

# **MATLAB, Simulink および Stateflow モデル作成規約**

**初版：2020年12月**

**航空機開発における MBD 技術情報交換会**

**Model-Based Aviation development Consortium**

**(MBAC)**

## 目次

1. 総則 .....	1
1.1. 概要 .....	1
1.2. 対象ソフトウェアバージョン .....	1
1.3. ガイドラインについて .....	1
1.3.1. テンプレート .....	1
1.3.2. 規約 ID .....	2
1.3.3. サブ ID .....	2
1.3.4. タイトル .....	2
1.3.5. カスタムパラメーター .....	2
1.3.6. 根拠 .....	2
1.3.7. 参考規約 ID .....	2
2. Simulink/Stateflow 共通 .....	3
2.1. 命名規則 (ファイル、フォルダ) .....	3
2.1.1. a_0001 : ファイル名およびフォルダ名に使用できる文字 .....	3
2.1.2. a_0002 : ファイル名の文字数制限 .....	5
2.2. 命名規則 (モデル) .....	6
2.2.1. b_0001 : ブロック名に使用できる文字 .....	6
2.2.2. b_0002 : ブロック名の文字数制限 .....	8
2.2.3. b_0003 : 信号/バス/パラメーター名に使用できる文字 .....	9
2.2.4. c_0001 : Stateflow データ名に使用できる文字 .....	11
2.2.5. bc_0001 : 定義データ名の重複 .....	12
2.2.6. x_0001 : 未使用のデータ .....	13
2.3. その他 .....	14
2.3.1. v_0001 : モデルで使用するフォントとフォントサイズ .....	14
2.3.2. bc_0002 : 型の設定方法 .....	15
3. Simulink .....	17
3.1. コンフィギュレーションパラメーター .....	17
3.1.1. b_0004 : 論理信号に対する最適化パラメーター設定 .....	17
3.1.2. b_0005 : 整数丸めモードの設定 .....	19
3.1.3. b_0006 : 不正な演算結果の検出 .....	20
3.2. ダイアグラムの外観 .....	22
3.2.1. v_0002 : Simulink モデルの表示設定 .....	22
3.2.2. v_0003 : ブロックのサイズ調整 .....	23
3.2.3. v_0004 : ブロック名の位置 .....	24
3.2.4. v_0005 : ブロック名の表示 .....	25
3.2.5. v_0006 : ブロックパラメーターの表示 .....	26
3.2.6. v_0007 : モデルの説明 .....	27
3.2.7. v_0008 : ブロックの陰影 .....	28

3.2.8.	b_0007 : 未接続の信号/ブロック .....	29
3.2.9.	v_0009 : 信号線の結線.....	31
3.2.10.	v_0010 : Simulink モデルの信号フロー .....	34
3.2.11.	v_0011 : 構造サブシステム間の接続関係の明確化.....	37
3.2.12.	v_0012 : モデル要素の名前の一致 .....	40
3.2.13.	v_0013 : トリガー信号の名前.....	42
3.3.	信号.....	43
3.3.1.	b_0008 : ベクトル信号/バス信号の仕様 .....	43
3.4.	ブロック共通 .....	44
3.4.1.	b_0009 : インデックスの使用法 .....	44
3.4.2.	b_0010 : ブロックパラメーターの記述方法.....	45
3.4.3.	b_0011 : キャリブレーション対象の名前付き定数設定.....	47
3.4.4.	b_0012 : サンプル時間の設定.....	48
3.4.5.	b_0013 : 固定小数点設定 .....	49
3.5.	条件付きサブシステム関連.....	50
3.5.1.	b_0014 : 条件付きサブシステム内のブロック配置.....	50
3.5.2.	b_0015 : 条件付きサブシステムにおける Output ブロックの初期値設定.....	52
3.5.3.	b_0016 : Merge ブロックへ入力する信号線の使用制限.....	53
3.5.4.	b_0017 : if ブロックの使用法 .....	54
3.5.5.	b_0018 : 条件付き制御フローブロックの使用法.....	55
3.5.6.	b_0019 : 条件付き制御フローブロックと Merge ブロックによる出力値保持 .....	57
3.6.	演算系ブロック .....	59
3.6.1.	b_0020 : 論理演算と数値演算の適切な実装.....	59
3.6.2.	b_0021 : 加減算ブロックの使用法.....	61
3.6.3.	b_0022 : 乗除算ブロックの演算子順序 .....	64
3.6.4.	b_0023 : 乗除算ブロックの入力符号.....	65
3.6.5.	b_0024 : Simulink における除算 .....	66
3.6.6.	b_0025 : 数値演算ブロックの入力 .....	67
3.6.7.	b_0026 : Fcn ブロックの使用法 .....	73
3.6.8.	b_0027 : 論理演算ブロックのアイコン形状.....	74
3.6.9.	b_0028 : Relational Operator の使用法.....	75
3.6.10.	b_0029 : Simulink における浮動小数点型の比較 .....	76
3.6.11.	b_0030 : Lookup Table 系ブロックの使用法.....	77
3.6.12.	b_0031 : 離散系遅延ブロックの使い分け .....	78
3.6.13.	b_0032 : Tapped Delay ブロック/Delay ブロックの使用法 .....	79
3.6.14.	b_0033 : Discrete-TimeIntegrator ブロックの使用法.....	80
3.6.15.	b_0034 : Saturation ブロックの使用法 .....	82
3.6.16.	b_0035 : 型変換を実施する場合の使用法.....	83
3.7.	その他のブロック .....	84
3.7.1.	b_0036 : Inport ブロック / Outport ブロックの使用法 .....	84
3.7.2.	b_0037 : Inport ブロック / Outport ブロックのアイコン表示.....	86

3.7.1.	b_0038 : Goto/From の範囲.....	87
3.7.2.	b_0039 : Data Store Memory ブロックの定義方法 .....	88
3.7.3.	b_0040 : Switch ブロックの使用方法 .....	89
3.7.4.	b_0041 : 切替機能を持つブロックの入出力データ型 .....	90
3.7.5.	b_0042 : Multiport Switch ブロックの使用方法.....	91
4.	Stateflow.....	93
5.	その他.....	94
5.1.1.	d_0001 : 列挙型の規定値の定義.....	94
5.1.2.	d_0002 : コメント記号/*、*/の使用禁止 .....	95

## 1. 総則

### 1.1. 概要

航空機用制御装置のコントローラモデルにおいて、作成者、使用者およびレビュー者間の中で共通の理解が得られるようにモデルの記述方法をルール化したものである。実際の適用にあたっては、各社またはプロジェクトの特徴に応じて具体化・テーラリングを行うことを前提に、本規約では基本的なルールを示している。なお、本規約作成にあたっては、JMAAB 作成の「制御モデリングガイドライン Ver5.1」を元にテーラリングを実施して本規約を作成している。

### 1.2. 対象ソフトウェアバージョン

本規約は、MATLAB バージョン 2019b を対象とする。

### 1.3. ガイドラインについて

#### 1.3.1. テンプレート

本ガイドラインは、以下に示す形式で規約を作成している。本ガイドラインをテーラリングして規約を作成する場合においても、本テンプレートを使用することを推奨する。各項目の詳細に関しては次章以降に説明を示す。

規約 ID		
規約名		
参考規約 ID		
規約		
サブ ID	記述内容	カスタムパラメーター
根拠		
サブ ID	内容	

### 1.3.2. 規約 ID

規約 ID は、アルファベットの小文字 1 文字または 2 文字の組み合わせと 4 桁の数字をアンダースコアで繋いだものを使用する。ID は、ユニークな設定かつ永久固定とする。

本ガイドラインでは、以下のルールに従って規約 ID のアルファベットを設定している。

アルファベット	内容
a	ファイル、フォルダー関係
b	Simulink 関係
c	Stateflow 関係
d	MATLAB 関係
v	モデルの可読性
x	参考として将来に検討

2 種の内容を含む場合は、アルファベット順で 2 文字組合わせて使用する。

例：「bc\_0001」：Simulink および Stateflow に関する規約を意味する。

また、JMAAB 作成の「制御モデリングガイドライン Ver5.1」ルール ID との重複を避けるため以下のアルファベットの使用は可能な限り避ける。

**ar , db, jc , jm, na,**

### 1.3.3. サブ ID

サブ ID は、規約 ID 一つに対して複数存在し、「採否選択式」と「排他選択式」の 2 種がある。採否選択式は、アルファベットの小文字 1 文字を使用する。「採否選択式」のサブ ID の採用可否は、各社またはプロジェクトの特徴に応じて任意に決める。「排他選択式」は、アルファベットの小文字 1 文字と 1 桁の数字を繋いだものを使用し、各社またはプロジェクトの特徴に応じていずれかを選択する。

### 1.3.4. タイトル

規約内容を、簡潔に説明しているものであり、ユニークな内容を記述する。

### 1.3.5. カスタムパラメーター

カスタムパラメーター欄に記述のある規約は、記述されている項目について各社またはプロジェクトの特性に合わせて対象範囲を決定する必要がある。

### 1.3.6. 根拠

規約の記述内容について、可読性やコード効率等の観点から規約とした理由を記述する。




### 1.3.7. 参考規約 ID


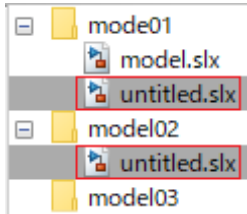
JMAAB 作成の「制御モデリングガイドライン Ver5.1」をテーラリングしているのでルール ID を参考に示している。本ガイドラインをテーラリングする場合は、本ガイドラインの規約 ID を記述する。

2. Simulink/Stateflow 共通

2.1. 命名規則(ファイル、フォルダ)

2.1.1. a\_0001: ファイル名およびフォルダ名に使用できる文字

規約 ID	a_0001	
規約名	ファイル名およびフォルダ名に使用できる文字	
参考規約 ID	ar_0001 (サブ ID : b を除く) / ar_0002	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>ファイル名およびフォルダ名で使用可能な文字の種類は、下記のみとする。</p> <ul style="list-style-type: none"> <li>・ 半角英数字</li> <li>・ 半角アンダースコア</li> </ul> <p><b>【誤例】</b></p> <p>MBAC Model.slx      半角スペースを使用している。</p> <p>MBAC-Param.m      記号を使用している。</p> <p>名前      ^</p> <p> MBACモデルファイル      全角文字を使用している。</p>	ファイル (拡張子)
b	<p>ファイル名の先頭は、英字を使用する。</p> <p><b>【誤例】</b></p> <p>1MBAC.slx</p> <p>_MBAC.slx</p>	ファイル (拡張子)
c	<p>ファイル名およびフォルダ名は、末尾にアンダースコアを使用しない。</p> <p><b>【誤例】</b></p> <p>MBAC_.slx</p> <p>名前</p> <p> MBAC_</p>	ファイル (拡張子)
d	<p>ファイル名およびフォルダ名は、連続したアンダースコアを使用しない。</p> <p><b>【誤例】</b></p> <p>MBAC__Model.slx</p> <p>名前      ^</p> <p> MBAC_Model</p>	ファイル (拡張子)

規約		
サブ ID	記述内容	カスタムパラメーター
e	<p>ファイル名およびフォルダー名は、完全一致する MATLAB 予約語にしない。</p> <p><b>【誤例】</b></p> <p>Ans.slx double.slx</p> <p>名前</p> <p> week</p>	ファイル (拡張子)
f	<p>MATLAB のパスを通したファイル名は、同一にしない。</p> <p><b>【誤例】</b></p> 	ファイル (拡張子)
根拠		
サブ ID	内容	
ae	<ul style="list-style-type: none"> <li>・可読性が損なわれる。</li> <li>・一般的な作法から外れることにより、想定外の問題が起こるリスクがある。</li> </ul>	
b	<ul style="list-style-type: none"> <li>・MATLAB にて禁止されており、保存することが出来ない。</li> </ul>	
cd	<ul style="list-style-type: none"> <li>・可読性が損なわれる。</li> </ul>	
f	<ul style="list-style-type: none"> <li>・同一名称のファイル名が複数ある場合、パスの優先順位が高い方のファイルが読み込まれる。これにより、意図しないファイルを読み込む可能性がある。</li> <li>・可読性が損なわれる。</li> <li>・一般的な作法から外れることにより、想定外の問題が起こるリスクがある。</li> </ul>	



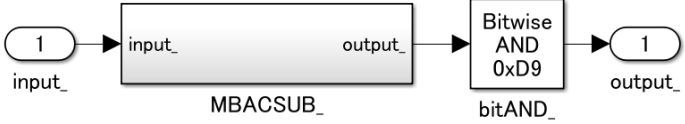
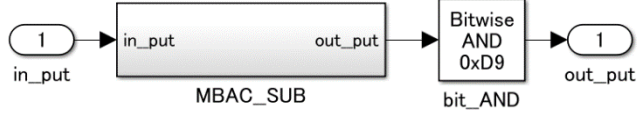

## 2.1.2. a\_0002: ファイル名の文字数制限

規約 ID	a_0002	
規約名	ファイル名の文字数制限	
参考規約 ID	jc_0241	
規約		
サブ ID	記述内容	カスタムパラメーター
a	モデルファイル名（ドットおよび拡張子を除いた部分）の最大文字数は、59 文字以内とする。	モデルファイル名の最大文字数
b	スクリプトファイル（M ファイル）名（ドットおよび拡張子を除いた部分）の最大文字数は、63 文字以内とする。	スクリプトファイル（M ファイル）名の最大文字数
根拠		
サブ ID	内容	
a	・モデルリファレンスで参照することが出来ない。	
b	・ファイルを実行することが出来ない。	


2.2. 命名規則(モデル)

2.2.1. b\_0001: ブロック名に使用できる文字

規約 ID	b_0001	
規約名	ブロック名に使用できる文字	
参考規約 ID	jc_0201 / jc_0231 / jc_0211	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>ブロック名で使用可能な文字の種類は、下記のみとする。</p> <ul style="list-style-type: none"> <li>・半角英数字</li> <li>・半角アンダースコア</li> </ul> <p><b>【誤例】</b></p> <p>半角スペースを使用している。</p> <p>全角文字を使用している。</p> <p>記号を使用している。</p>	—
b	<p>ブロック名は、先頭に数字を使用しない。</p> <p><b>【誤例】</b></p>	—
c	<p>ブロック名は、先頭にアンダースコアを使用しない。</p> <p><b>【誤例】</b></p>	—

規約		
サブ ID	記述内容	カスタムパラメーター
d	<p>ブロック名は、末尾にアンダースコアを使用しない。</p> <p>【誤例】</p> 	—
e	<p>ブロック名は、連続したアンダースコアを使用しない。</p> <p>【誤例】</p> 	—
f	<p>ブロック名は、完全一致する MATLAB 予約語にしない。</p> 	—
根拠		
サブ ID	内容	
ab	<ul style="list-style-type: none"> <li>逸脱した場合、モデルとコードの整合が取りづらくなる。</li> <li>設定した構造サブシステム名でコード生成できなくなる。</li> </ul>	
c	<ul style="list-style-type: none"> <li>逸脱した場合、可読性が損なわれる。</li> <li>設定した構造サブシステム名でコード生成できなくなる。</li> <li>MISRA-C:2004 に逸脱する可能性がある。</li> </ul>	
d	<ul style="list-style-type: none"> <li>逸脱した場合、可読性が損なわれる。</li> <li>設定した構造サブシステム名でコード生成できなくなる可能性がある。</li> </ul>	
e	<ul style="list-style-type: none"> <li>逸脱した場合、可読性が損なわれる。</li> <li>設定した構造サブシステム名でコード生成できなくなる可能性がある。</li> </ul>	
f	<ul style="list-style-type: none"> <li>逸脱した場合、可読性が損なわれる。</li> <li>設定した構造サブシステム名でコード生成できなくなる可能性がある。</li> <li>一般的な作法から外れることにより、想定外の問題が起こるリスクがある。</li> </ul>	

