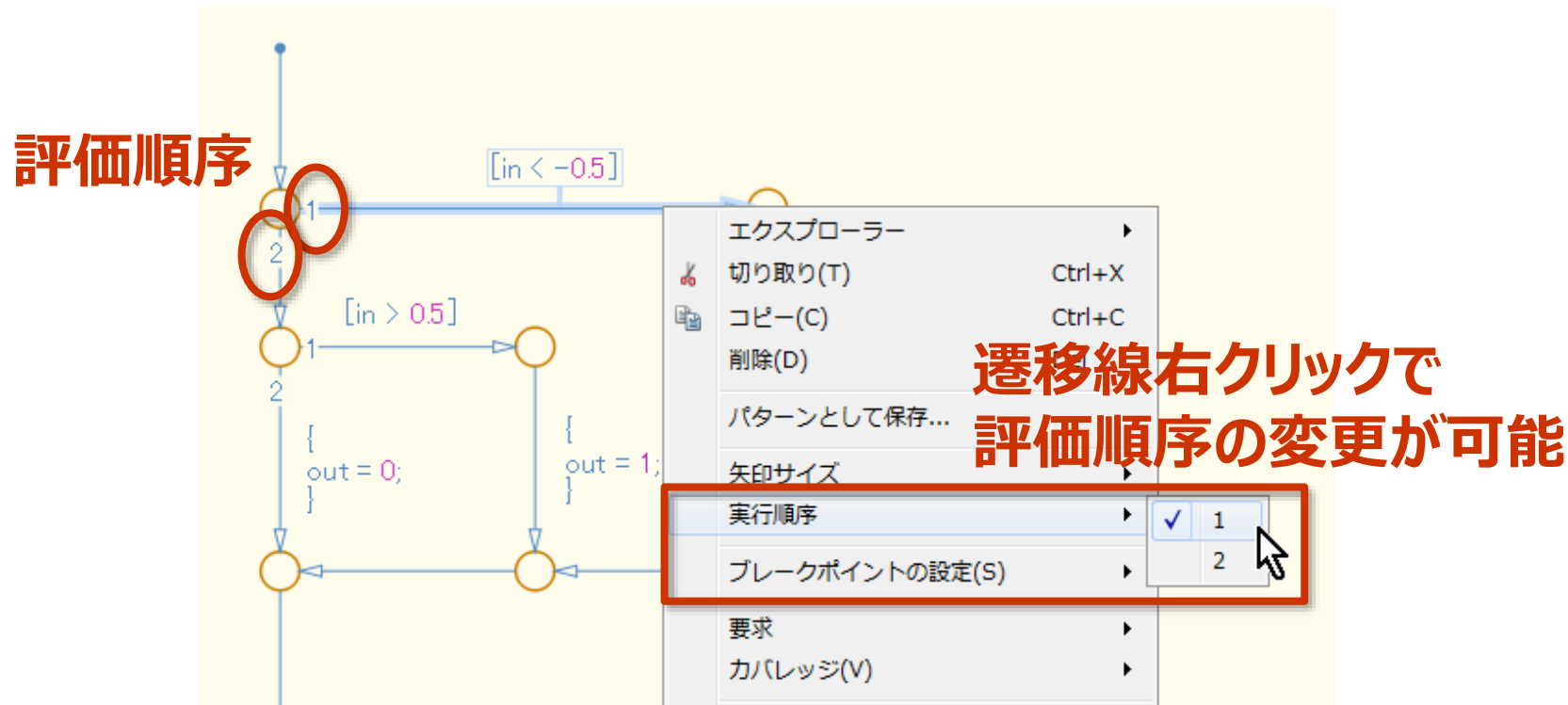
データを定義せずにシミュレーションを実行したとき

1. エラーダイアログと共に、データを追加するためのシンボルウィザードが起動します。



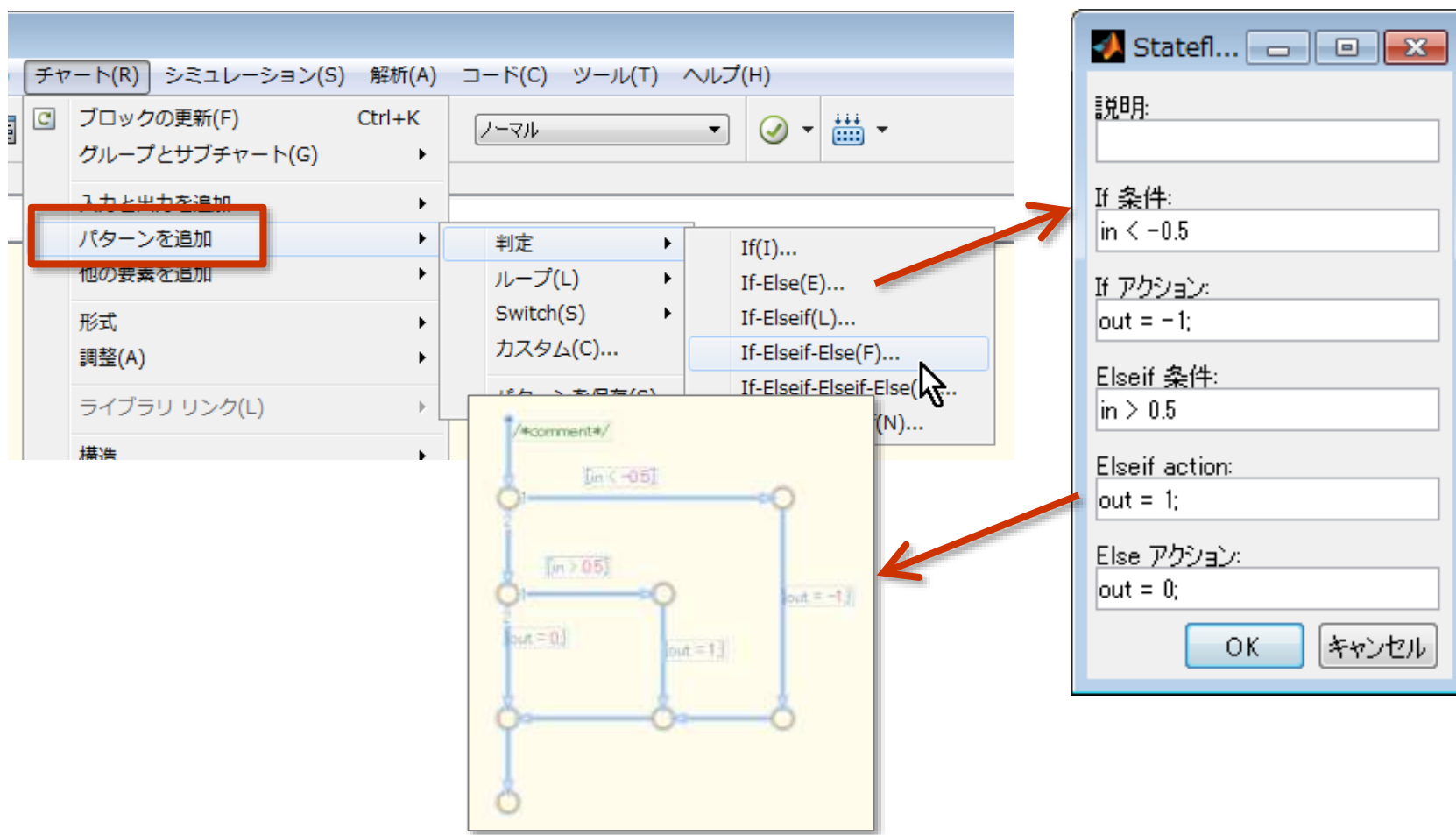
遷移線の評価順序

- 同一の分岐点・状態から複数の遷移線が出ているとき、その評価順序を設定できます。
- 通常、無条件の遷移線（elseに相当するモノ）があれば、それを最後に実行するように設定します。



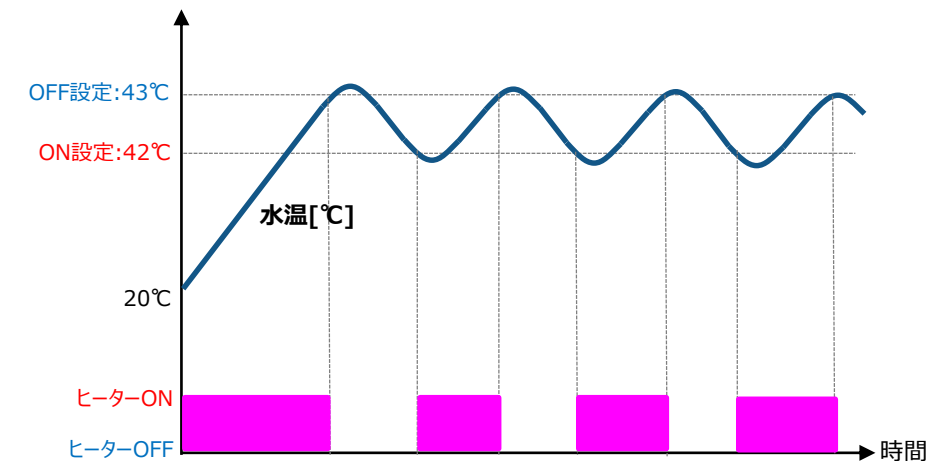
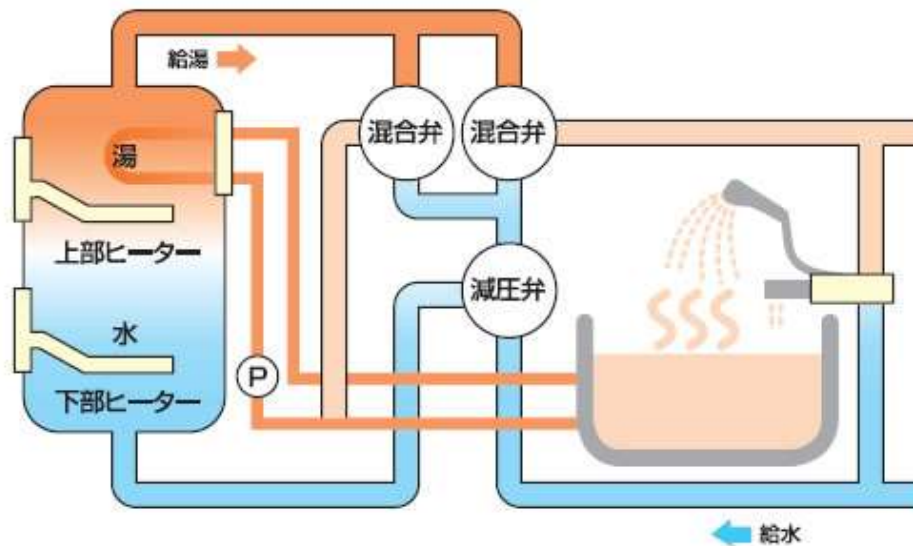
フローパターンの利用

1. 下記メニューから典型的な処理（条件分岐、繰り返し）を選んでフローチャートを作成することができます。



状態遷移図例題： ボイラー水温のON/OFF制御

- 水温が **42 °C ↓** で **ヒーターON：加熱**
- 水温が **43 °C ↑** で **ヒーターOFF：自然冷却**
- 初期水温： **20°C**

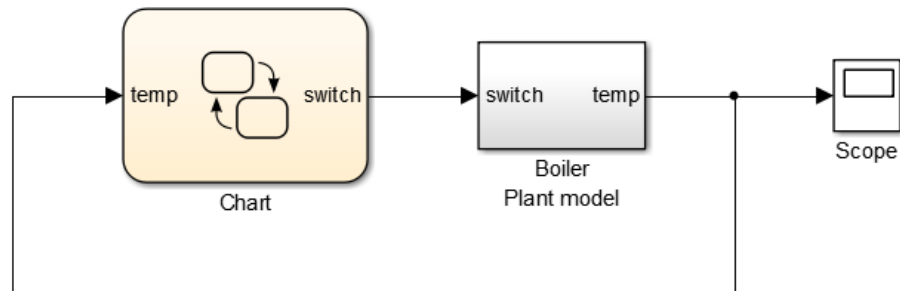


画像引用元：http://www.yonden-yes.co.jp/jyutaku/els_qanda

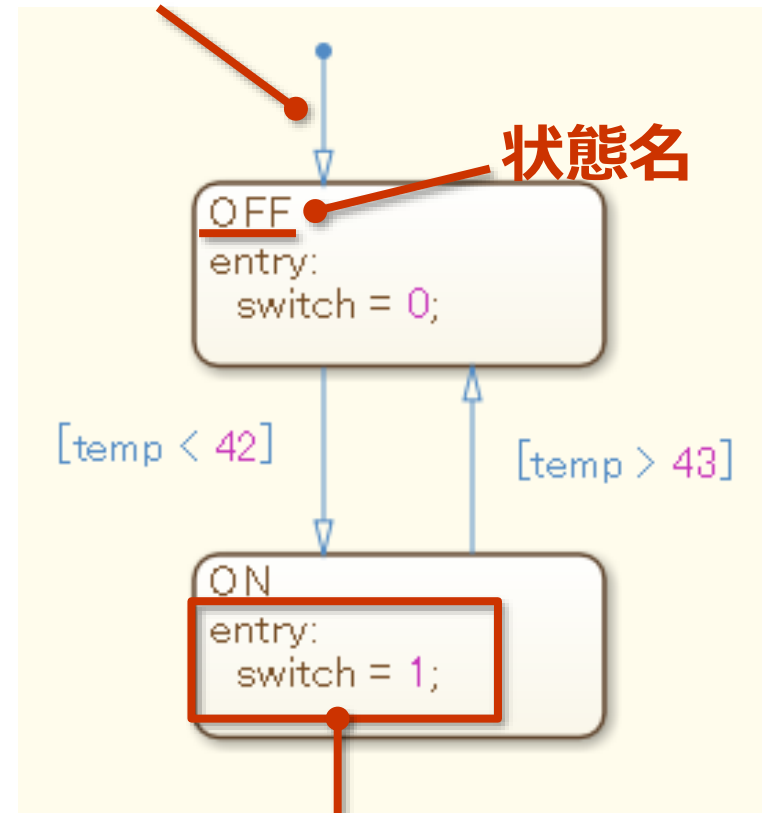
例題2 状態遷移図① 基本

処理内容

- ON/OFF状態：ヒータスイッチON/OFF
- OFF状態から開始する。
- 水温が42℃未満だとスイッチON
- 水温が43℃を超えるとスイッチOFF



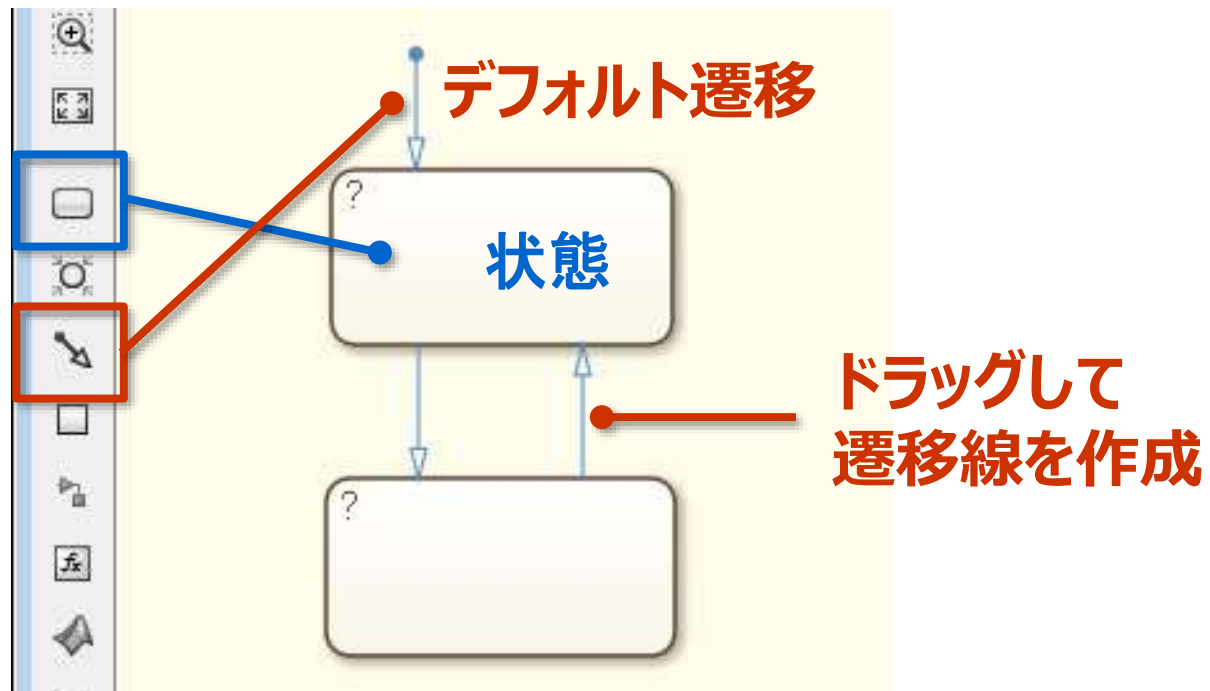
デフォルト遷移 (最初の状態を指定)



状態アクション

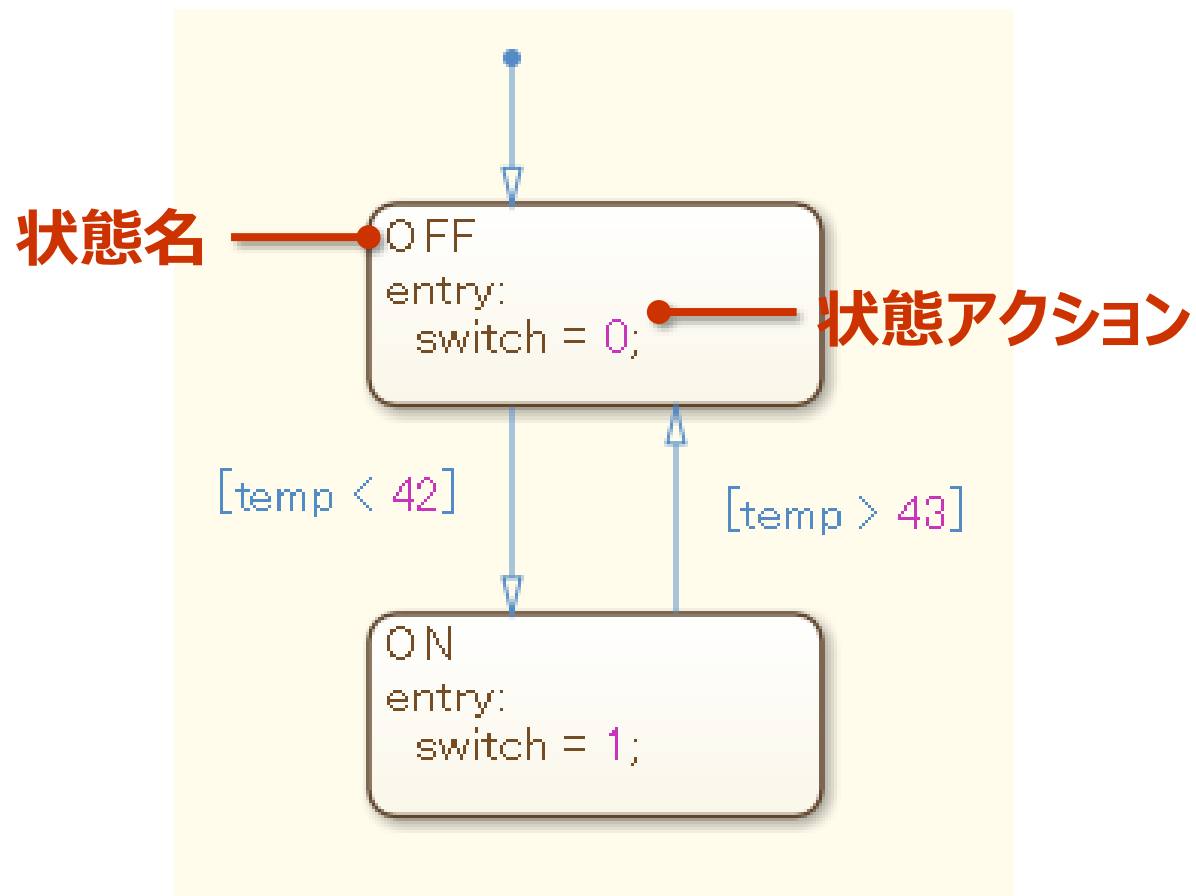
例題モデルオープン & 状態・遷移線の追加

1. ex2_state_chart.slx を開き、Chartブロックを開きます。
2. オブジェクトパレットから状態を追加します。
3. 遷移線を追加してデフォルト遷移を定義します。
4. 状態近傍で左クリック、そのままドラッグすると遷移線を作成できます。



状態名・状態アクション・遷移ラベルの編集

1. 状態内の?を左クリックすると編集できます。
2. 1行目が状態名、2行目以降が状態アクションになります。



状態アクション

アクション	略記	特徴
entry	en	状態がアクティブになった時に実行
during	du	状態がアクティブな間に実行
exit	ex	状態が非アクティブな時に実行