2.2.2. b\_0002: ブロック名の文字数制限

規約 ID	b_0002	
規約名	ブロック名の文字数制限	
参考規約 ID	jc_0243 / jc_0247 / jc_244	
規約		
サブ ID	記述内容	カスタムパラメーター
a	ブロック名の最大文字数は、63 文字以内とする。  <p>a123456789112345678921234567893123456789412345678951234567896123</p>	ブロック名の最大文字数
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>コード生成できない可能性があります。</li> </ul>	

2.2.3. b\_0003: 信号/バス/パラメーター名に使用できる文字

規約 ID	b_0003	
規約名	信号/バス/パラメーター名に使用できる文字	
参考規約 ID	jc_0222 / jc_0232	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>信号/バス/パラメーター名で使用可能な文字の種類は、下記のみとする。</p> <ul style="list-style-type: none"> <li>・ 半角英数字</li> <li>・ 半角アンダースコア</li> </ul> <p><b>【誤例】</b></p> <p>半角スペースを使用している。</p> <p>記号を使用している。</p> <p>全角文字を使用している。</p>	—
b	<p>信号/バス/パラメーター名は、先頭に数字を使用しない。</p>	—
c	<p>信号/バス/パラメーター名は、先頭にアンダースコアを使用しない。</p>	—

規約		
サブ ID	記述内容	カスタムパラメーター
d	信号／バス／パラメーター名は、末尾にアンダースコアを使用しない。	—
e	信号／バス／パラメーター名は、連続したアンダースコアを使用しない。	—
f	信号／バス／パラメーター名は、完全一致する MATLAB 予約語にしない。	—
根拠		
サブ ID	内容	
ab	<ul style="list-style-type: none"> <li>逸脱した場合、モデルとコードの整合が取りづらくなる。</li> <li>ブロックパラメーターの場合は、エラーとなり設定できない。</li> </ul>	
c	<ul style="list-style-type: none"> <li>逸脱した場合、可読性が損なわれる。</li> <li>ブロックパラメーターの場合は、エラーとなり設定できない。</li> <li>MISRA-C:2004 に逸脱する可能性がある。</li> </ul>	
d	<ul style="list-style-type: none"> <li>逸脱した場合、可読性が損なわれる。</li> <li>アンダースコアは一般的に単語の区切りとして用いるため、記述ミスのように見える。</li> </ul>	
e	<ul style="list-style-type: none"> <li>逸脱した場合、可読性が損なわれる。</li> </ul>	
f	<ul style="list-style-type: none"> <li>逸脱した場合、可読性が損なわれる。</li> <li>一般的な作法から外れることにより、想定外の問題が起こるリスクがある。</li> </ul>	

## 2.2.4. c\_0001: Stateflow データ名に使用できる文字

規約 ID	c_0001	
規約名	Stateflow データ名に使用できる文字	
参考規約 ID	jc_0795	
規約		
サブ ID	記述内容	カスタムパラメーター
a	Stateflow データ名で使用可能な文字の種類は、下記のみとする。 ・半角英数字 ・半角アンダースコア	—
b	Stateflow データ名は、先頭に数字を使用しない。	—
c	Stateflow データ名は、先頭にアンダースコアを使用しない。	—
d	Stateflow データ名は、末尾にアンダースコアを使用しない。	—
e	Stateflow データ名は、連続したアンダースコアを使用しない。	—
f	Stateflow データ名は、完全一致する MATLAB 予約語にしない。	—
根拠		
サブ ID	内容	
abcdef	<ul style="list-style-type: none"> <li>・逸脱した場合、可読性が損なわれる。</li> <li>・逸脱した場合、意図したコードとならない可能性があります。</li> </ul>	

## 2.2.5. bc\_0001: 定義データ名の重複

規約 ID	bc_0001	
規約名	定義データ名の重複	
参考規約 ID	jc_0791	
規約		
サブ ID	記述内容	カスタムパラメーター
a	ベースワークスペース、モデルワークスペース間でデータ名の重複は禁止する。	—
b	ベースワークスペース、データディクショナリ(sldd)間でデータ名の重複は禁止する。	データディクショナリの種類
c	モデルワークスペース、データディクショナリ(sldd)間でデータ名の重複は禁止する。	データディクショナリの種類
根拠		
サブ ID	内容	
abc	・データ名が重複するとモデルが意図しない動作をする可能性がある。	



## 2.2.6. x\_0001:未使用のデータ

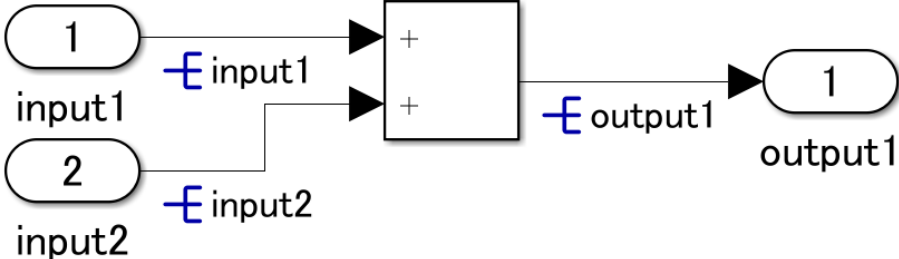
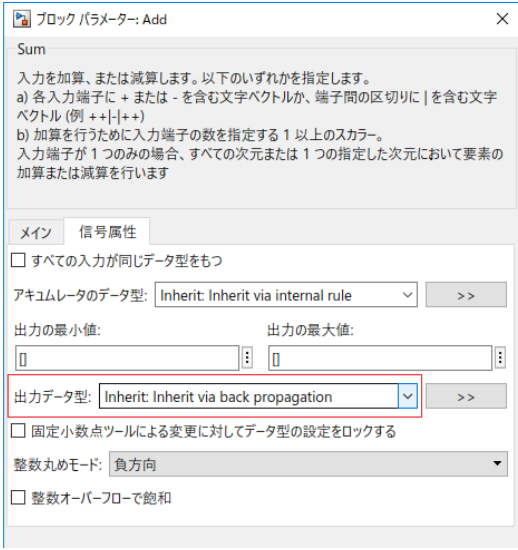

規約 ID	x_0001	
規約名	未使用のデータ	
参考規約 ID	jc_0792	
規約		
サブ ID	記述内容	カスタムパラメーター
a	モデルワークスペースには、Simulink/Stateflow モデルで使用されるデータだけを定義します。	—
b	データディクショナリ(sldd)には、Simulink/Stateflow モデルで使用されるデータだけを定義します。	データディクショナリの種類
根拠		
サブ ID	内容	
ab	<ul style="list-style-type: none"> <li>会社間でモデルを利用する際に汎用的な定義を使用する場合に未使用データを利用する可能性があり、MBAC 規約としての採否は保留として、モデル流通 WG を通して検討する。</li> </ul>	

## 2.3. その他

## 2.3.1. v\_0001: モデルで使用するフォントとフォントサイズ

規約 ID	v_0001	
規約名	モデルで使用するフォントとフォントサイズ	
参考規約 ID	db_0043	
規約		
サブ ID	記述内容	カスタムパラメーター
a	{フォント}、{フォントスタイル}をプロジェクトで定めた設定とする。	フォント フォントスタイル
b	フォントの{サイズ}をプロジェクトで定めた設定とする。	フォントサイズ
根拠		
サブ ID	内容	
ab	・フォント、フォントスタイルおよびフォントサイズを統一することで可読性が向上する。	

2.3.2. bc\_0002: 型の設定方法


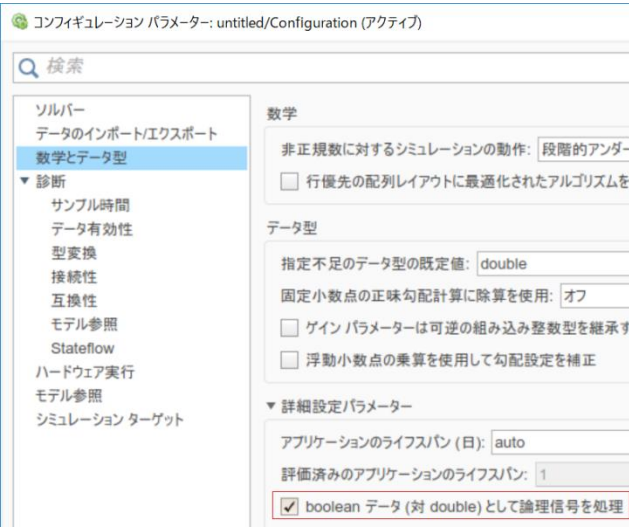
規約 ID	bc_0002	
規約名	型の設定方法	
参考規約 ID	jc_0644	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>データオブジェクトで型を設定している場合は、ブロックと Stateflow データでは型を設定しない。</p> <p><b>【例外】</b></p> <ul style="list-style-type: none"> <li>・ 再利用可能な関数内部</li> <li>・ [Data Type Conversion]</li> <li>・ "fixdt"による型設定</li> <li>・ double 型、boolean 型の指定</li> </ul> <p><b>【例】</b></p> <p>データオブジェクトで型を設定し、ブロックでは型を設定していません。</p> 	—
 		

根拠	
サブ ID	内容
a	<ul style="list-style-type: none"> <li>・ブロックの型をデータオブジェクトでの設定と個別にブロックでの設定が混在している場合、どちらの設定が正しいか判断がつかず可読性が損なわれる。</li> <li>・ブロックに型設定をすると、信号線の型が変更になったときの保守性が落ちる。</li> </ul> <p><b>【除外対象の説明】</b></p> <ul style="list-style-type: none"> <li>・再利用可能な関数内部            中のブロック構成がすべて同一でも入出力の型が異なれば C ソースが別々にできてしまい、再利用にならない。再利用可能な関数は、少なくとも入出力ブロックの型を固定する。</li> <li>・[Data Type Conversion]            [Data Type Conversion]は、型を埋め込む為に作られたブロックである。必要に応じてこのブロックを使って型の変更を明示的に指定する。</li> <li>・”fixdt”による型設定            固定小数点を指定した場合、ブロック毎に桁が変わるので、細かい指定が必要である。データオブジェクトだけではコントロールできない。</li> <li>・double 型、boolean 型の指定            boolean 型は、ブロック内部で直接指定しなければならないブロック種別が存在する。double 型は通常プラントモデルや RCP で使われる。それらは対象外とする。組み込みソフトの内部で double 型を使用する場合もあるが、そのような環境は特殊である。double 型の使用箇所を最小とする必要があり、ブロック毎に細かく指定する。</li> </ul>

3. Simulink

3.1. コンフィギュレーションパラメーター

3.1.1. b\_0004: 論理信号に対する最適化パラメーター設定

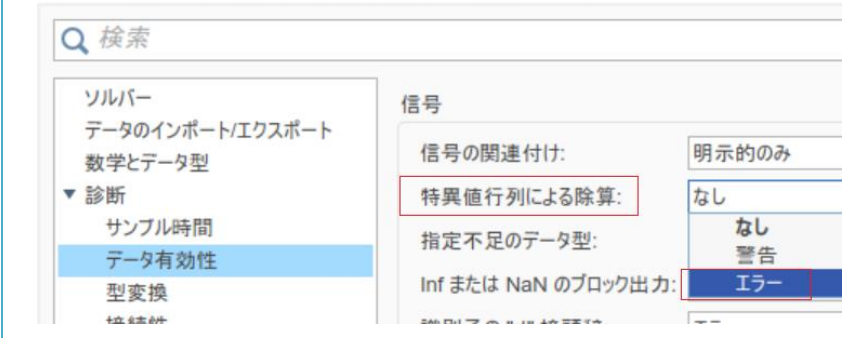
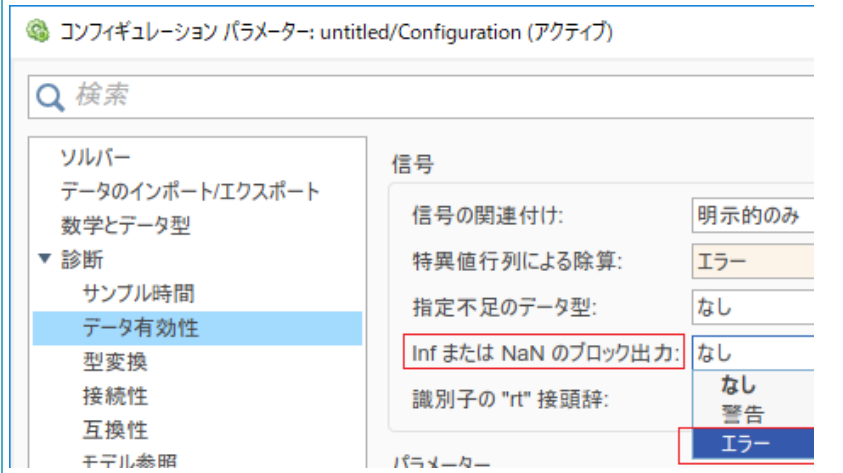
規約 ID	b_0004	
規約名	論理信号に対する最適化パラメーター設定	
参考規約 ID	jc_0011	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>論理信号に対する最適化パラメーターは無効にする。                      「コンフィギュレーションパラメーター」の{数学とデータ型} - {詳細設定パラメーター} - {boolean データ(対 double)}として論理信号を処理}のチェックを外す。</p>	—
	<p><b>【例】</b></p>  <p>このスクリーンショットは、Simulink の「コンフィギュレーション パラメーター」ウィンドウの「数学とデータ型」セクションを示しています。左側のナビゲーションメニューには「診断」が展開されており、「boolean データ(対 double)として論理信号を処理」の項目がリストアップされています。この項目のチェックボックスはオフ（未チェック）の状態です。</p>	
	<p><b>【誤例】</b></p> <p>デフォルト設定で、有効になっている。(チェックあり)</p>  <p>このスクリーンショットは、Simulink の「コンフィギュレーション パラメーター」ウィンドウの「数学とデータ型」セクションを示しています。左側のナビゲーションメニューには「診断」が展開されており、「boolean データ(対 double)として論理信号を処理」の項目がリストアップされています。この項目のチェックボックスはオン（チェックあり）の状態です。</p>	

根拠	
サブ ID	内容
a	<ul style="list-style-type: none"><li>自動的に型変換が実施されることで、意図していない動作などが生じた場合の問題特定を困難とする。なお、JMAAB ではコード生成時の RAM 容量削減の目的で、本設定を有効としている。</li></ul>


3.1.2. b\_0005: 整数丸めモードの設定

規約 ID	b_0005	
規約名	整数丸めモードの設定	
参考規約 ID	jc_0642	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>符号付き整数の除算の丸めは「未定義」以外に設定する。 「コンフィギュレーションパラメーター」の{ハードウェア実行}・{デバイスの詳細}・{符号付き整数の除算の丸め}</p> <p><b>【例】</b></p>  <p><b>【注意】</b> 各ブロックで個別に整数丸めを設定している場合は、個別設定が優先される。</p>	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>除算結果が意図しない丸めになることを防止する。</li> </ul>	