状態アクション表記法

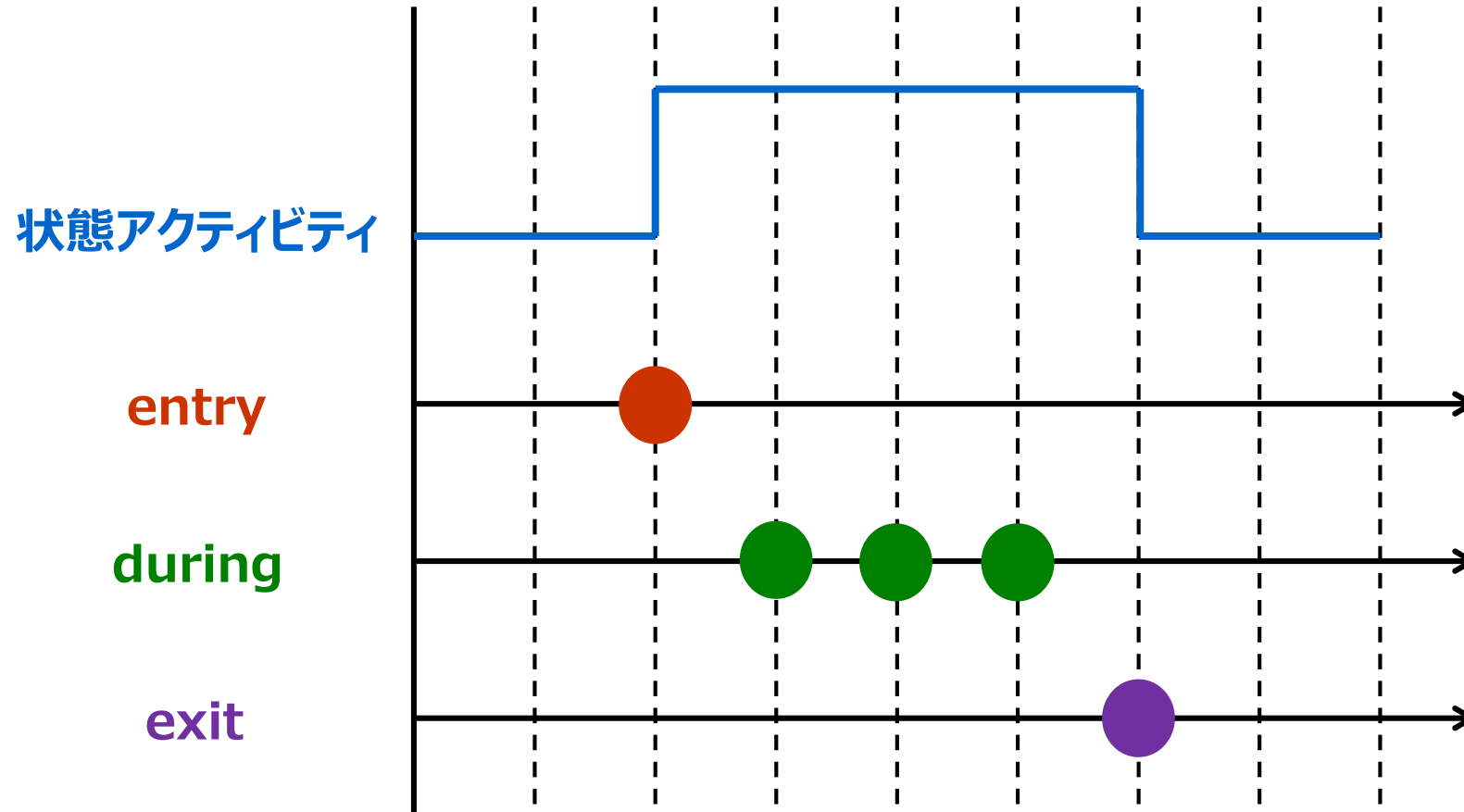
- **アクション名: 状態アクション**

アクション名と状態アクションの間は改行可能です。

(例) entry: out = 1; (en: out = 1;), en, du: out = 1;

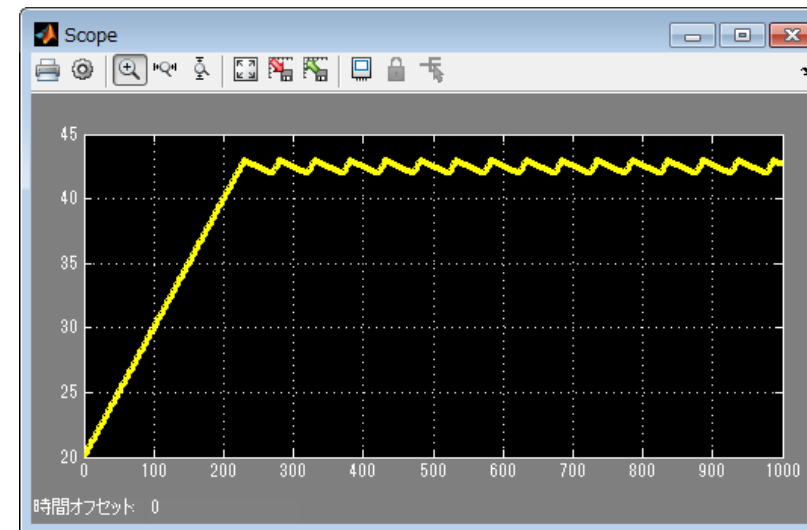
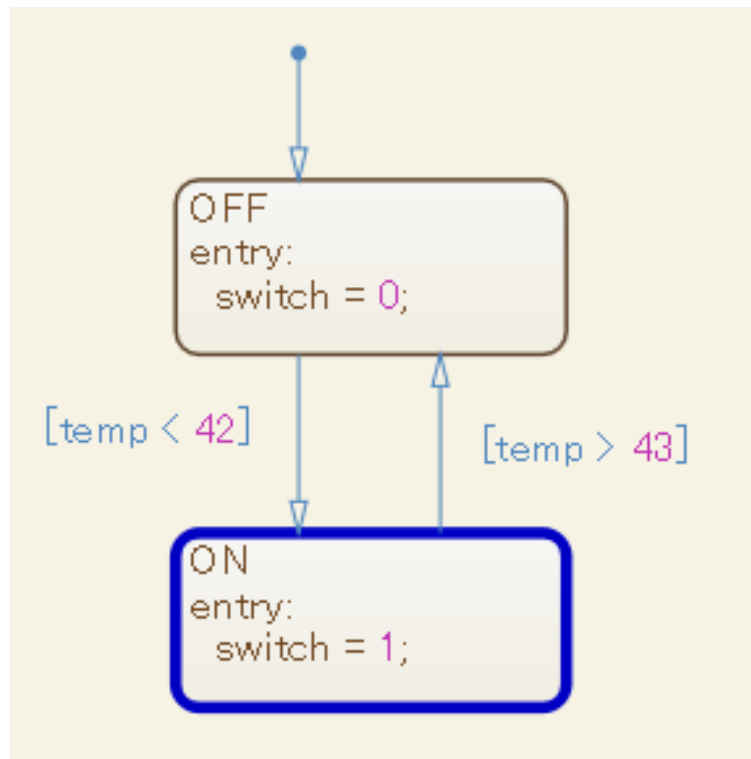
- アクション名を省略するとentryアクションになります。
- entry と duringで同じ処理を行うときは entry, during: と書けます。

状態アクションの実行タイミング



シミュレーション

1. シミュレーションを実行します。
2. シミュレーション中にアクティブな状態をハイライト表示します。

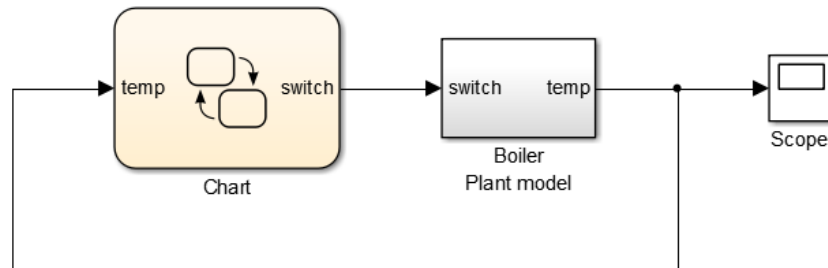


**42℃～43℃の範囲内に
制御されています**

例題3 状態遷移図② 時相論理

処理内容（例題2に仕様追加）

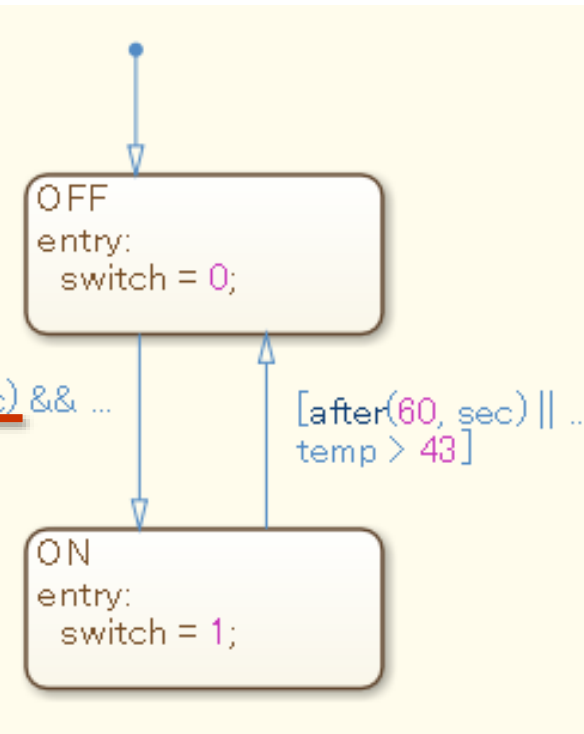
- スイッチOFFからONの切替を行う際に20秒待つ。
- スイッチONの状態が60秒続くといったんOFFする。
(ヒータの過動作防止)



時相論理

[after(20, sec) && ...
temp < 42]