3.1.3. b\_0006: 不正な演算結果の検出

規約 ID	b_0006	
規約名	不正な演算結果の検出	
参考規約 ID	jc_0806	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>コンフィギュレーションパラメーターの{診断} - {データ有効性} - {信号} - {特異値行列による除算}を「エラー」に設定する。</p> <p><b>【例】</b></p> 	—
b	<p>コンフィギュレーションパラメーターの{診断} - {データ有効性} - {信号} - {Inf または NaN のブロック出力}を「エラー」に設定する。</p> 	—



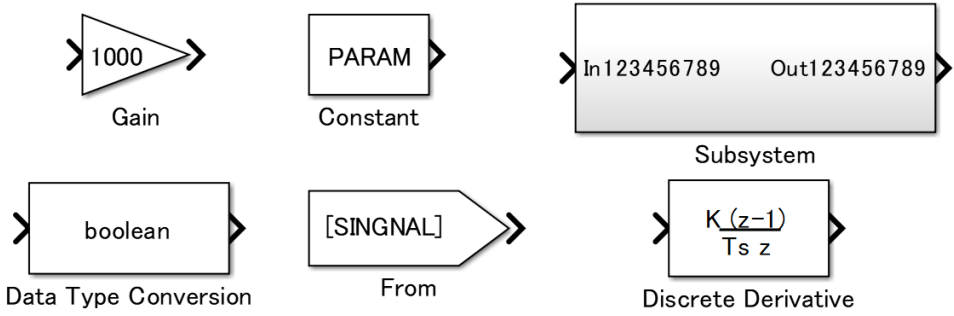
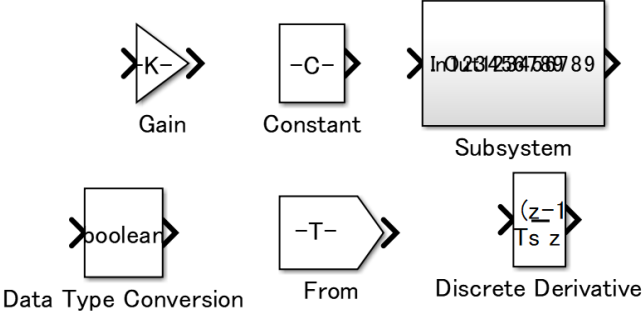
規約		
サブ ID	記述内容	カスタムパラメーター
c	<p>コンフィギュレーションパラメーターの{診断} - {データ有効性} - {パラメーター} - {オーバーフローの検出}を「エラー」に設定する。</p>	—
		
d	<p>コンフィギュレーションパラメーターの{診断} - {データ有効性} - {信号} - {オーバーフロー時にラップ}を「エラー」に設定する。</p>	—
		
根拠		
サブ ID	内容	
abcd	<ul style="list-style-type: none"> <li>不正な値による演算を検出できる。</li> </ul>	

## 3.2. ダイアグラムの外観

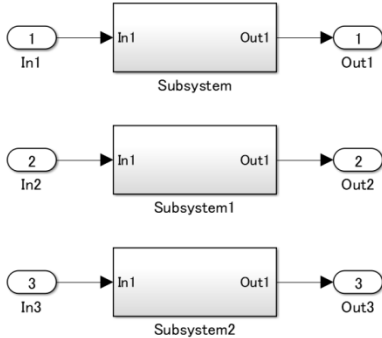
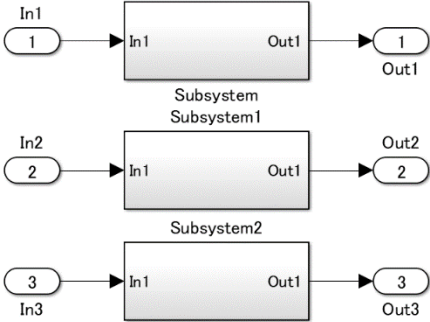
## 3.2.1. v\_0002: Simulink モデルの表示設定

規約 ID	v_0002	
規約名	Simulink モデルの表示設定	
参考規約 ID	na_0004	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>Simulink モデルの表示オプションは、プロジェクトで定めた設定にします。</p> <p><b>【例】</b></p> <p>表示オプション</p> <ul style="list-style-type: none"> <li>・モデルブラウザー：無効</li> <li>・ツールバー：有効</li> </ul> <p>ブロック表示オプション</p> <ul style="list-style-type: none"> <li>・背景色：青</li> <li>・前景色：黒</li> </ul>	表示オプション
根拠		
サブ ID	内容	
a	・モデルの表示設定を統一することで、可読性が向上する。	

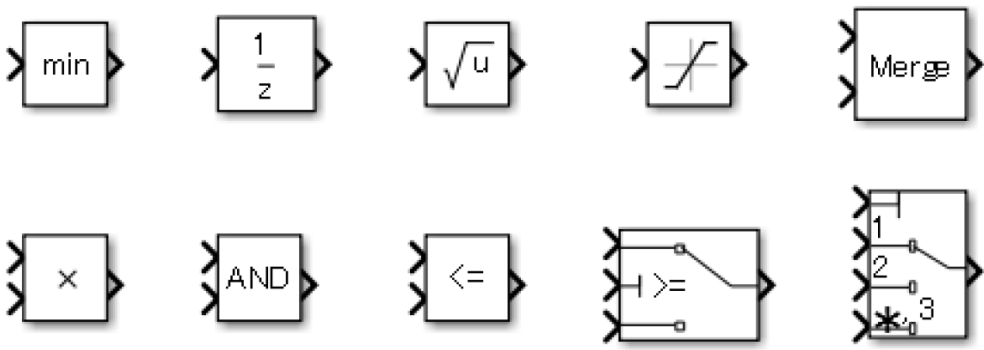
3.2.2. v\_0003: ブロックのサイズ調整

規約 ID	v_0003	
規約名	ブロックのサイズ調整	
参考規約 ID	jm_0002	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>ブロックはブロック内のアイコン表示が確認できるサイズに調整する。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—
根拠		
サブ ID	内容	
a	<p>・ブロックサイズが小さすぎると、内部にアイコン表示されるテキスト等が見えなくなるため可読性が損なわれる。</p>	

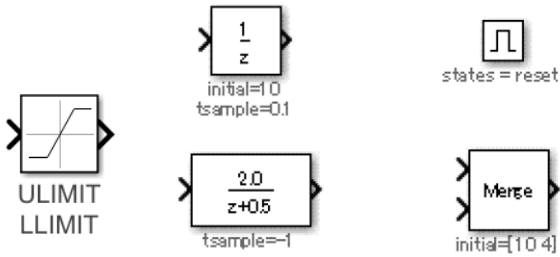
3.2.3. v\_0004: ブロック名の位置

規約 ID	v_0004	
規約名	ブロック名の位置	
参考規約 ID	db_0142	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>ブロック名はブロックの下側に表示する。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・上下にブロックが並ぶ場合、ブロック名の位置を上下で混在させると、どちらのブロック名であるかが判断できなくなる場合がある。</li> </ul>	

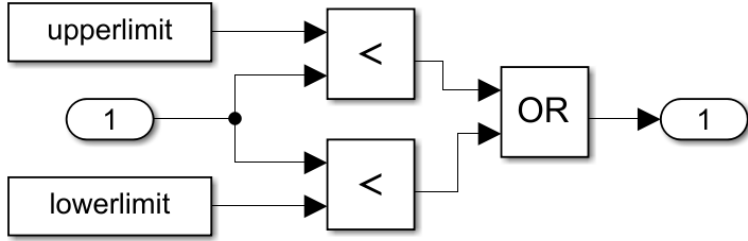
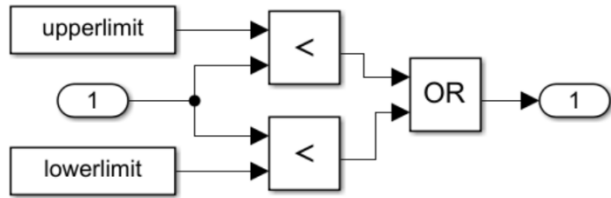
## 3.2.4. v\_0005: ブロック名の表示

規約 ID	v_0005	
規約名	ブロック名の表示	
参考規約 ID	jc_0061	
規約		
サブ ID	記述内容	カスタムパラメーター
a	次に示すブロックのブロック名は非表示にする。	外観から演算内容が分かるブロック
	<p>【例】</p> <p>カスタムパラメーターにて設定したブロックは名称を表示させない。</p> 	
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・ブロックの外観からブロックの種類が判別できる。</li> <li>・可読性を向上させる。</li> </ul>	

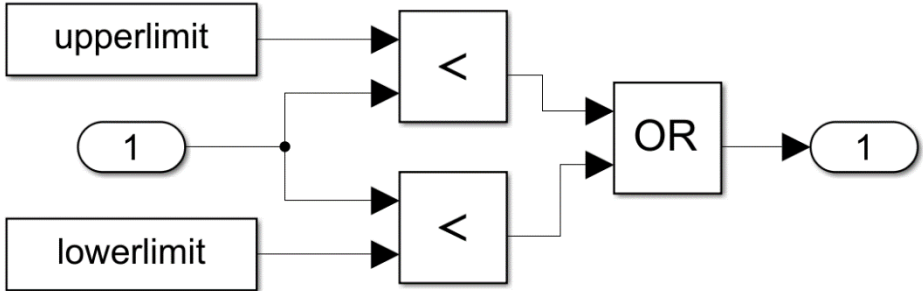
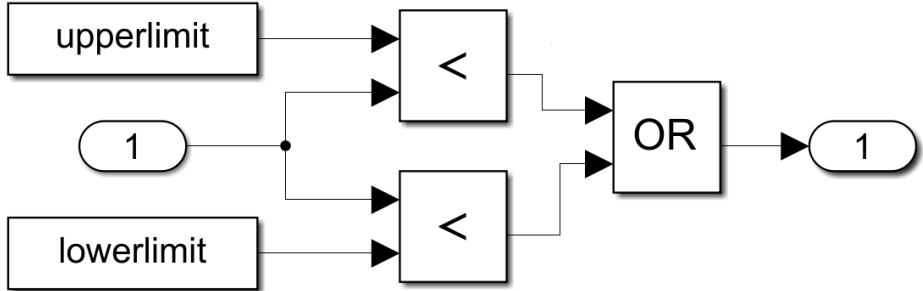
3.2.5. v\_0006: ブロックパラメーターの表示

規約 ID	v_0006	
規約名	ブロックパラメーターの表示	
参考規約 ID	db_0140	
規約		
サブ ID	記述内容	カスタムパラメーター
a	次に示すブロックはブロック注釈にてブロックパラメーターを表示する。	ブロックパラメーター
	<p><b>【例】</b>                      カスタムパラメーターで設定したブロックはパラメーターを表示させる。</p> 	
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>設定されているブロックパラメーターを外観で確認できることで、可読性を向上させる。</li> </ul>	

3.2.6. v\_0007: モデルの説明

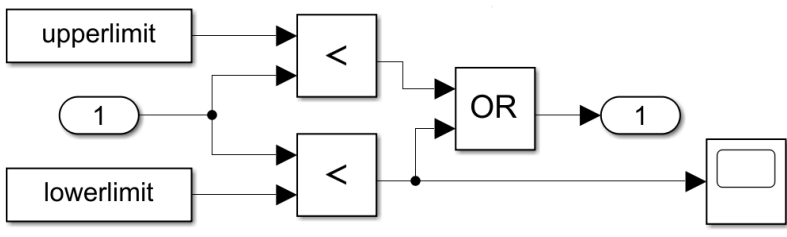
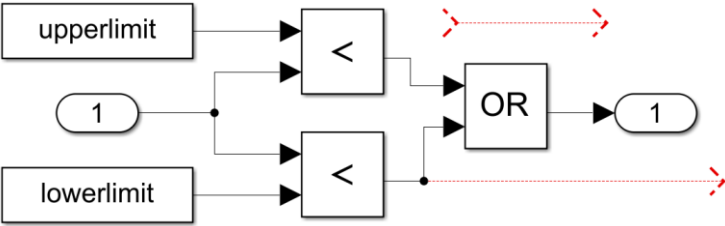
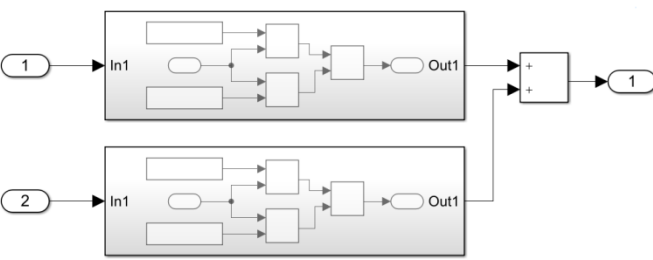
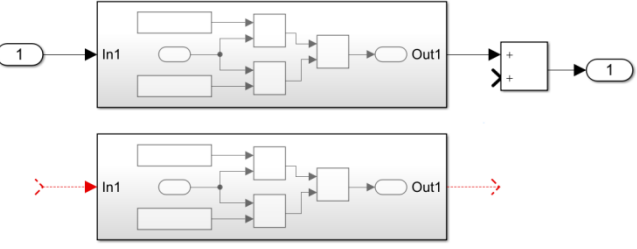
規約 ID	v_0007									
規約名	モデルの説明									
参考規約 ID	jc_0603									
規約										
サブ ID	記述内容	カスタムパラメーター								
a	<p>モデルのレイヤには、そのレイヤの説明を記載する。 説明を記載するレイヤはプロジェクトで定める。</p> <p>演算周期: 100msec 処理概要: 入力信号に対して上下限の超過を検査して、超過している場合エラーフラグを出力する。</p>  <p>表形式の注釈例</p>  <table border="1" data-bbox="451 1429 1238 1624"> <tr> <td>所属</td> <td>株式会社ABCD</td> </tr> <tr> <td>作成者</td> <td>坂井 俊哉</td> </tr> <tr> <td>演算周期</td> <td>100[msec]</td> </tr> <tr> <td>処理概要</td> <td>入力信号に対して上下限の超過を検査して、超過している場合エラーフラグを出力する。</td> </tr> </table>	所属	株式会社ABCD	作成者	坂井 俊哉	演算周期	100[msec]	処理概要	入力信号に対して上下限の超過を検査して、超過している場合エラーフラグを出力する。	説明記載オブジェクト 説明記載対象レイヤ
所属	株式会社ABCD									
作成者	坂井 俊哉									
演算周期	100[msec]									
処理概要	入力信号に対して上下限の超過を検査して、超過している場合エラーフラグを出力する。									
b	レイヤの説明はモデル全体で共通のフォーマットを使用します。	モデル説明のフォーマット								
根拠										
サブ ID	内容									
a	・説明が無いと可読性が低下し、使用性、保守性および移植性が低下する。									
b	・統一のフォーマットを使用することで可読性が向上する。									

3.2.7. v\_0008: ブロックの陰影

規約 ID	v_0008	
規約名	ブロックの陰影	
参考規約 ID	jc_0604	
規約		
サブ ID	記述内容	カスタムパラメーター
a	ブロックに陰影をつけない。 【例】  【誤例】 陰影が付いている。 	—
根拠		
サブ ID	内容	
a	・ 陰影によりポートの有無の識別が困難になり可読性が低下する。	



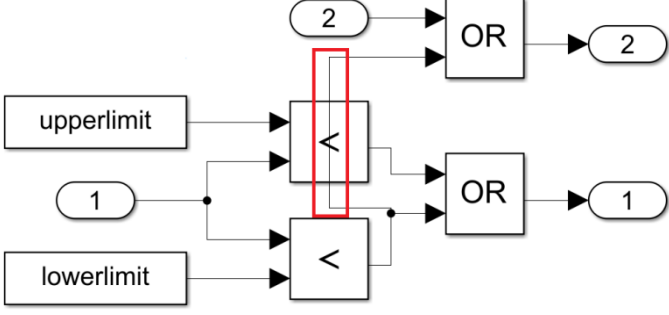
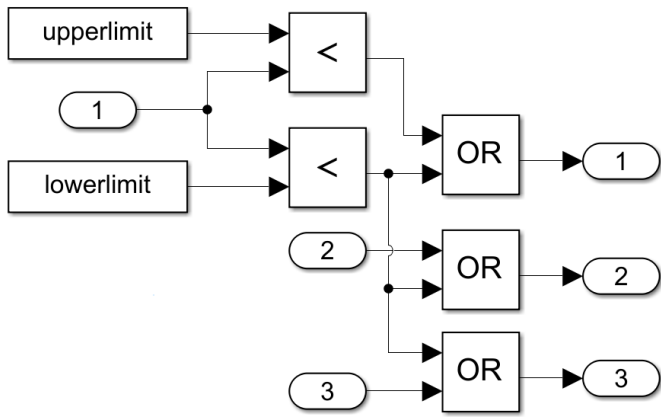
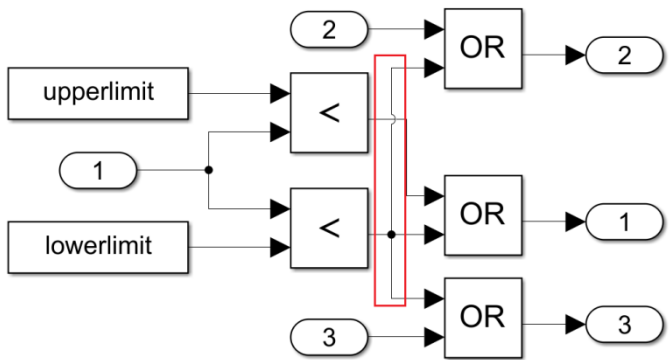
3.2.8. b\_0007: 未接続の信号/ブロック

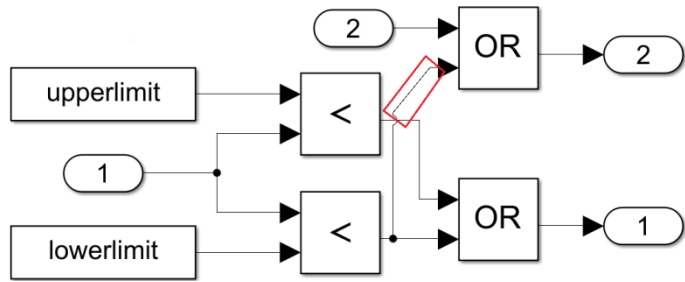
規約 ID	b_0007	
規約名	未接続の信号/ブロック	
参考規約 ID	db_0081	
規約		
サブ ID	記述内容	カスタムパラメーター
a	未接続の信号線を含めない。  【例】   【誤例】 	—
b	未接続のサブシステム、ブロックを含めない。  【例】   【誤例】 	—

根拠	
サブ ID	内容
ab	<ul style="list-style-type: none"><li>・シミュレーション結果の誤判定を生じる可能性がある。</li><li>・コード生成できない可能性がある。</li></ul>

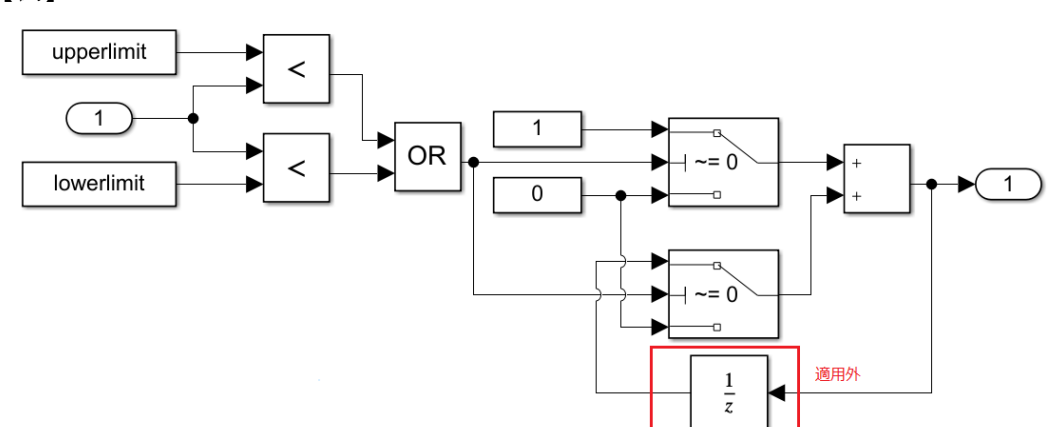
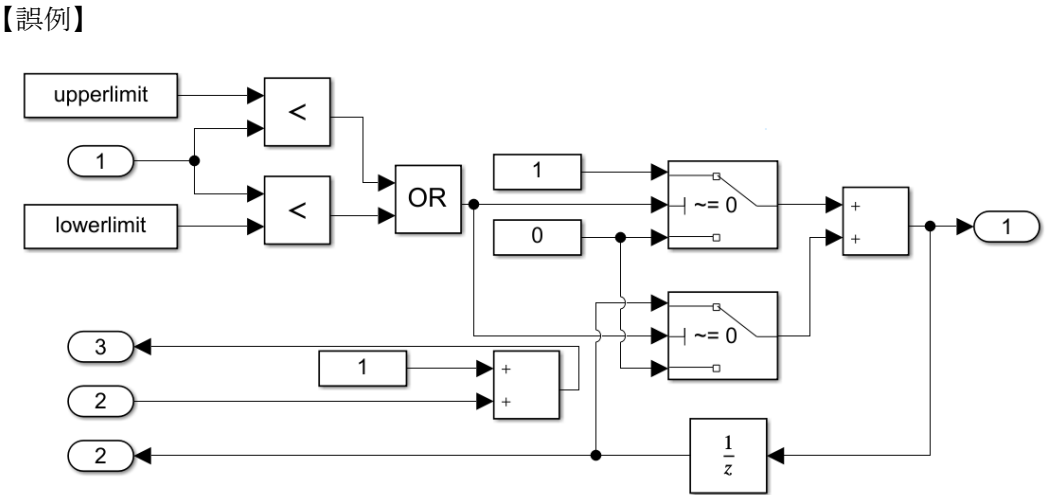
3.2.9. v\_0009: 信号線の結線

規約 ID	v_0009	
規約名	信号線の結線	
参考規約 ID	db_0032	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>{ライン交差のスタイル}を「飛び越し」に設定する。</p> <p>【例】</p>	—
b	<p>信号線は、他の信号線と重ねない。</p> <p>【例】</p> <p>【誤例】 信号線が重なっている。</p>	—

規約		
サブ ID	記述内容	カスタムパラメーター
c	<p>信号線は、ブロックを横切らない。</p> <p>【誤例】</p> 	—
d	<p>信号線は、1つの分岐点で2つ以上の分岐をしない。</p> <p>【例】</p>  <p>【誤例】</p> 	—

規約		
サブ ID	記述内容	カスタムパラメーター
e	<p>信号線は、垂直方向または水平方向に引く。</p> <p>【誤例】</p> 	—
根拠		
サブ ID	内容	
a	・ 交差と分岐の違いを明確にし、可読性を向上させる。	
bd	・ ブロック間の接続関係が分かりにくいため、可読性が損なわれる。	
ce	・ 可読性が損なわれる。	

3.2.10. v\_0010: Simulink モデルの信号フロー

規約 ID	v_0010	
規約名	Simulink モデルの信号フロー	
参考規約 ID	db_0141 / jc_0110	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>信号は、左から右に流す。</p> <p><b>【適用外】</b> フィードバックループおよび前回値は右から左に流しても良い。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—

規約

サブ ID

記述内容

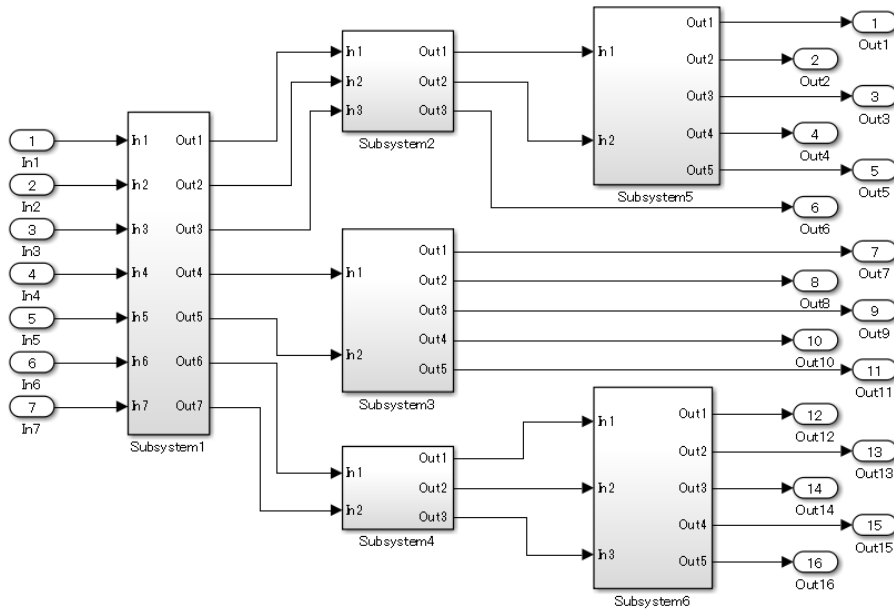
カスタムパラメーター

b

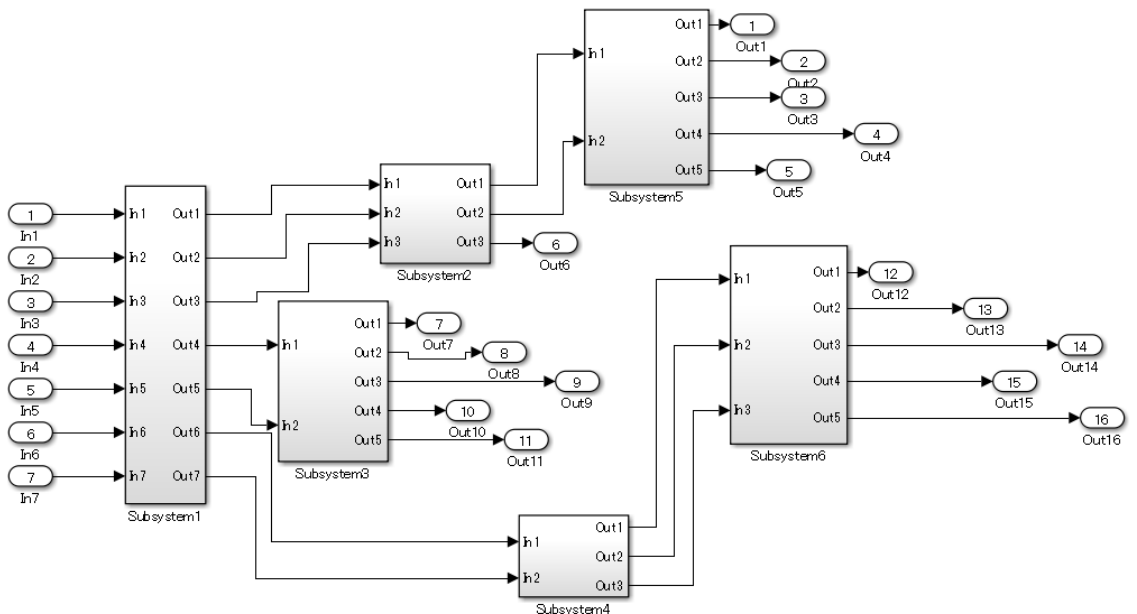
並列処理のブロックまたはサブシステムは上下に配置します。

並列処理の定義

【例】



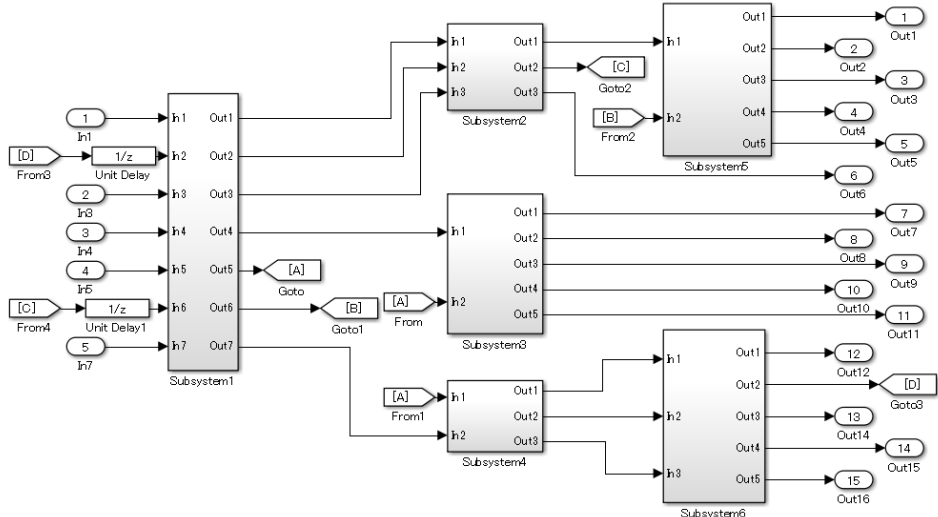
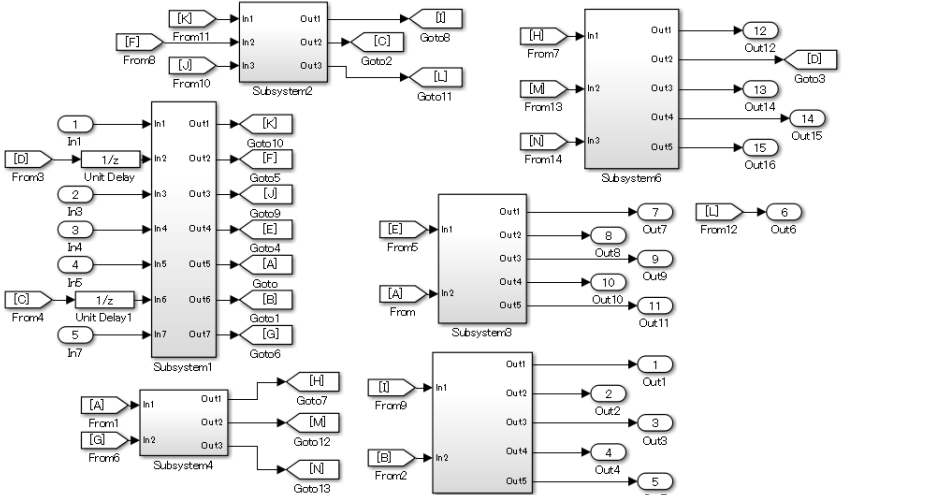
【誤例】