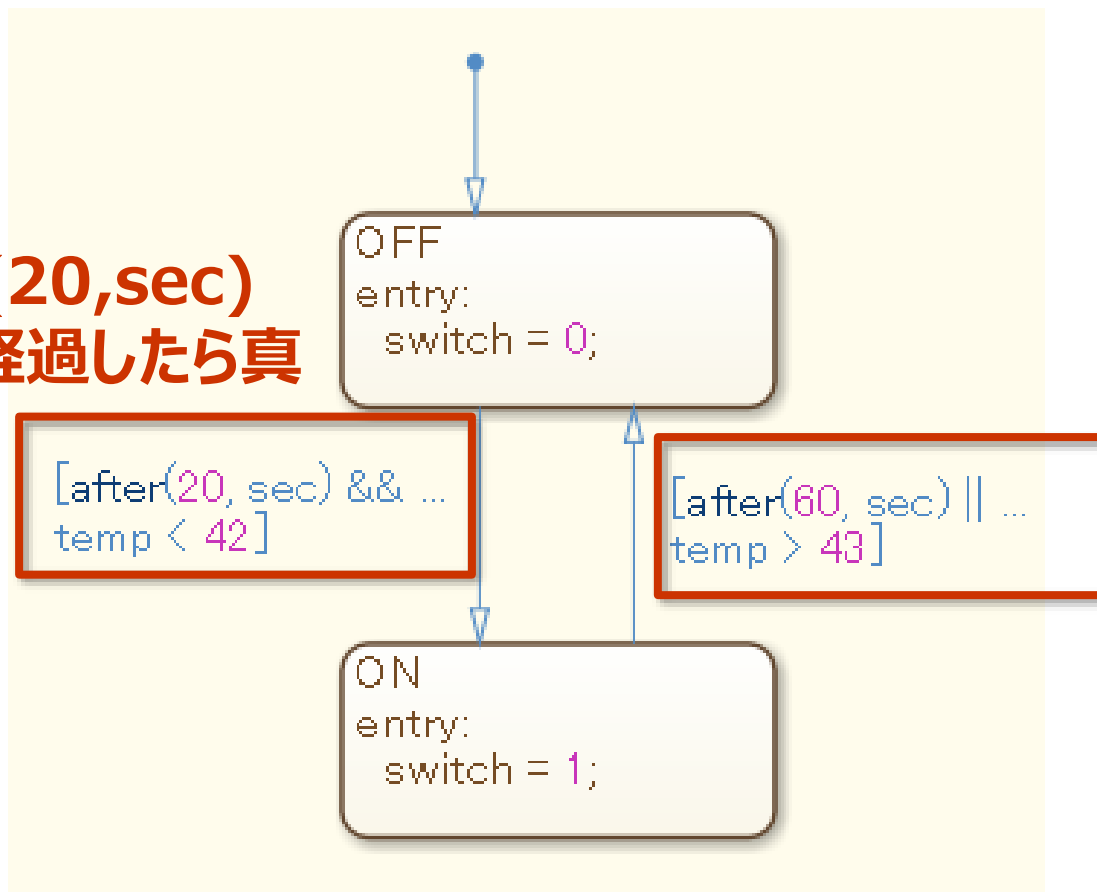
[after(60, sec) || ...
temp > 43]



例題モデルオープン & 時相論理の追加

1. ex3_state_chart_temporal.slx を開き、Chartブロックを開きます。
2. 遷移ラベルにafter関数を追加します。

**after(20,sec)
20秒経過したら真**



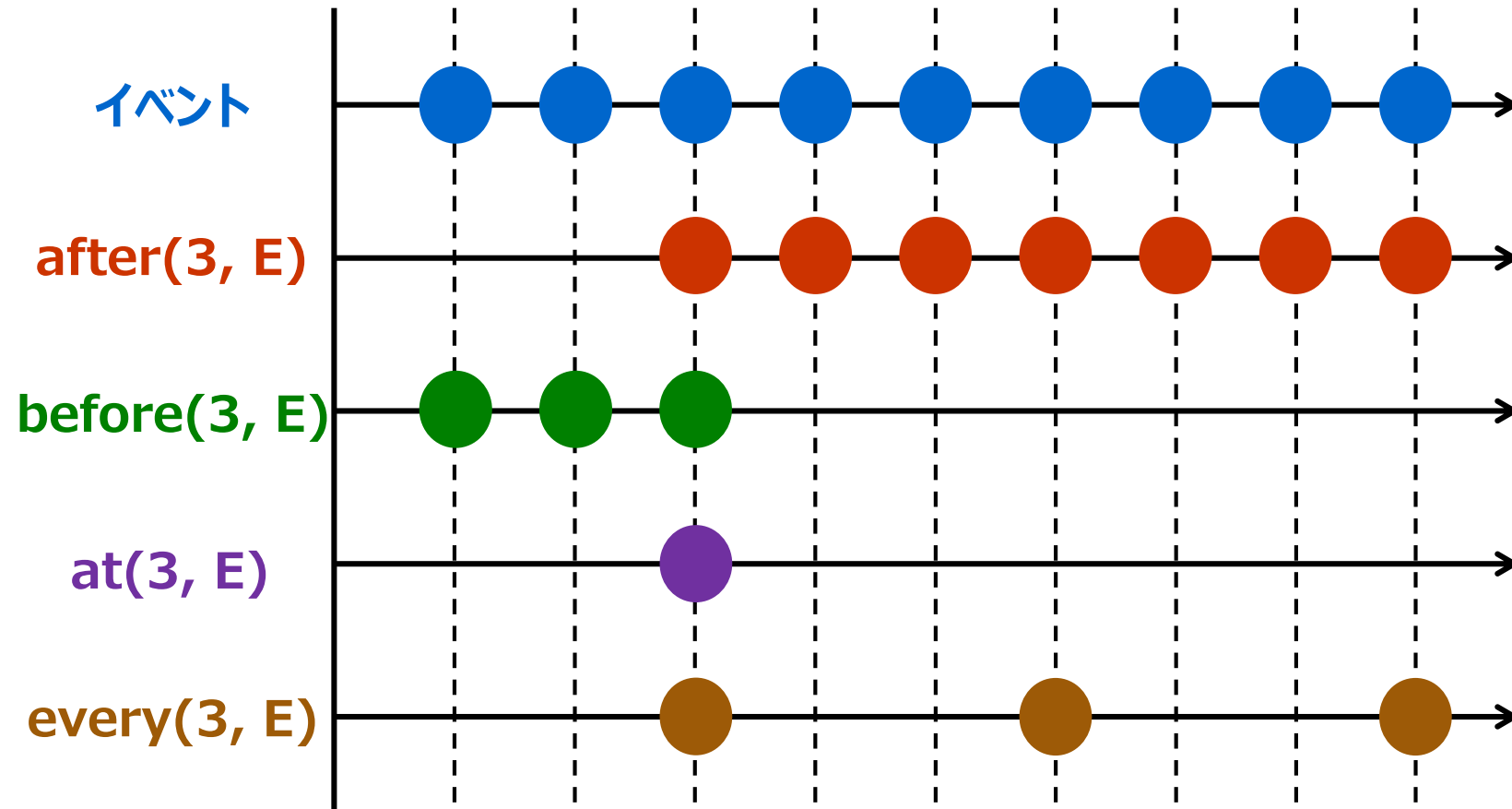
時相論理

関数	内容
after(N, E)	イベントEがN回発生以上のとき真
before(N, E)	イベントEがN回発生以下のとき真
at(N, E)	イベントEがN回発生した時点のみ真
every(N, E)	イベントEがN回発生するごとに真

時相論理表記法

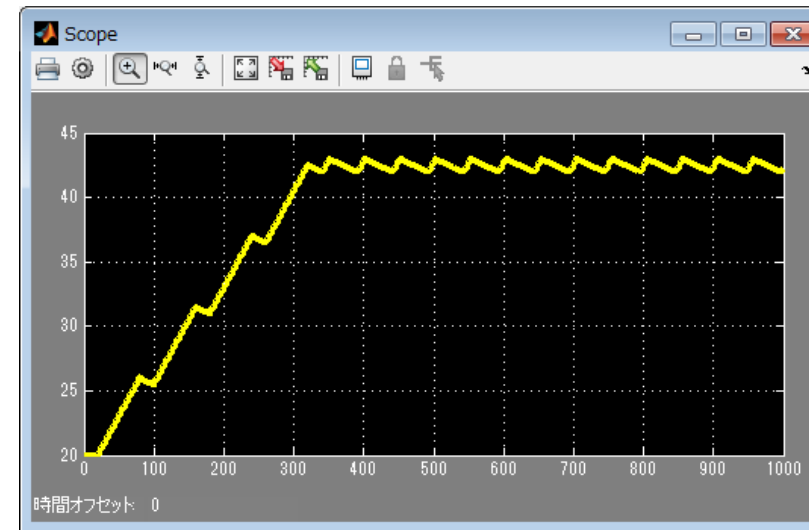
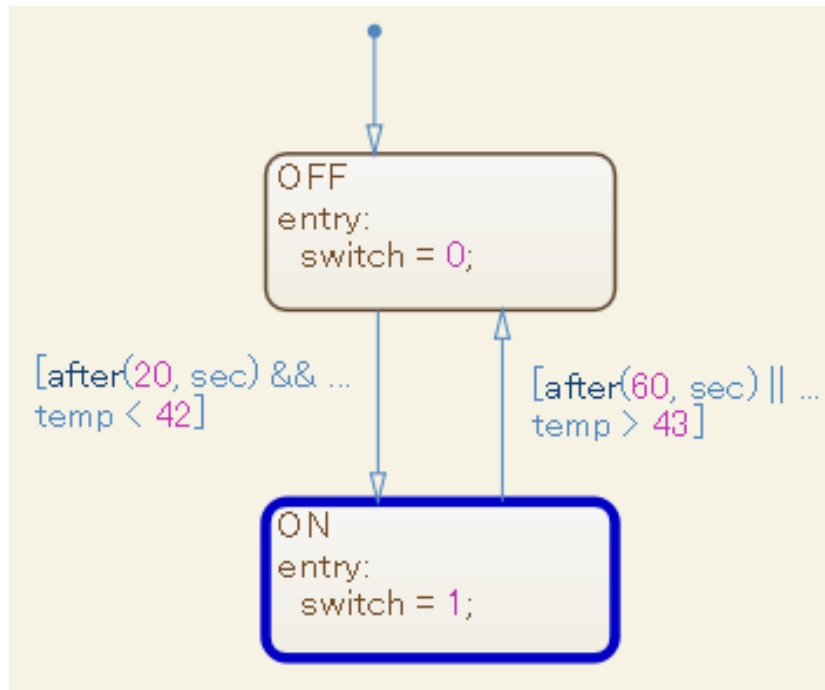
- **関数名(回数, イベント名)**
- 時間に関する予約イベントとして sec と tick があります。
 - sec : 絶対時間 [秒]
after(10, sec)で10秒経過したら真
 - tick : チャート駆動回数
チャートが100ms周期の場合、**after(100, tick)**で10秒経過したら真