規約		
サブ ID	記述内容	カスタムパラメーター
c	信号線は、不必要に折り曲げない。	信号線の折り曲げ回数
	<p>【誤例】</p>	
根拠		
サブ ID	内容	
abc	・可読性が向上する。	



3.2.11. v\_0011: 構造サブシステム間の接続関係の明確化

規約 ID	v_0011	
規約名	構造サブシステム間の接続関係の明確化	
参考規約 ID	jc_0171	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>・サブシステムからサブシステムへ信号線を接続する時は、1本は必ず信号線で接続する。</p> <p><b>【適用外】</b> サブシステムからサブシステムへ[Bus Creator]、[Merge]を経由して信号を接続する場合、これらのブロックに[Goto]、[From]のみで接続しても良い。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—

規約

サブ ID

記述内容

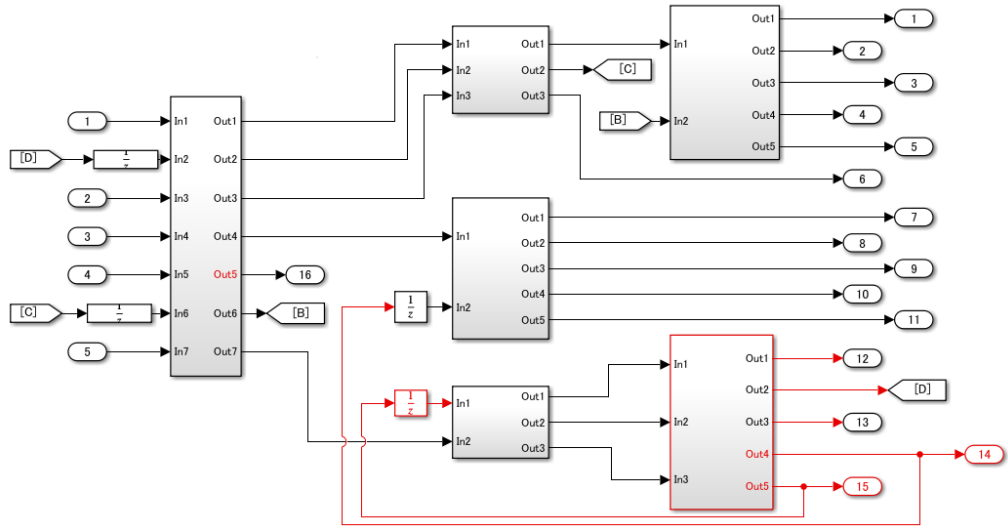
カスタムパラメーター

b

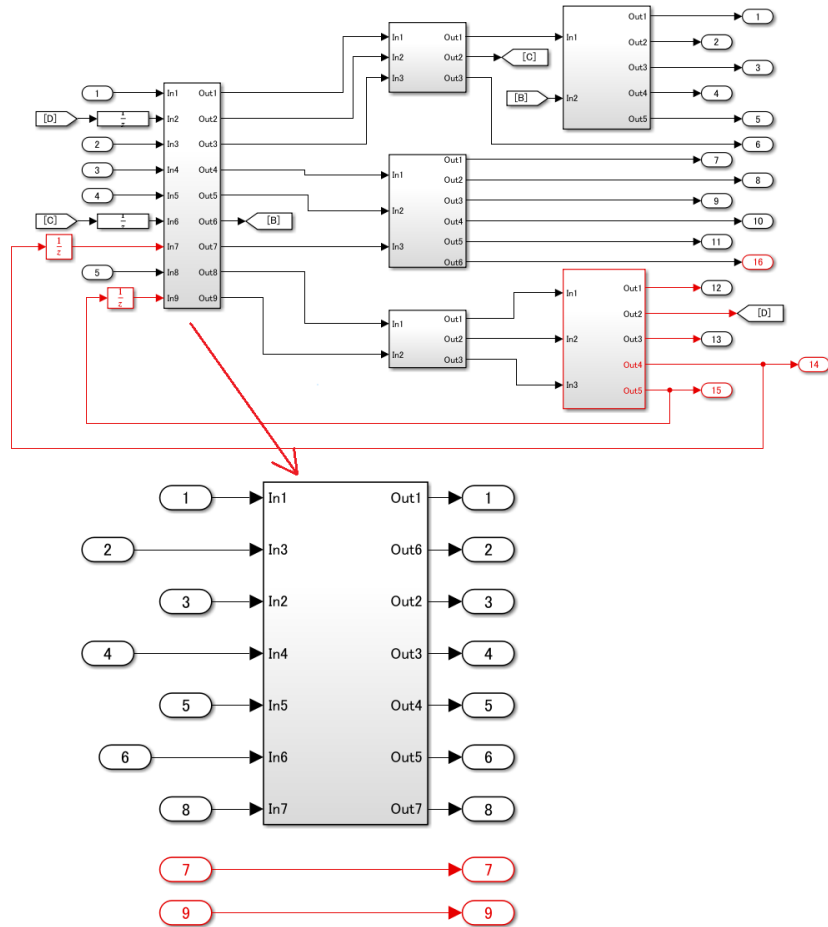
・サブシステム内部で使用しない信号をサブシステムへ入力してそのまま出力しない。

—

【例】



【誤例】 信号線の交差を避けるために、サブシステム内で処理せずに出力している。



根拠	
サブ ID	内容
a	<ul style="list-style-type: none"><li>・サブシステムの接続関係や実行順序が分かりにくいいため、可読性が損なわれる。</li></ul>
b	<ul style="list-style-type: none"><li>・不要な接続を減らすことで、可読性が向上する。</li><li>・サブシステムで未使用の入出力信号が使用されているように誤解され、可読性が損なわれる。</li><li>・不要な入出力がコード化される。</li></ul>

## 3.2.12. v\_0012: モデル要素の名前の一致

規約 ID	v_0012	
規約名	モデル要素の名前の一致	
参考規約 ID	jc_0602	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>信号線で直接接続された以下の名前は一致させる。</p> <ul style="list-style-type: none"> <li>• [Inport]のブロック名</li> <li>• [Outport]のブロック名</li> <li>• 構造サブシステムの入力ポートラベル名</li> <li>• 構造サブシステムの出力ポートラベル名</li> <li>• [From]のタグ名</li> <li>• [Goto]のタグ名</li> <li>• 信号線の信号名</li> </ul> <p>&lt;適用外 1&gt;</p> <p>以下のサブシステムと接続する信号線の信号名には例外的にサブシステムのポートラベル名と異なる名前を設定する。</p> <ul style="list-style-type: none"> <li>• ライブラリにリンクされているサブシステム</li> <li>• 再利用可能なサブシステム</li> </ul> <p>&lt;適用外 2&gt;</p> <p>[Inport]、[Outport]、その他のブロックの組み合わせが同じブロック名の場合は、[Inport]と [Outport]にサフィックスまたはプレフィックスを使用する。([Inport]、[Outport]のブロック名と信号名に異なる名前を設定する)。端子に使用するサフィックス、プレフィックスは任意ですが、一貫したサフィックス、プレフィックスを付与する。([Inport]では” in”、[Outport]では” out”等)</p>	—

規約

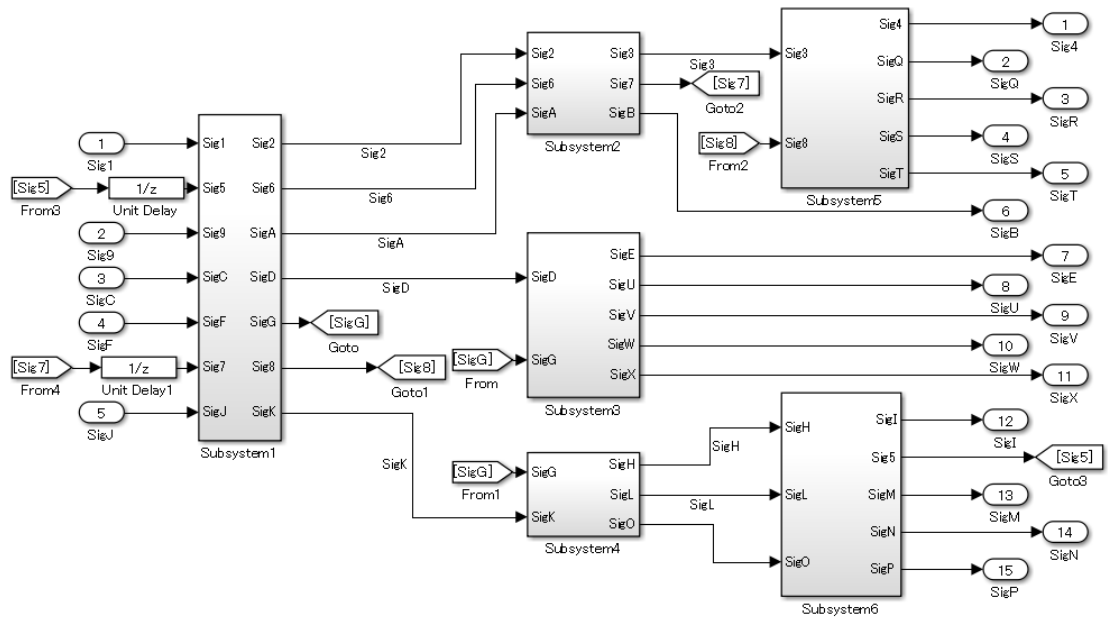
サブ ID

記述内容

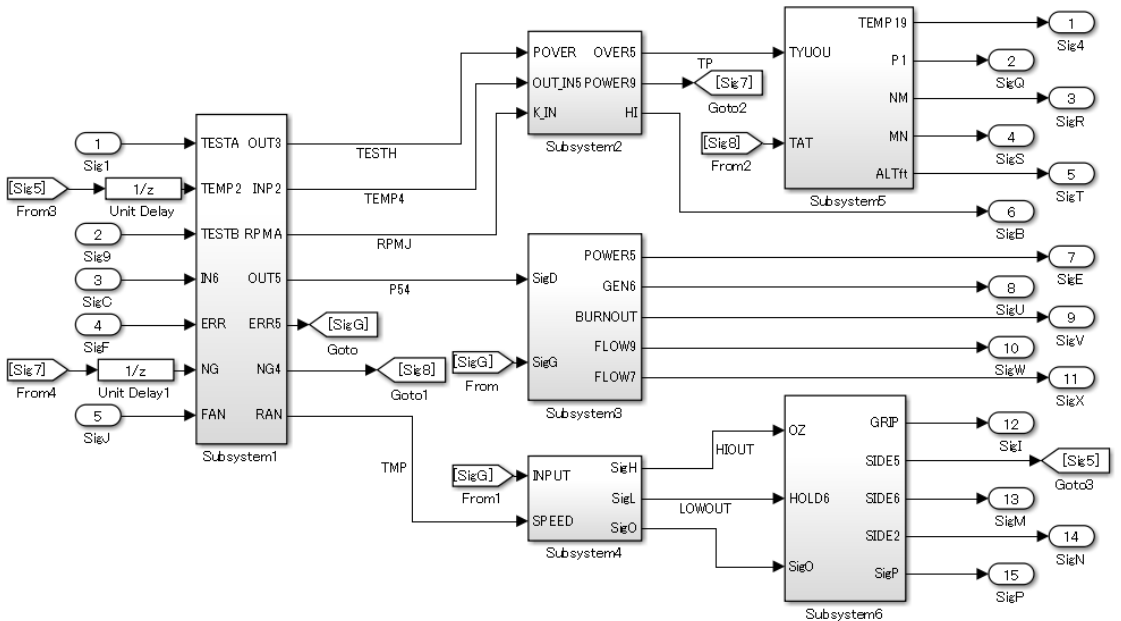
カスタムパラメーター

a

【例】



【誤例】



根拠

サブ ID

内容

a

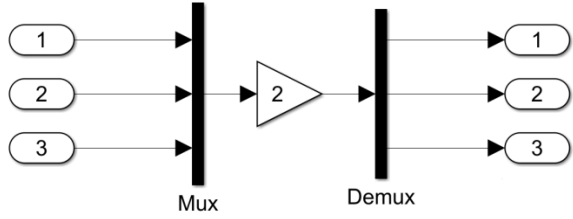
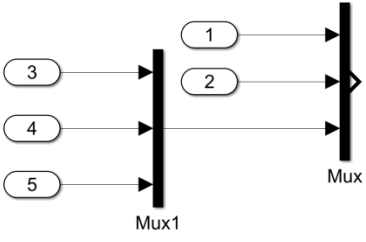
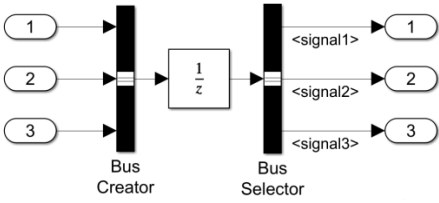
- ・ 信号線の接続ミスを防止する。
- ・ 可読性が損なわれる。

3.2.13. v\_0013:トリガー信号の名前

規約 ID	v_0013	
規約名	トリガー信号の名前	
参考規約 ID	jc_0281	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>トリガー信号の信号名またはブロック名と接続先の条件入力ブロック名または同じ名称が含まれるようにする。</p> <p>【例】</p>	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>トリガー信号の接続元と接続先の関連が分かりやすくなり、接続ミスを低減できる。</li> </ul>	

3.3. 信号

3.3.1. b\_0008: ベクトル信号／バス信号の仕様

規約 ID	b_0008	
規約名	ベクトル信号／バス信号の仕様	
参考規約 ID	na_0010	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>ベクトルの生成、分解には[Mux]、[Demux]を使用する。</p> <p>【例】</p> 	—
b	<p>[Mux]の入力はスカラー、ベクトルにする。</p> <p>【例】</p> 	—
c	<p>バスの生成、分解には[BusCreator]、[BusSelector]を使用する。</p> <p>【例】</p> 	—
d	<p>バスはバス対応ブロックに接続する。</p> <p>対応ブロックは、ヘルプの「バス対応ブロック」を参照</p>	—
根拠		
サブ ID	内容	
abcd	<ul style="list-style-type: none"> <li>ベクトル／バスの混在問題を回避できる。</li> </ul>	

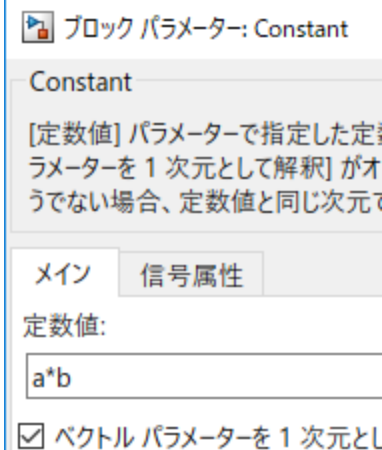
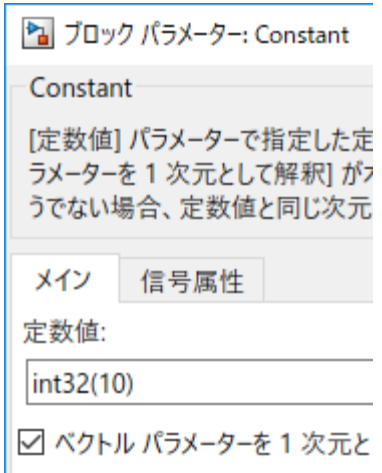
3.4. ブロック共通