時相論理の実行タイミング



シミュレーション

1. シミュレーションを実行します。

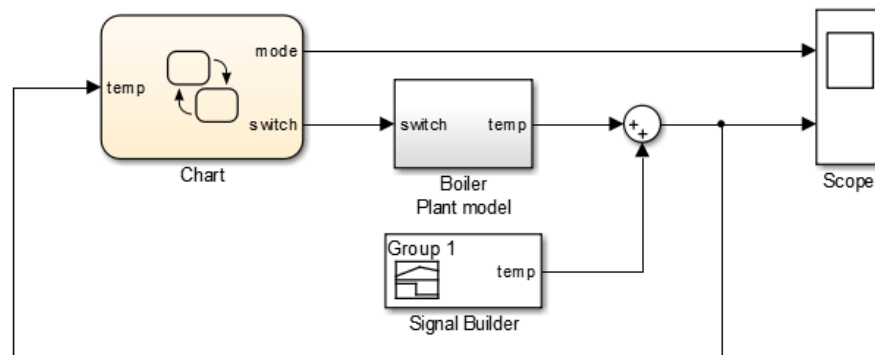


ヒータの連続動作が60秒毎にリセットされています

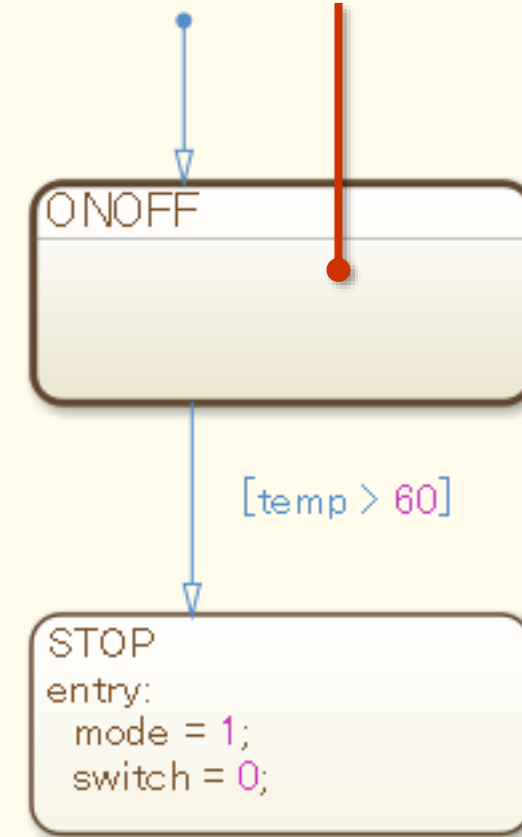
例題4 状態遷移図③ 階層化

処理内容（例題3に仕様追加）

- 水温が60℃を超えるとヒータを緊急停止する。
- 正常モード(0)と緊急停止モード(1)を出力する（データ名： mode）。



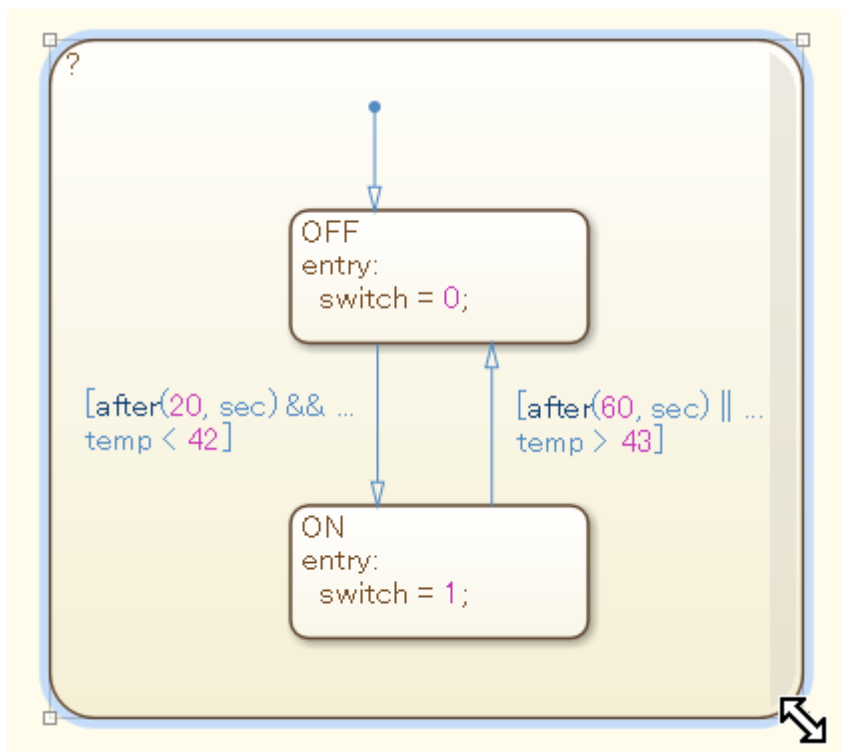
親状態
(スーパーステート)



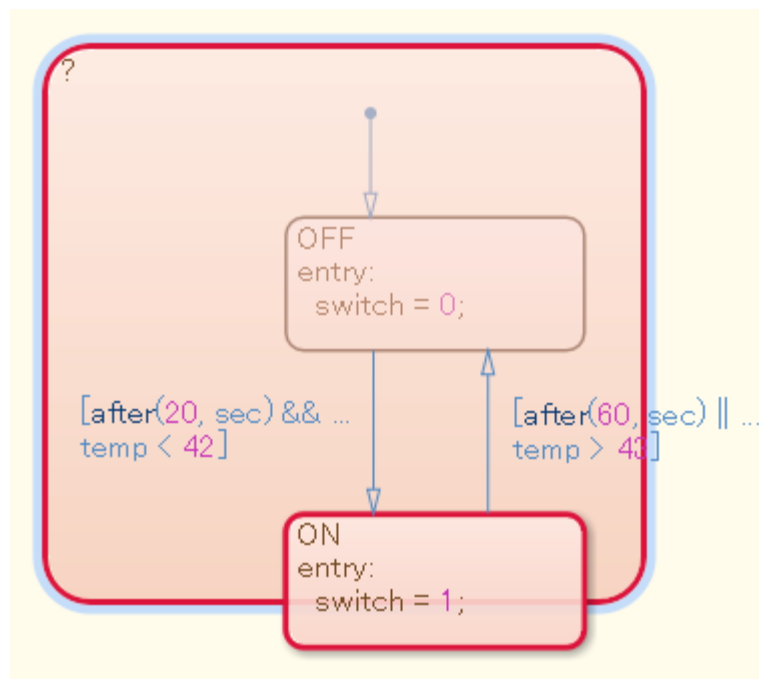
例題モデルオープン & 親状態の追加

1. ex4_state_chart_hierarchy.slx を開き、Chartブロックを開きます。
2. 状態オブジェクトを追加し、ON/OFF状態全体を囲みます。

全体を囲む

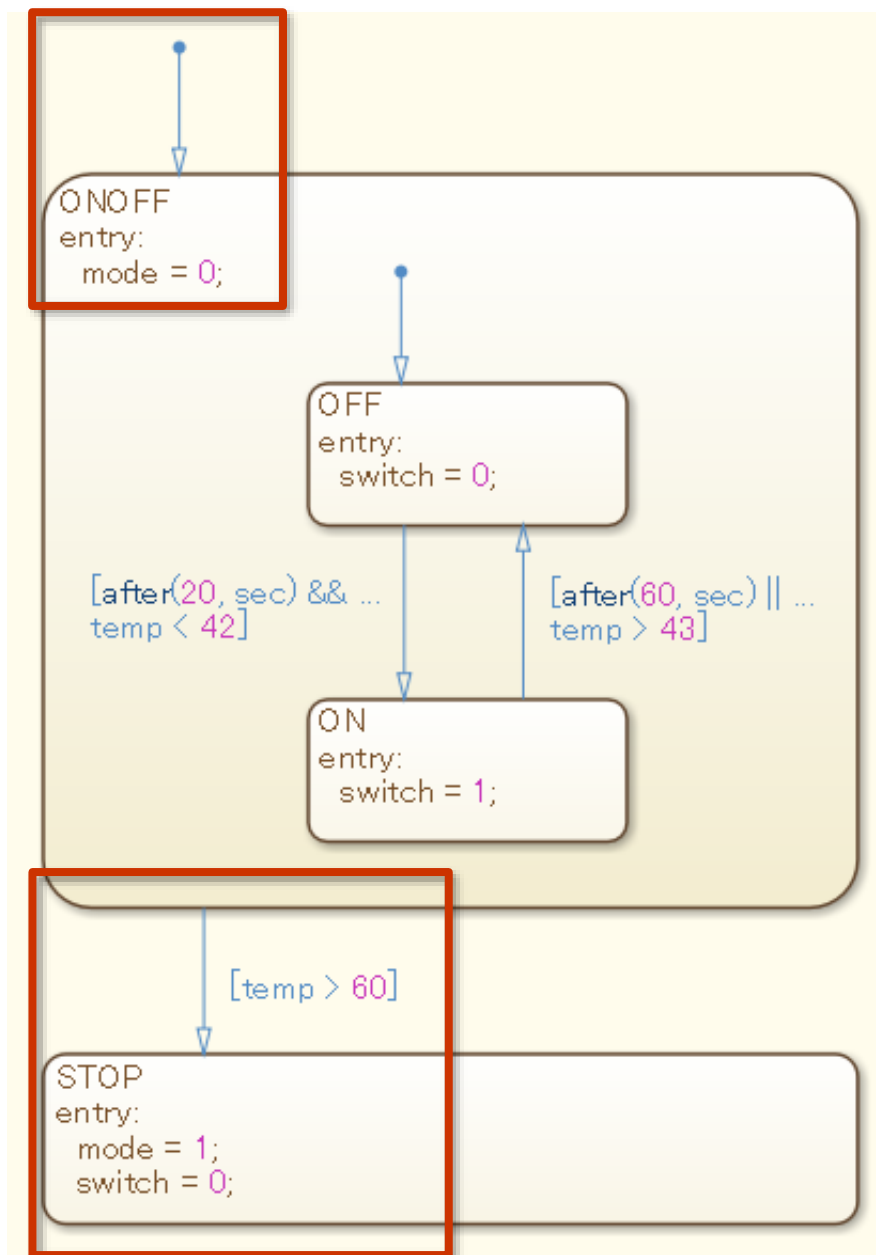


重なるとNG



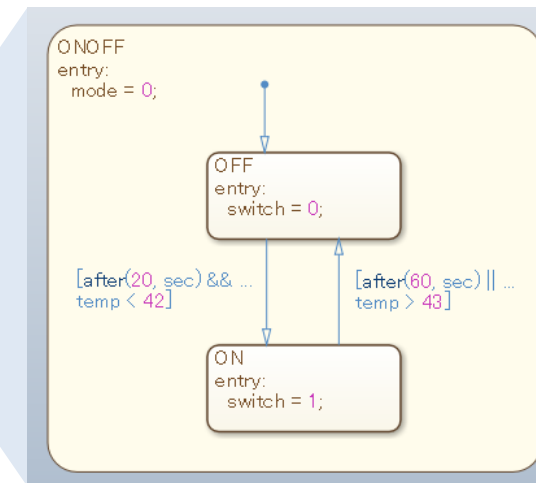
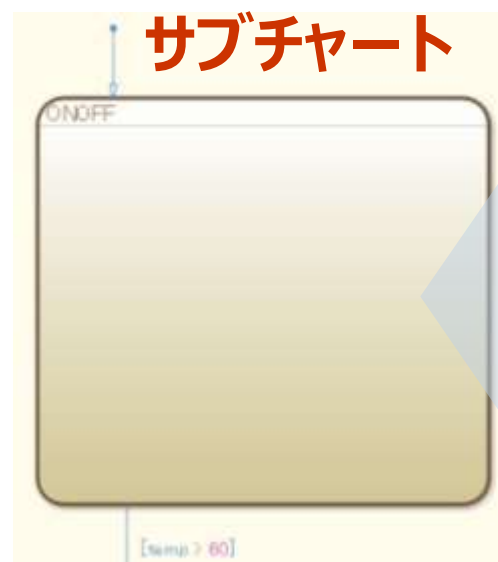
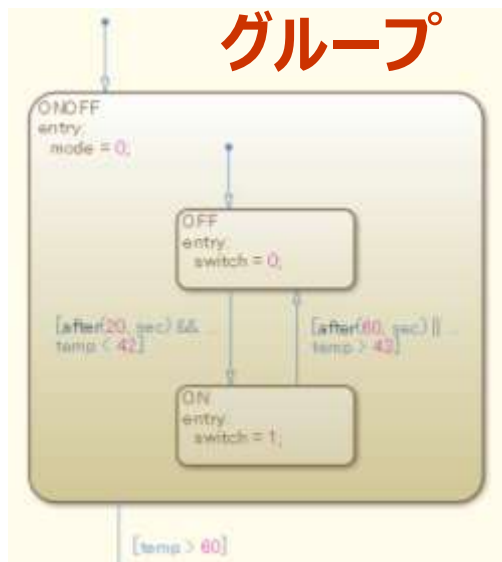
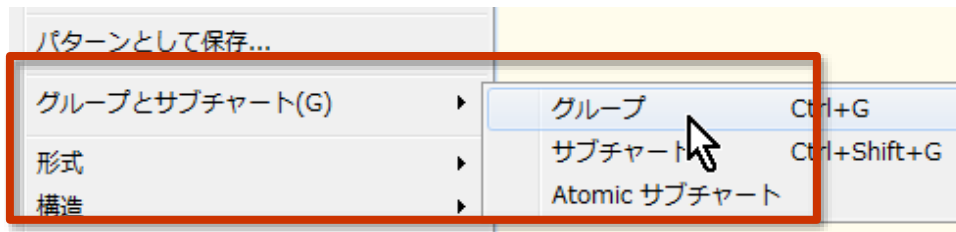
状態遷移図の作成