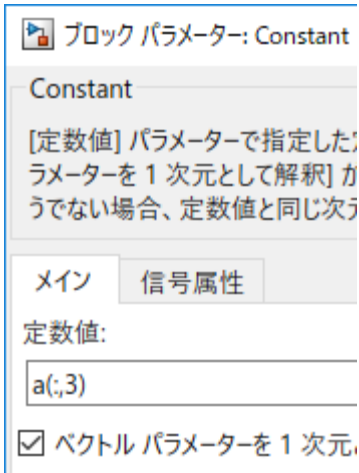
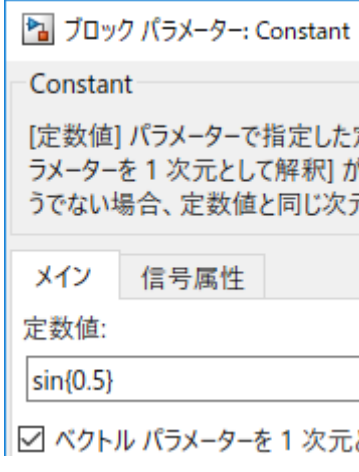
3.4.1. b\_0009: インデックスの使用方法

規約 ID	b_0009	
規約名	インデックスの使用方法	
参考規約 ID	db_0112	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>ベクトル信号の{インデックス モード}は”0 ベース”に統一する。</p> <p><b>【例】</b></p> <p><b>【誤例】</b></p> <p>インデックスが「0」に統一されていない。</p>	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・統一のインデックスを使用することで可読性が向上する。</li> <li>・C 言語の配列インデックスに合わせることで可読性が向上する。</li> </ul>	

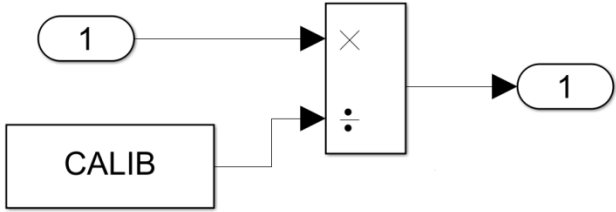
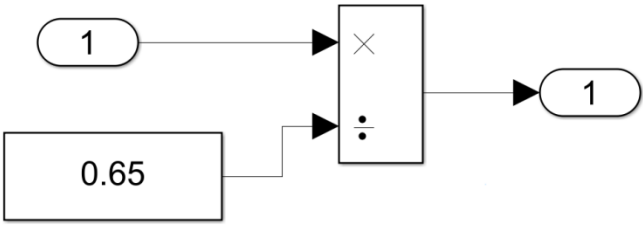


## 3.4.2. b\_0010: ブロックパラメーターの記述方法

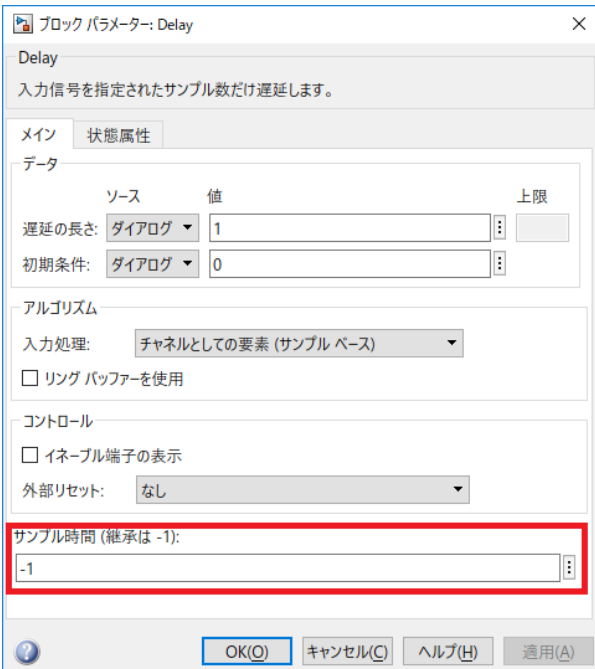
規約 ID	b_0010	
規約名	ブロックパラメーターの記述方法	
参考規約 ID	db_0110	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>ブロックパラメーターにおいて以下の記述をしません。</p> <ul style="list-style-type: none"> <li>・ 演算式の利用</li> <li>・ データ型の変換</li> <li>・ 行または列の選択</li> <li>・ MATLAB コマンド</li> </ul> <p>【誤例】</p> <ul style="list-style-type: none"> <li>・ 演算式の利用</li> </ul>  <ul style="list-style-type: none"> <li>・ データ型の変換</li> </ul> 	—

規約		
サブ ID	記述内容	カスタムパラメーター
a	<p><b>【誤例】</b></p> <ul style="list-style-type: none"> <li>行または列の選択</li> </ul>  <ul style="list-style-type: none"> <li>MATLAB コマンド</li> </ul> 	
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>演算、データ型変換、行または列の選択を記述すると、自動生成されたコード上ではマジックナンバーとなるので、モデルとコードの整合がとりづらくなる。また、パラメーターのキャリブレーションが出来なくなる。</li> <li>ブロック内に計算式を記述することで可読性が低下する。</li> <li>MATLABコマンドを記述するとコマンド実行結果がコードに反映されるため、モデルとコードの整合が取りづらくなる。</li> </ul>	


3.4.3. b\_0011: キャリブレーション対象の名前付き定数設定

規約 ID	b_0011	
規約名	キャリブレーション対象の名前付き定数設定	
参考規約 ID	jc_0645	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>キャリブレーション対象のブロックパラメーターは、名前付き定数として定義する。</p> <p>キャリブレーション対象外の例</p> <ul style="list-style-type: none"> <li>・初期値パラメーターの 0</li> <li>・インクリメント・デクリメントの 1</li> </ul> <p>【例】</p>  <p>【誤例】</p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・モデルで直値にするとコードも直値になるためキャリブレーション出来なくなる。</li> </ul>	

3.4.4. b\_0012: サンプル時間の設定

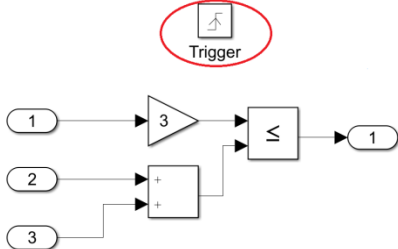
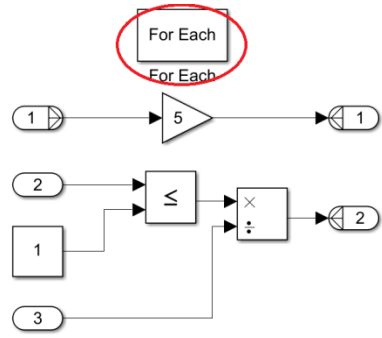
規約 ID	b_0012	
規約名	サンプル時間の設定	
参考規約 ID	jc_0641	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>ブロックの{サンプル時間}は” -1” (継承)に設定する。</p> <p>&lt;例外&gt;</p> <ul style="list-style-type: none"> <li>• [Inport]、[Outport]</li> <li>• [Atomic Subsystem]</li> <li>• [Unit Delay]、[Memory]等の状態変数を持つブロック</li> <li>• [DataTypeConversion]、[Rate Transition]等の信号変換ブロック</li> <li>• [Constant]等の外部入力を持たないブロック</li> <li>• [Chart]</li> </ul>	—
	<p><b>【例】</b></p> 	
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>• シミュレーション時間の違いによる処理結果の相違が起きる可能性がある。</li> <li>• 特定のサンプル時間がブロックパラメーター個別に設定されると、メンテナンス性が低下する。</li> </ul>	

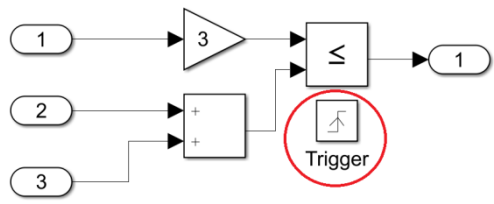
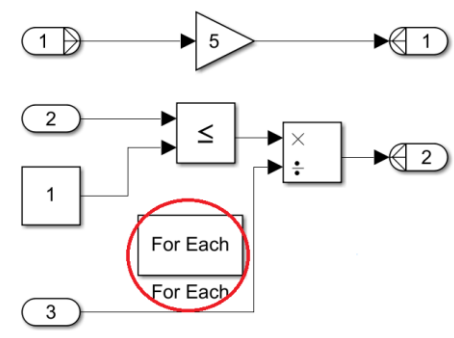
3.4.5. b\_0013: 固定小数点設定

規約 ID	b_0013	
規約名	固定小数点設定	
参考規約 ID	jc_0643	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>{データ型}が固定小数点で、{スケールリング}で"勾配とバイアス"を選択した場合、{バイアス}を"0"にする。</p> <p>【例】</p> 	—
根拠		
サブ ID	内容	
a	<p>バイアスがモデル内で統一されていないと下記の問題が発生する。</p> <ul style="list-style-type: none"> <li>・見た目で動作をイメージできず、意図しないオーバーフロー/アンダーフローが発生する。</li> <li>・無駄な演算が発生し、コード効率/演算負荷が悪化する。</li> </ul>	

3.5. 条件付きサブシステム関連

3.5.1. b\_0014: 条件付きサブシステム内のブロック配置

規約 ID	b_0014	
規約名	条件付きサブシステム内のブロック配置	
参考規約 ID	db_0146	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>下記ブロックはサブシステム内の最上位に配置します。</p> <ul style="list-style-type: none"> <li>・条件入力ブロック</li> <li>・ [ForEach]</li> </ul> <p><b>【例】</b></p> <ul style="list-style-type: none"> <li>・ 条件入力ブロック</li> </ul>  <ul style="list-style-type: none"> <li>・ [ForEach]</li> </ul> 	—

規約		
サブ ID	記述内容	カスタムパラメーター
a	<p><b>【誤例】</b></p> <ul style="list-style-type: none"> <li>条件入力ブロック</li> </ul>  <ul style="list-style-type: none"> <li>[ForEach]</li> </ul> 	
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>条件付きサブシステム直下の階層なのかが判断しやすくなる。</li> <li>配置を同一にすることで可読性が向上する。</li> </ul>	

## 3.5.2. b\_0015: 条件付きサブシステムにおける Outport ブロックの初期値設定

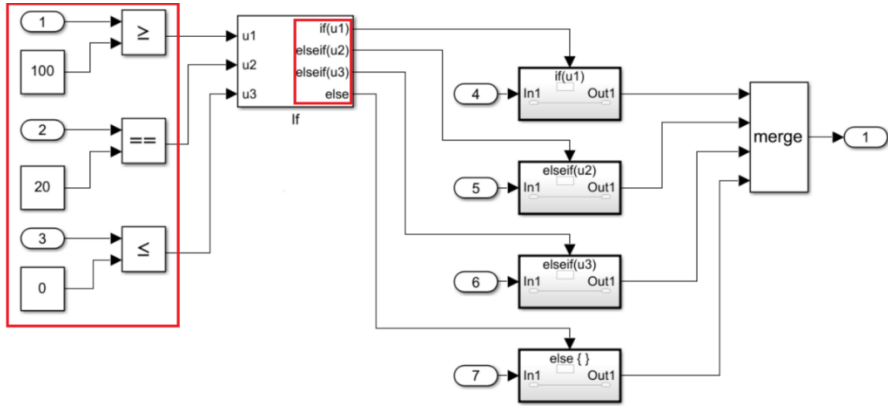
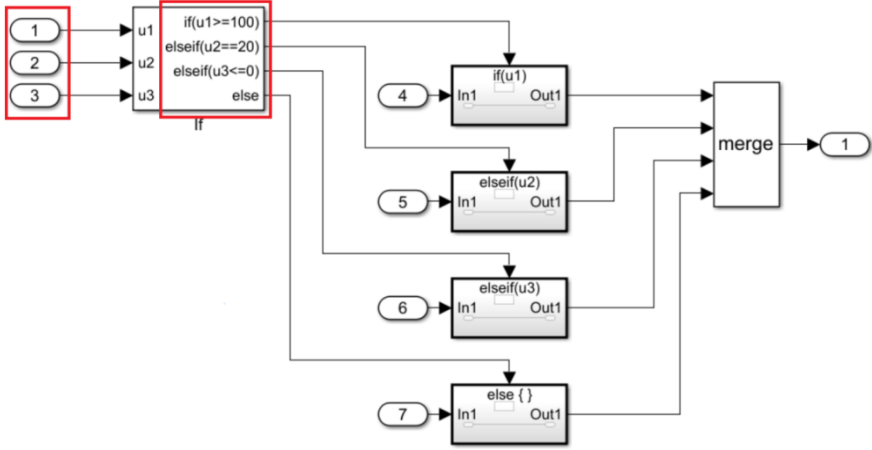
規約 ID	b_0015	
規約名	条件付きサブシステムにおける Outport ブロックの初期値設定	
参考規約 ID	jc_0640	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>モデルに含まれている条件付きサブシステムにおいて、初期条件を設定したブロック ([Constant]、[Delay]など)を [Outport]に接続している場合、その初期値を [Outport]にも明示的に設定する。</p> <p>ただし、条件付きサブシステムの出力信号が [Merge]に接続されている場合は、その初期値は [Merge]に明示的に設定する。</p>	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>初期値を明確にすることで可読性が向上する。</li> </ul>	



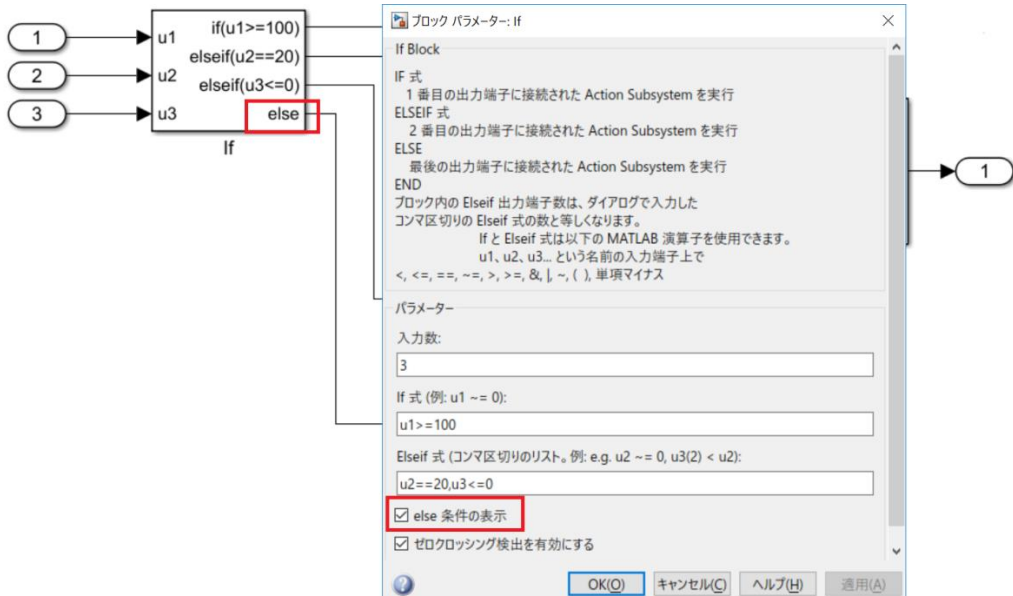

3.5.3. b\_0016: Merge ブロックへ入力する信号線の使用制限

規約 ID	b_0016	
規約名	Merge ブロックへ入力する信号線の使用制限	
参考規約 ID	jc_0659	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[Merge]には、条件付きサブシステムの出力信号のみを入力する。</p> <p>【例】</p> <p>【誤例】</p>	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>意図した通りにシミュレーションが動作しなくなることを防止する。</li> </ul>	