1. 右の完成図を参考にして、必要なオブジェクトの追加・編集を行います。



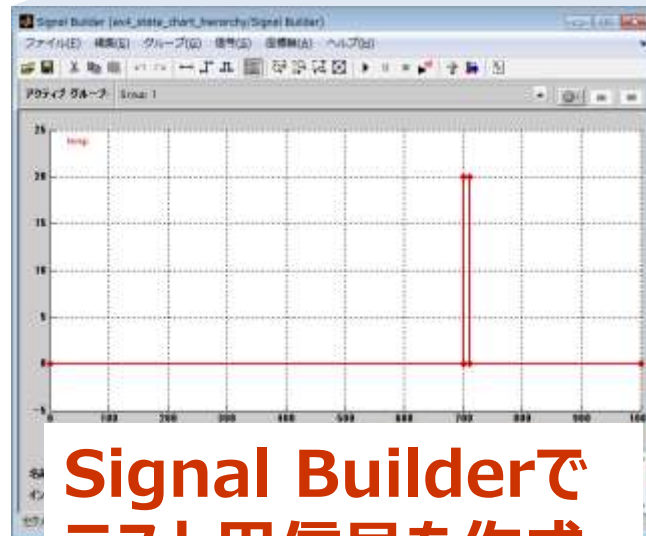
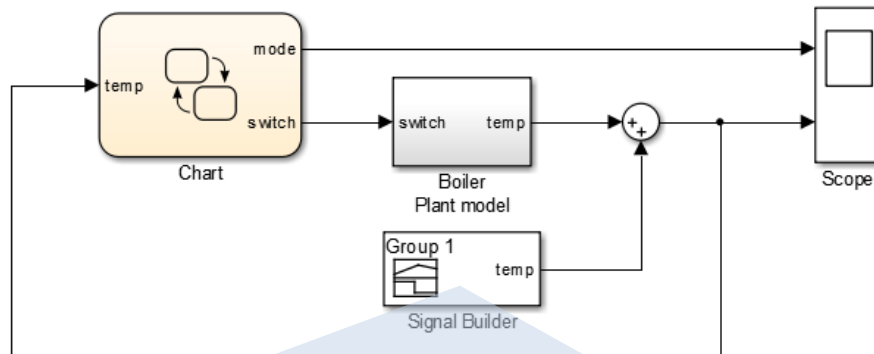
グループとサブチャート

1. 状態内の余白を右クリックして下記メニューを呼び出します。
2. [サブチャート]を選択すると、状態遷移図を階層化できます。
3. [グループ]を選択すると、グループ化できます。



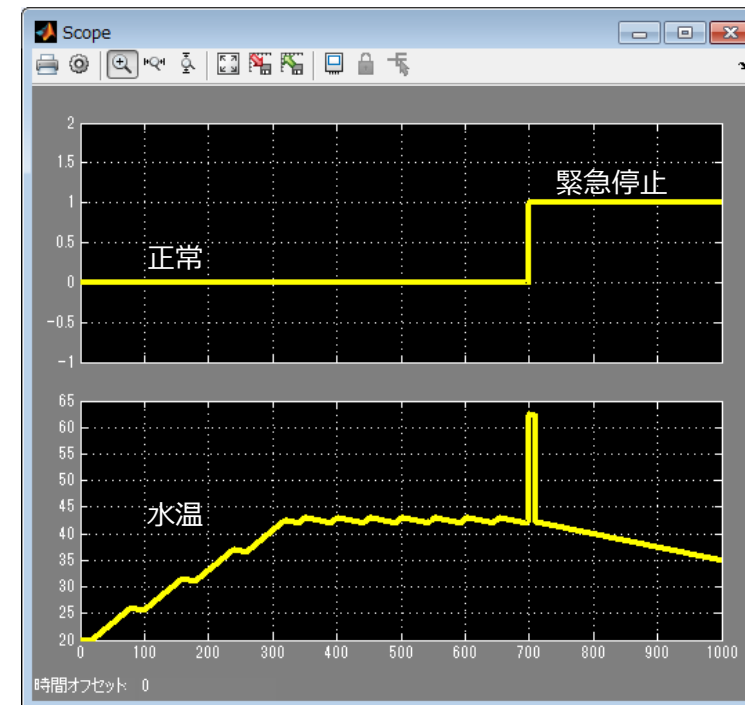
シミュレーション

1. シミュレーションを実行します。



**Signal Builderで
テスト用信号を作成**

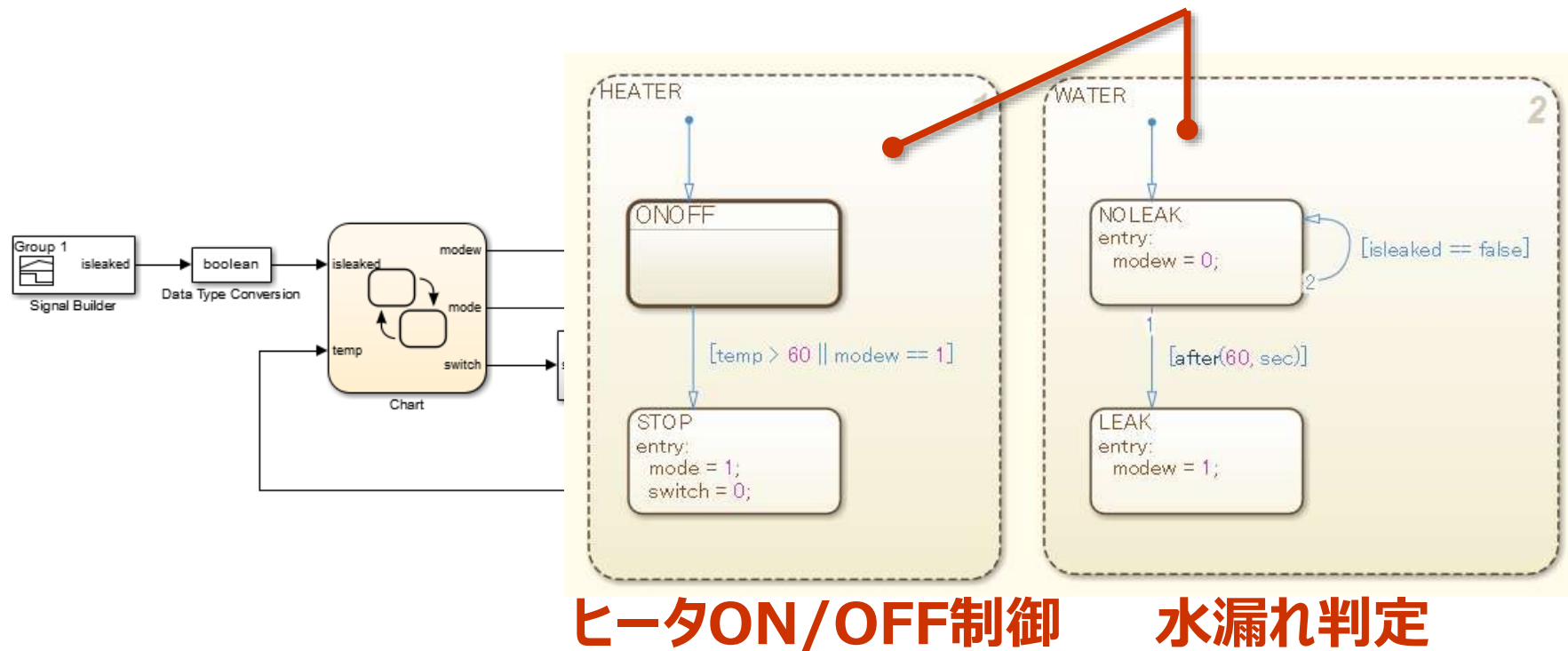
**水温が60℃を超えると
ヒータが停止しています**



例題5 状態遷移図④ 並列状態

処理内容（例題4に仕様追加）

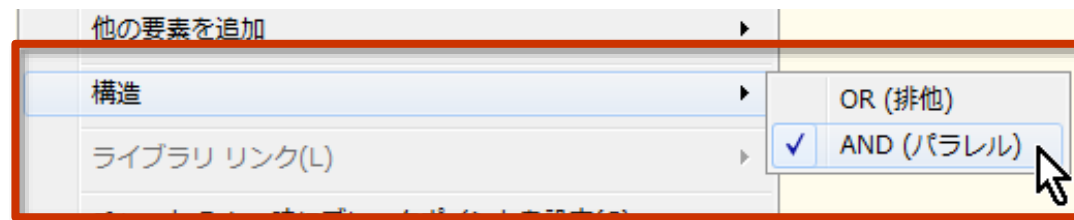
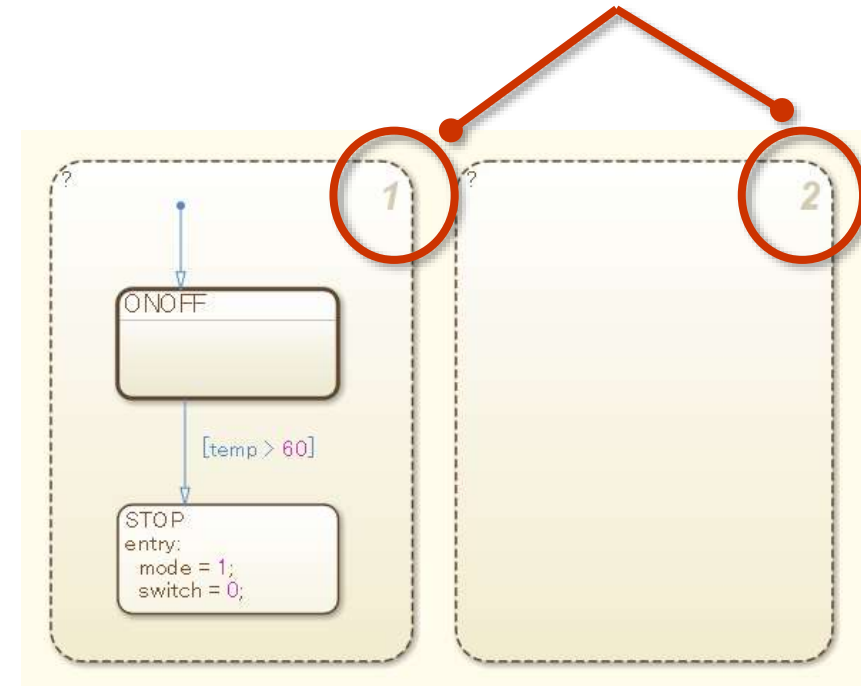
- 水漏れが60秒以上続いたとき、ヒータを緊急停止する。
- 水漏れチェック入力データ：isleaked
- 水漏れ判定モードデータ：modew