3.5.4. b\_0017:if ブロックの使用方法

規約 ID	b_0017	
規約名	if ブロックの使用方法	
参考規約 ID	na_0003	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[If]の{If 式}、{Elseif 式}には入力信号以外を定義しない。</p> <p>—</p> <p><b>【例】</b> if 式には入力変数のみ定義している。</p>  <p><b>【誤例】</b> if 式に比較演算子を使用して定義している。</p> 	
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・ [if]の外に条件をブロックで定義することで可読性が向上する。</li> <li>・ [if]の外に記述することで、論理演算式に焦点をあてた検証を可能にする。</li> </ul>	

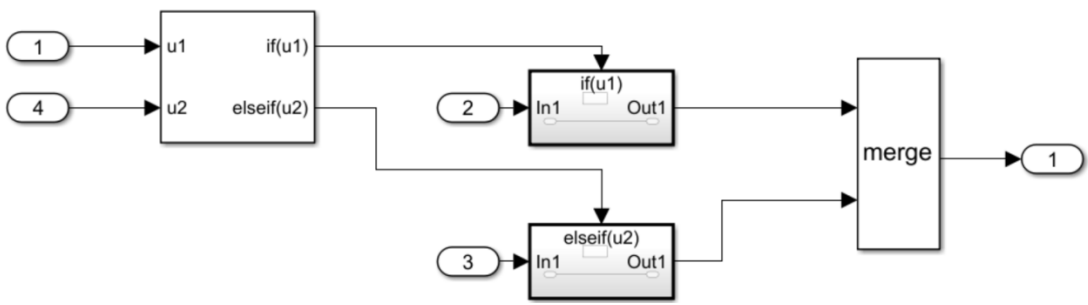
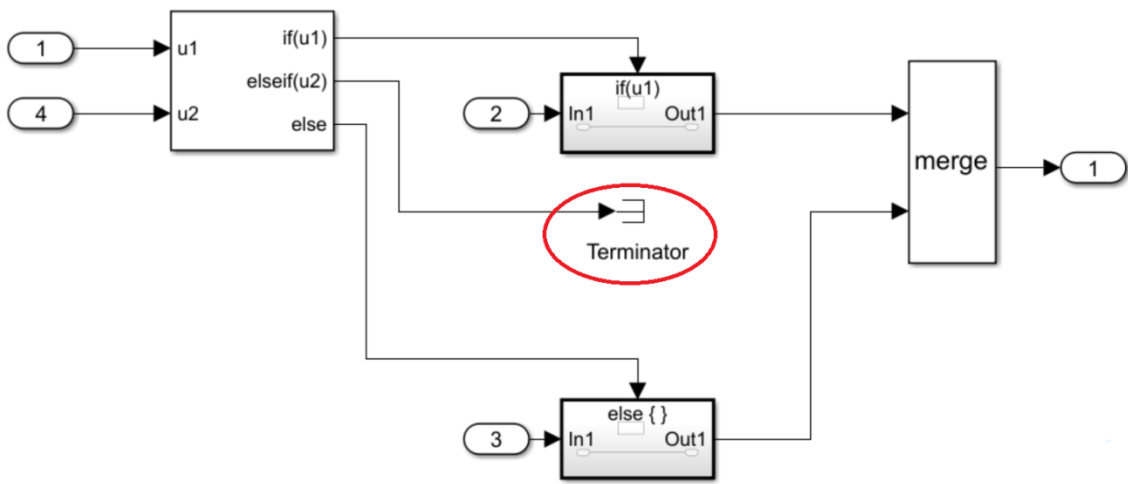
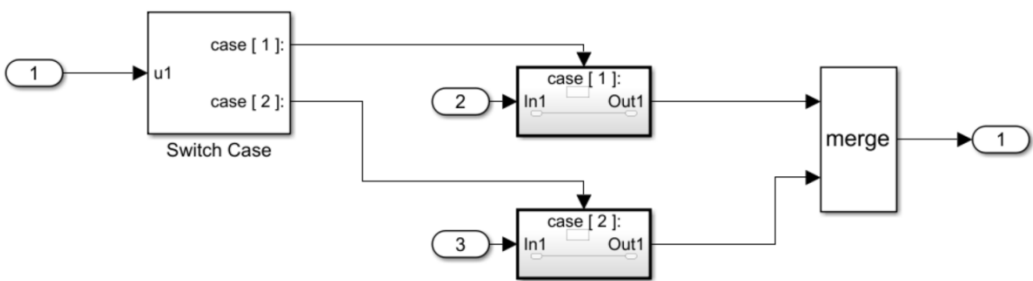
3.5.5. b\_0018: 条件付き制御フローブロックの使用方法

規約 ID	b_0018	
規約名	条件付き制御フローブロックの使用方法	
参考規約 ID	jc_0656	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[If]、[Switch Case]は、下記の設定により、全ての条件における動作を明示的にする。</p> <ul style="list-style-type: none"> <li>・ [if]は、{else 条件の表示}にチェックを入れる。</li> <li>・ [Switch Case]は、{default ケースを表示}にチェックを入れる。</li> </ul>	—
	<p><b>【例】 if 式に else 条件あり</b></p>  <p><b>switch に default の設定あり</b></p> 	

規約		
サブ ID	記述内容	カスタムパラメーター
a	<p><b>【誤例】</b></p> <p>if 式に else 条件なし</p> <p>switch に default の設定なし</p>	
<p><b>根拠</b></p>		
サブ ID	内容	
a	<p>・モデル上で例外処理 (else、default) を明示的に設定することで、何も行わない処理なのか、設計漏れ (記述忘れ) なのか判断し易くなる。</p>	

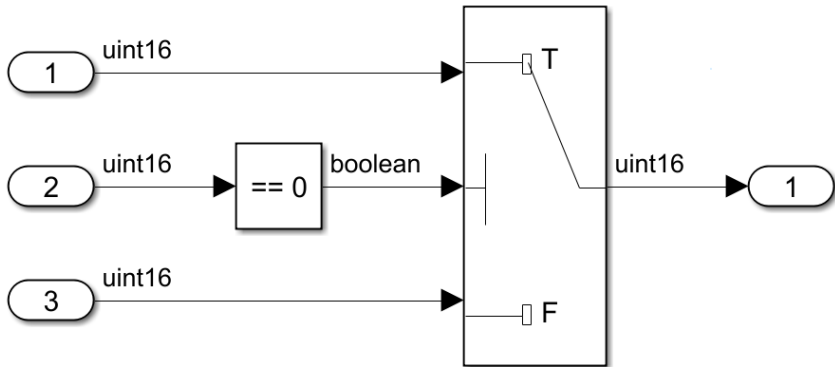
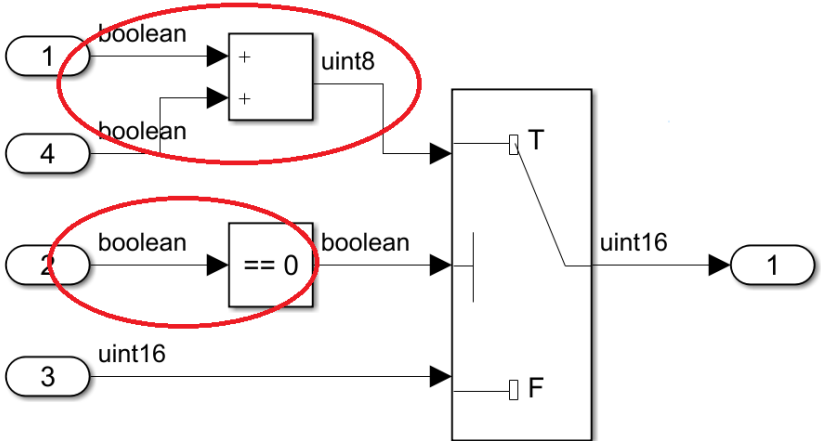
3.5.6. b\_0019: 条件付き制御フローブロックと Merge ブロックによる出力値保持

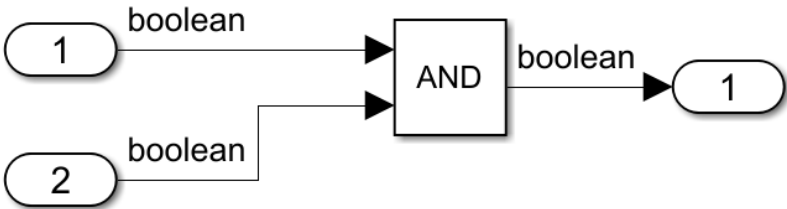
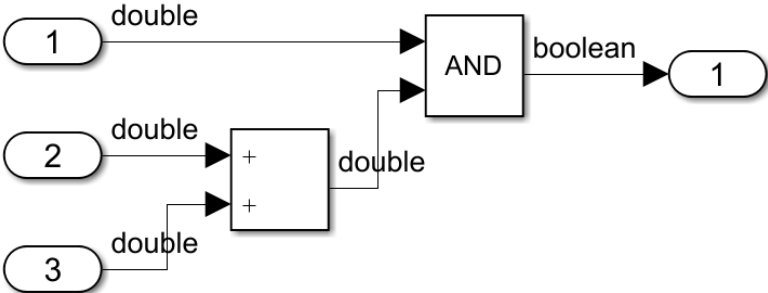
規約 ID	b_0019	
規約名	条件付き制御フローブロックと Merge ブロックによる出力値保持	
参考規約 ID	jc_0657	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[If]、[Switch Case]と[Merge]で機能を切り替える際に、過去の値を保持したい場合は、遅延ブロックを使用してフィードバックループにする。</p> <p><b>【例】</b> if の例</p> <p>switch の例</p>	—

規約		
サブ ID	記述内容	カスタムパラメーター
a	<p><b>【誤例】</b> if の例</p>  <p>[Terminator]を使用しても前回値保持可能であるが、使用禁止とする。</p>  <p>switch の例</p> 	
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>過去の値を保持することが明示的になる。</li> </ul>	

3.6. 演算系ブロック

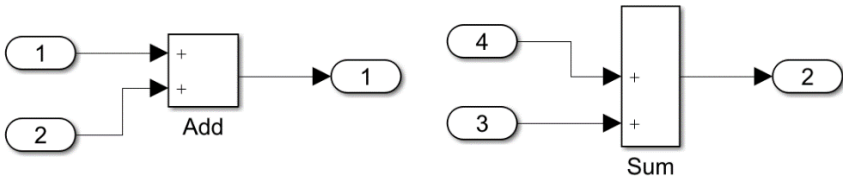
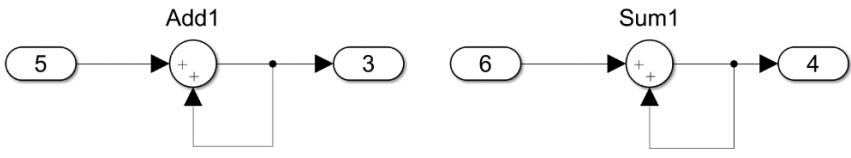
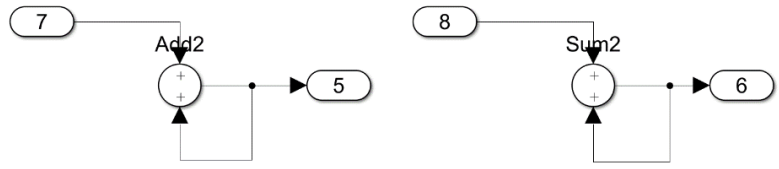
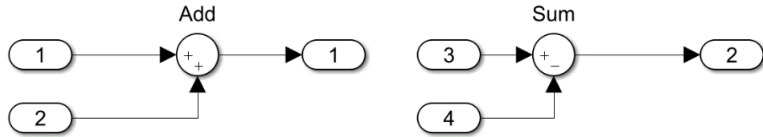
3.6.1. b\_0020: 論理演算と数値演算の適切な実装

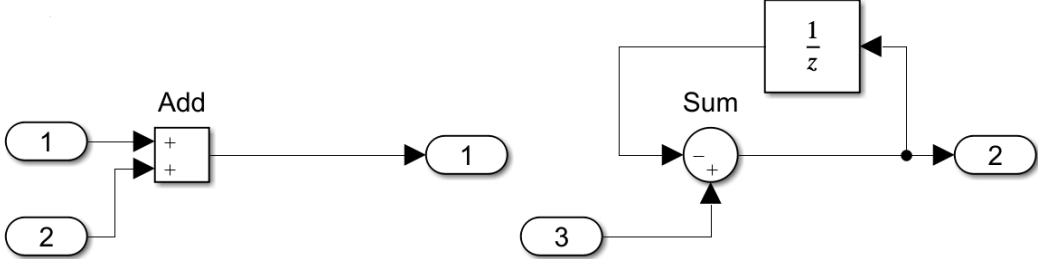
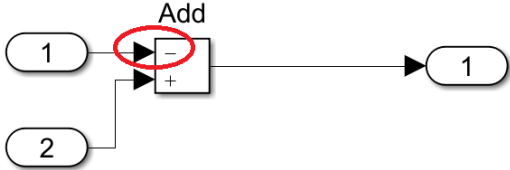
規約 ID	b_0020	
規約名	論理演算と数値演算の適切な実装	
参考規約 ID	na_0002	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>数値型信号の入力を期待するブロックには、論理型信号を入力しない。なお、数値型信号を入力するブロックに関してはプロジェクトで定めた設定とする。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> <p>論理型を、数値入力进行处理するブロックに接続している。 論理型と数値型を比較している。</p> 	数値型信号の入力を期待するブロック