例題モデルオープン & 並列状態の追加

1. ex5_state_chart_parallel.slx
を開き、Chartブロックを開きます。
2. 状態オブジェクトを追加し、
既存の状態全体を囲みます。
3. もう一つ状態を追加します。
4. エディタの余白を右クリックして
下記メニューを呼び出し、
[AND (パラレル)] を選択します。

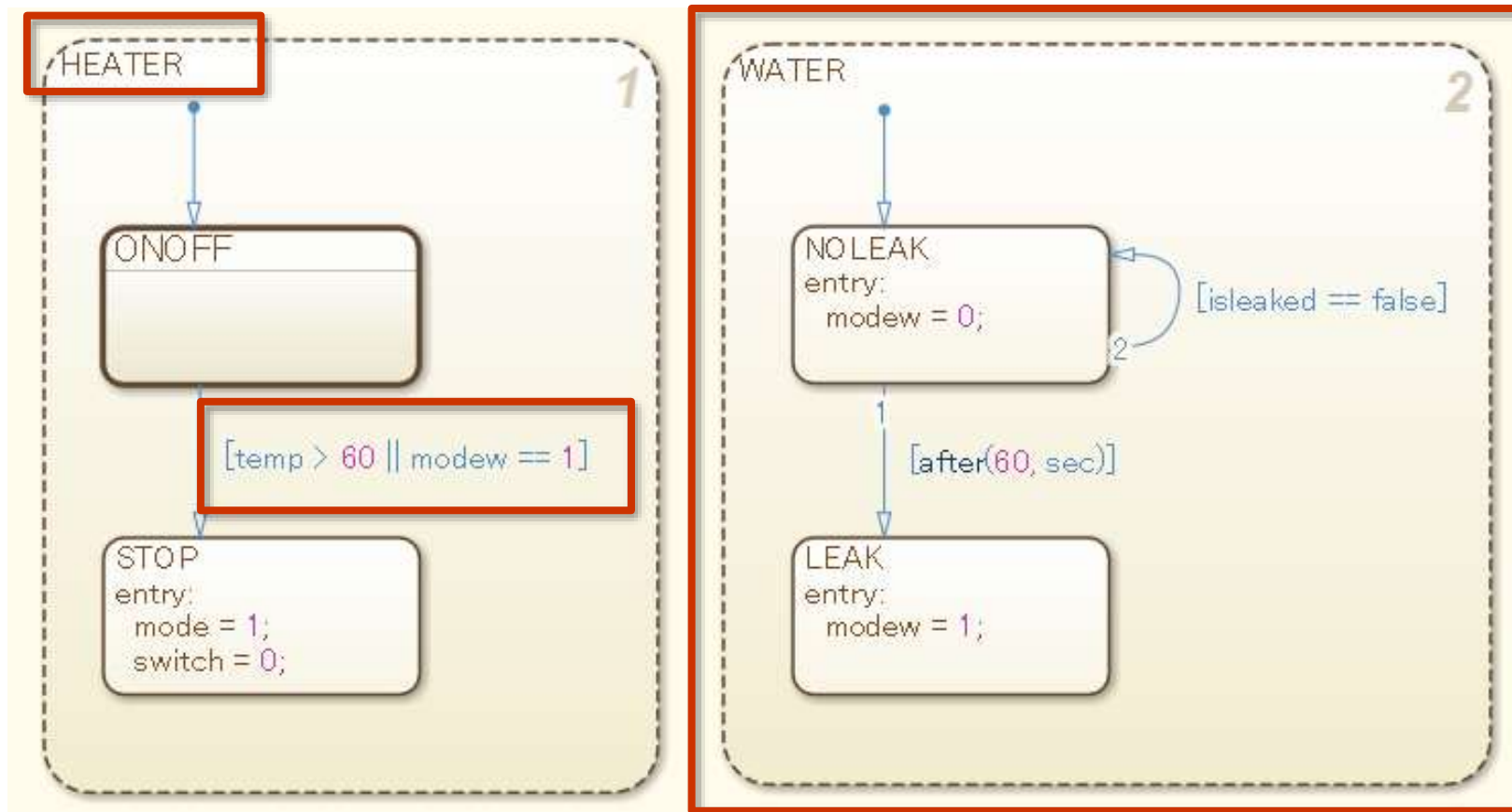
同一時刻での実行順序



並列状態を右クリック
→[実行順序]で実行
順序を変更できます

並列状態の編集

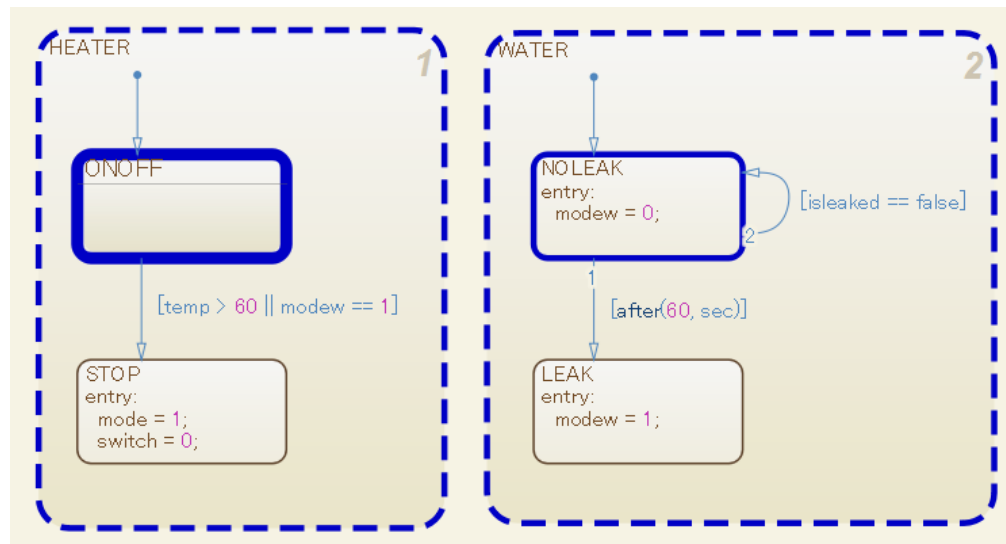
1. 下の完成図を参考にして、必要なオブジェクトの追加・編集を行います。



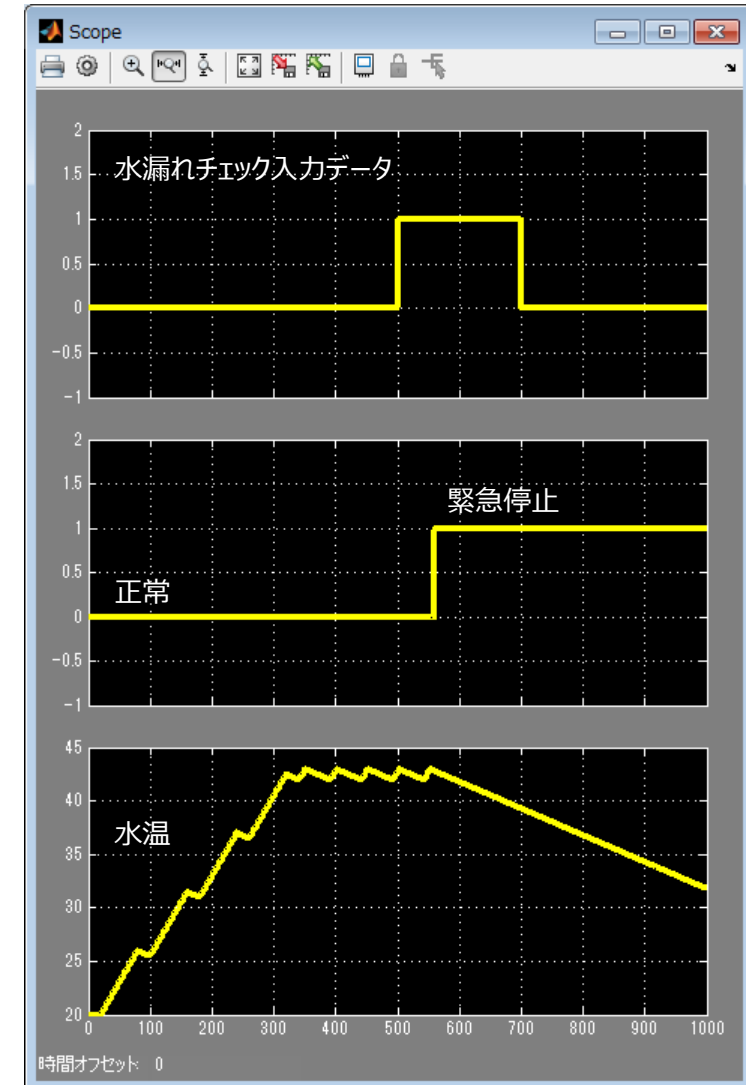
シミュレーション

1. シミュレーションを実行します。
2. シミュレーションが遅い場合は、Stateflowエディタを閉じて再度実行してください。
(アニメーション表示に時間がかかるケースがあるため)

2つの状態が動作しています



水漏れが60秒以上続くとヒータが停止しています



Stateflow トレーニングサービスのご紹介

- より深くStateflowを学びたい方にお奨めです。お客様サイトでの実施も可能です。
- 自動車分野向け Stateflow 基礎コース



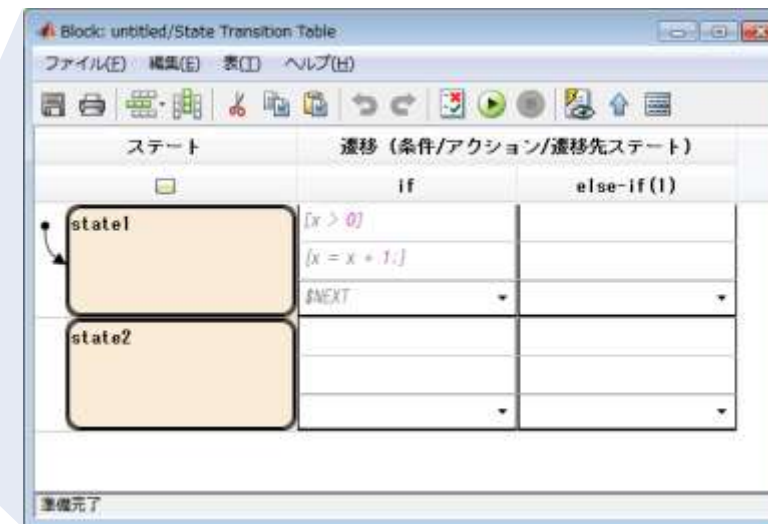
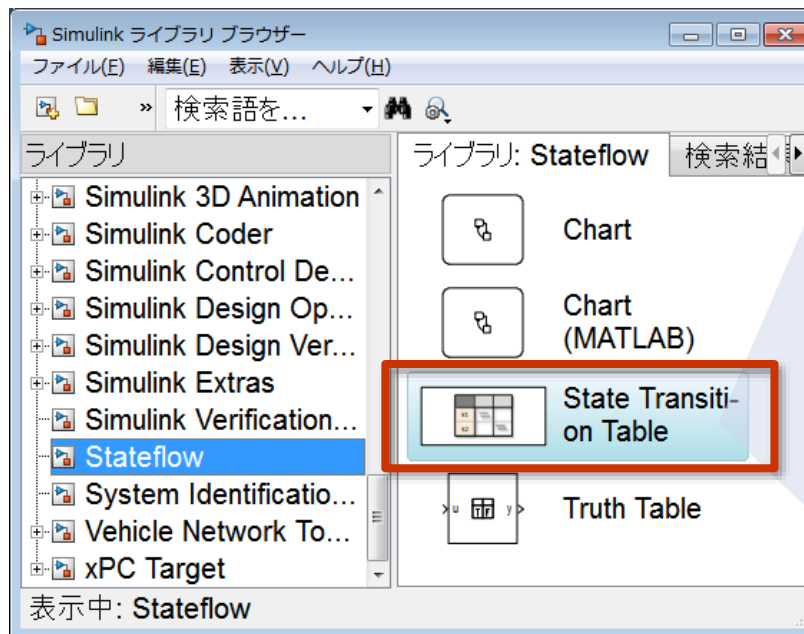
目次

- Stateflow 紹介
 - Stateflowとは
 - Stateflow基本環境
- Stateflow 演習
 - 例題1 フローチャート
 - 例題2 状態遷移図① 基本
 - 例題3 状態遷移図② 時相論理
 - 例題4 状態遷移図③ 階層化
 - 例題5 状態遷移図④ 並列状態
- 補足

状態遷移表

従来の表よりコンパクトに書ける形式を採用

- 状態 + イベント表は**状態遷移行列**と呼んで区別、表示機能を提供
- MATLAB言語で記述（MATLABチャートと同じ）
- コード生成、モデル検証等各種機能に対応



Stateflowの状態遷移表とは？ (1/2)

表編集 診断 状態遷移図表示

デフォルト遷移 状態 階層化 (子状態)

状態遷移図表示

遷移条件 遷移アクション 遷移先状態

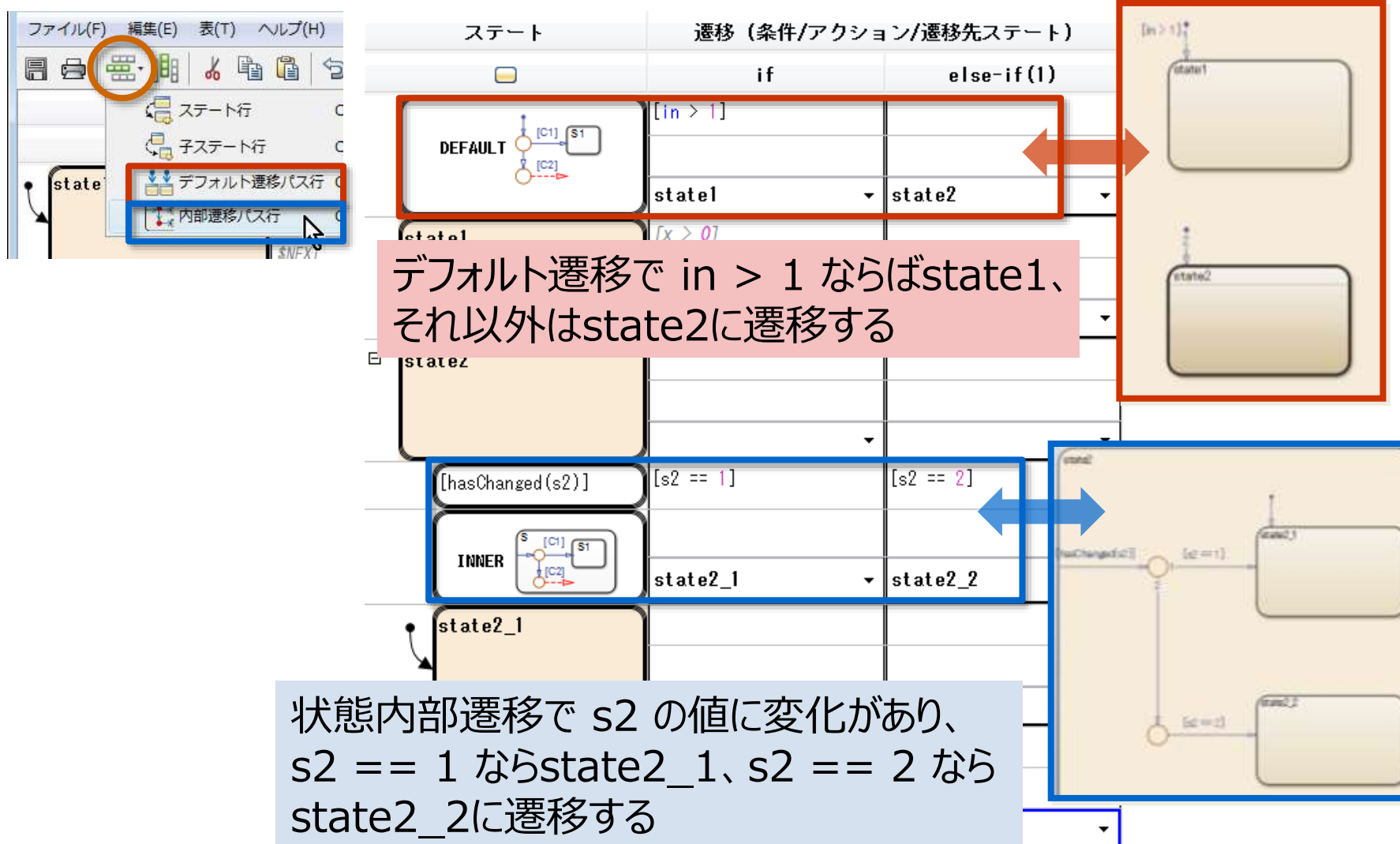
state2で $in < -1$ が成立すればstate1に遷移する

遷移先リスト

ステート	遷移 (条件/アクション/遷移先ステート)	
	if	else-if(1)
state1 en: out = 1;	[in > 1]	[in > 2]
state1_1 en, du: cnt = 0;	[flag]	[x > 0] [x = x + 1;]
state1_2 en: cnt = cnt + 1;	[~flag]	
state2 en: out = 2;	[in < -1]	
state3 en: out = 3;	[after(5, sec)]	

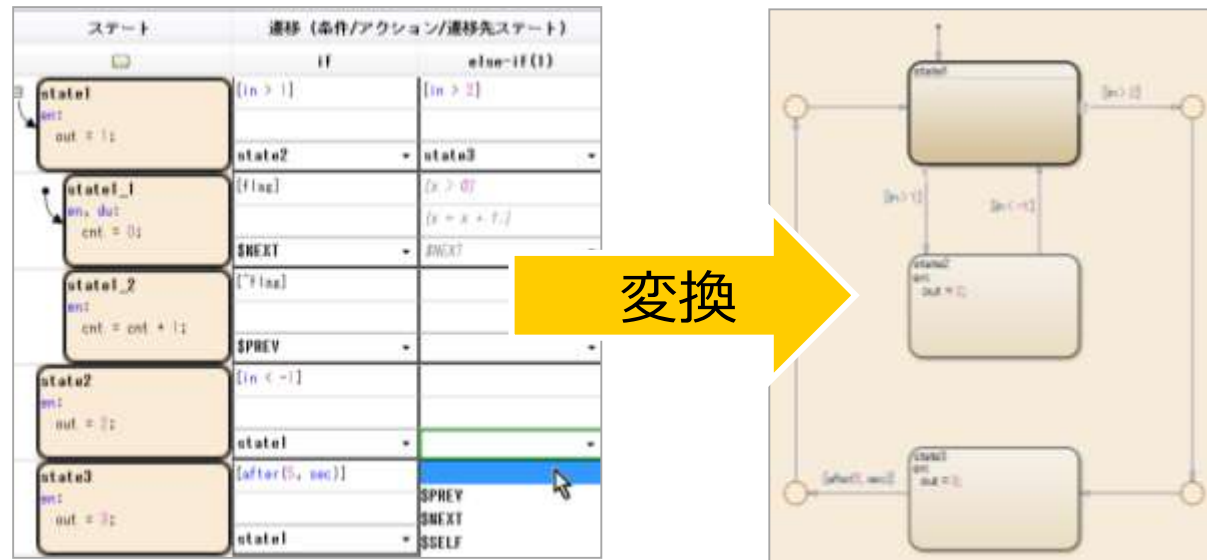
遷移先リスト: \$PREV, \$NEXT, \$SELF, state1, state3, % IGNORE %

Stateflowの状態遷移表とは？ (2/2)

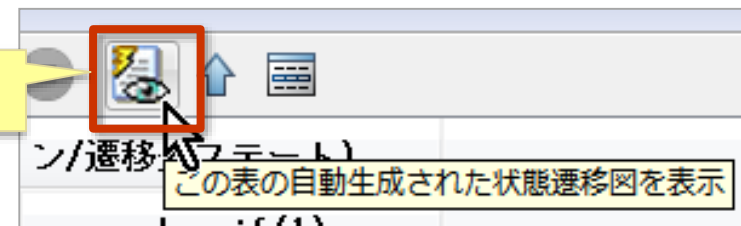


状態遷移表ブロックの動作メカニズム

- 状態遷移表をMATLABチャートに自動変換して実行
 - シミュレーション・コード生成には変換されたMATLABチャートを利用
 - R2012bでは表アニメ機能が無いのでチャートアニメで代替



状態遷移図表示





© 2018 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.