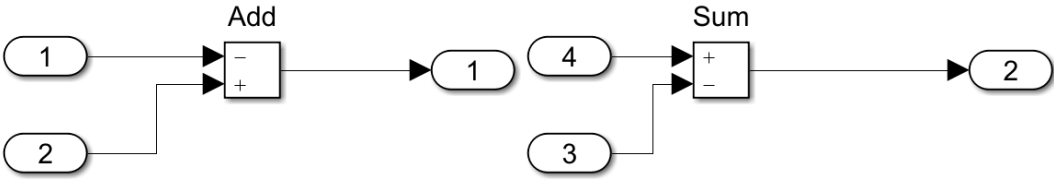
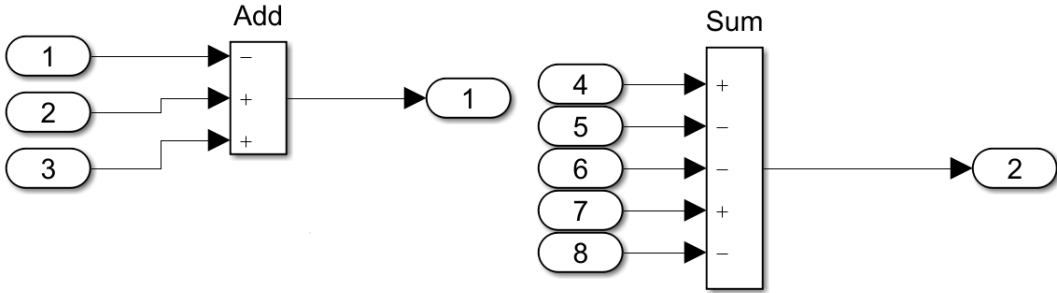
規約		
サブ ID	記述内容	カスタムパラメーター
b	<p>論理型信号の入力を期待するブロックには、数値型信号を入力しない。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> <p>数値出力を論理入力として処理するブロックの入力に接続している。</p> 	数値型信号の入力を期待するブロック
根拠		
サブ ID	内容	
ab	<p>・論理型と数値型を同じ扱いにして計算すると、設計意図が不明になり、改修時に意図を間違え不具合の要因になる。</p>	



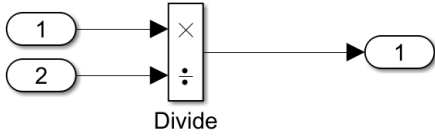
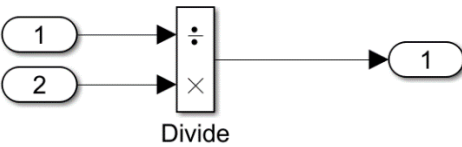
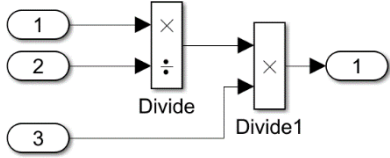
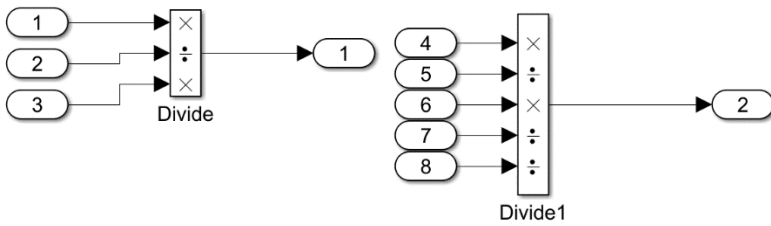
3.6.2. b\_0021: 加減算ブロックの使用法

規約 ID	b_0021	
規約名	加減算ブロックの使用法	
参考規約 ID	jc_0121	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>加減算を処理する場合は、{アイコン形状}を「四角形」に設定する。</p> <p><b>【例外】</b> フィードバックループの場合は、{アイコン形状}を「円形」に設定しても良い。</p> <p><b>【例】</b> 「Sum」、「Add」に関係なく「四角形」に設定する。</p>  <p><b>【例外】</b> フィードバックループの例</p>  <p>なお、「円形」に設定した場合は、符号リストに「 ++」など先頭に「 」を追加しないと下図のような形状になるため注意が必要である。</p>  <p><b>【誤例】</b> 「円形」に設定している。</p> 	—

規約		
サブ ID	記述内容	カスタムパラメーター
b	<p>加減算処理の第一入力の符号は「+」に設定する。 ただし、フィードバックループの場合は、第一入力の符号を「-」に設定することもできる。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> <p>第一入力の符号を「-」に設定している。</p> 	—

規約		
サブ ID	記述内容	カスタムパラメーター
c	<p>加減算処理の入力数は 2 つ以下に設定します。</p> <p>【例】</p>  <p>【誤例】</p> <p>入力数が 3 つ以上に設定されている。</p> 	—
根拠		
サブ ID	内容	
a	・使用方法をそろえることで可読性が向上する。	
b	・第一入力の符号が統一することで可読性が向上する。	
c	<ul style="list-style-type: none"> <li>・入力数を規定することで、可読性が向上する。</li> <li>・演算順序を明確に規定できる。</li> </ul>	

3.6.3. b\_0022: 乗除算ブロックの演算子順序

規約 ID	b_0022	
規約名	乗除算ブロックの演算子順序	
参考規約 ID	jc_0610	
規約		
サブ ID	記述内容	カスタムパラメーター
a	乗除算ブロックの第一入力の符号は”*”に設定する。  【例】   【誤例】 	—
b	乗除算ブロックの入力数は2つ以下に設定する。  【例】   【誤例】 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>数式とブロックの入力順が逆になり、可読性が悪くなる。</li> <li>浮動小数点の場合、ブロック通りの演算順序((1÷第一入力) × 第二入力)のコードが生成されるが、除算を後にすれば演算回数を削減できる。</li> </ul>	
b	演算順序が明確になる。	

## 3.6.4. b\_0023: 乗除算ブロックの入力符号

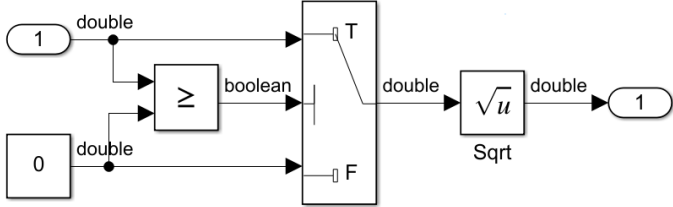
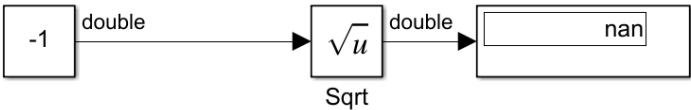
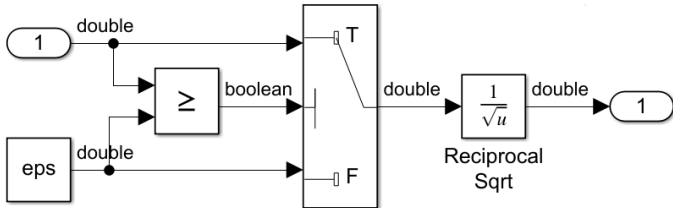
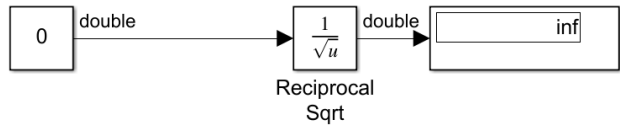
規約 ID	b_0023	
規約名	乗除算ブロックの入力符号	
参考規約 ID	jc_0611	
規約		
サブ ID	記述内容	カスタムパラメーター
a	乗除算ブロックの入力に固定小数点型を利用する場合は、入力信号のデータ型の符号の有無を同一にする。	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>固定小数点コードの生成時に LSB 毎のユーティリティ関数が作成され、使用する型を統一する事でユーティリティ関数の数を抑制できる。</li> </ul>	

## 3.6.5. b\_0024: Simulink における除算

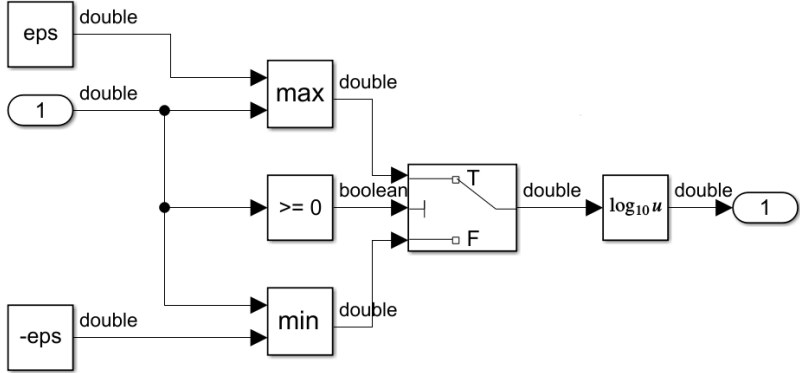
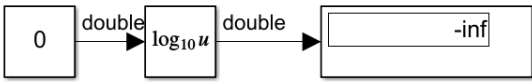
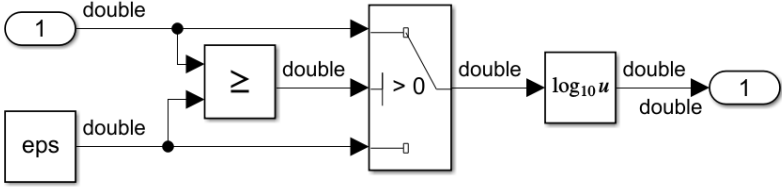
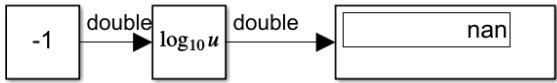
規約 ID	b_0024	
規約名	Simulink における除算	
参考規約 ID	jc_0794	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>除算を行う場合、ゼロ割回避の処理を入れる。</p> <p><b>【例外】</b> 分母に定数等を設定し、ゼロ割が発生しない場合は処理を追加しなくても良い。</p>	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・意図した動作、コード生成結果にならない可能性がある。</li> </ul>	

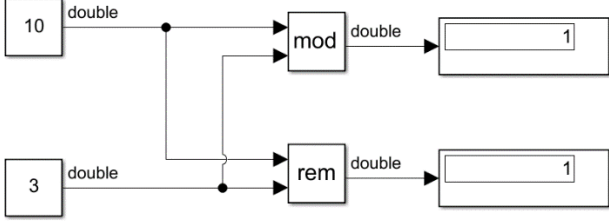
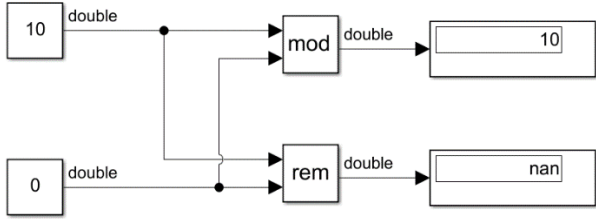
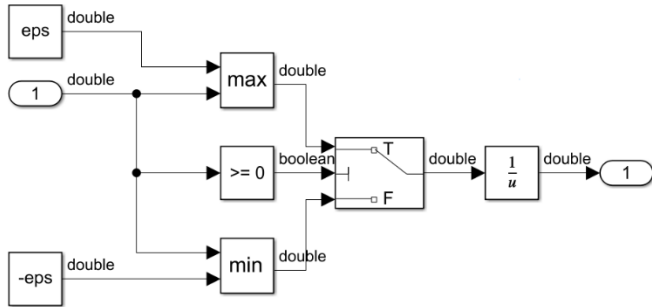
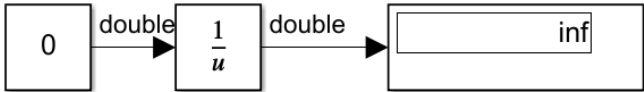
3.6.6. b\_0025: 数値演算ブロックの入力

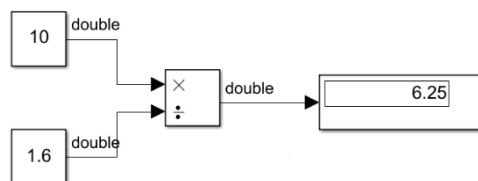
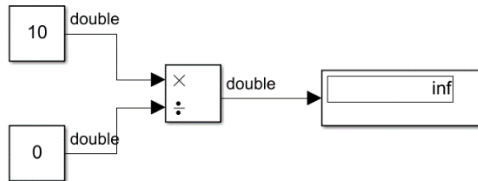
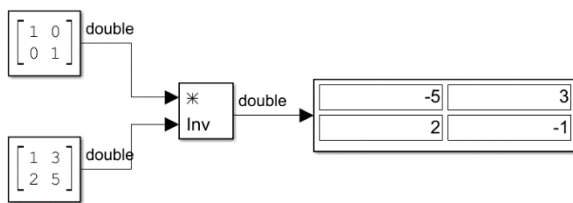
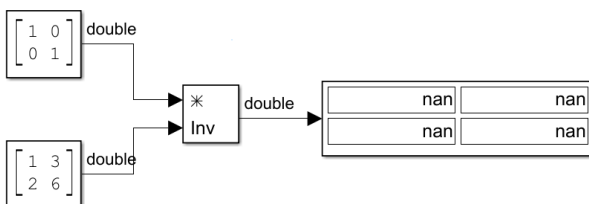
規約 ID	b_0025	
規約名	数値演算ブロックの入力	
参考規約 ID	jc_0805	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[abs]を使用する場合は、符号付き整数型の最小値を入力しない。</p> <p>または、[abs]を使用する場合は、{整数オーバーフローで飽和}にチェックを入れる。</p>	—
	<p><b>【例】</b></p> <p><b>【誤例】</b></p>	
b	<p>[abs]を使用する場合は、符号なしの整数型や固定小数点型を入力しない。</p>	—
	<p><b>【例】</b></p> <p><b>【誤例】</b></p>	

規約		
サブ ID	記述内容	カスタムパラメーター
c	<p>[Sqrt]を使用する場合は、負の数を入力しない。</p> <p><b>【例】</b> 0未満の入力をしないように制限している。</p>  <p><b>【誤例】</b></p> 	—
d	<p>[Reciprocal Sqrt]を使用する場合は、ゼロ以下の数を入力しない。</p> <p><b>【例】</b> 0以下の入力をしないように制限している。</p>  <p><b>【誤例】</b></p> 	—



規約		
サブ ID	記述内容	カスタムパラメーター
e	<p>[Math Function]の{関数}を「log」または「log10」に設定して使用する場合、ゼロを入力しない。</p> <p>【例】</p>  <p>【誤例】</p> 	—
f	<p>[Math Function]の{関数}を「log」または「log10」に設定して使用する場合、負の数を入力しない。</p> <p>【例】</p>  <p>【誤例】</p> 	—

規約		
サブ ID	記述内容	カスタムパラメーター
g	<p>[Math Function]の{関数}を「mod」または「rem」に設定して使用する場合、第二引数にゼロを入力しない。</p> <p>【例】</p>  <p>【誤例】</p> 	—
h	<p>[Math Function]の{関数}を「reciprocal」に設定して使用する場合、ゼロを入力しない。</p> <p>【例】</p>  <p>【誤例】</p> 	—

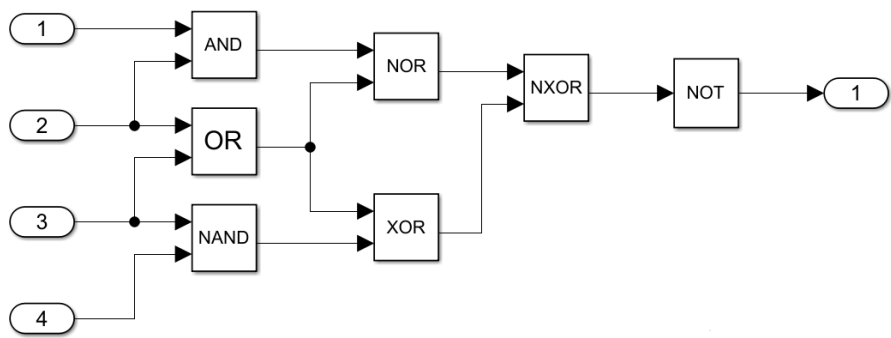
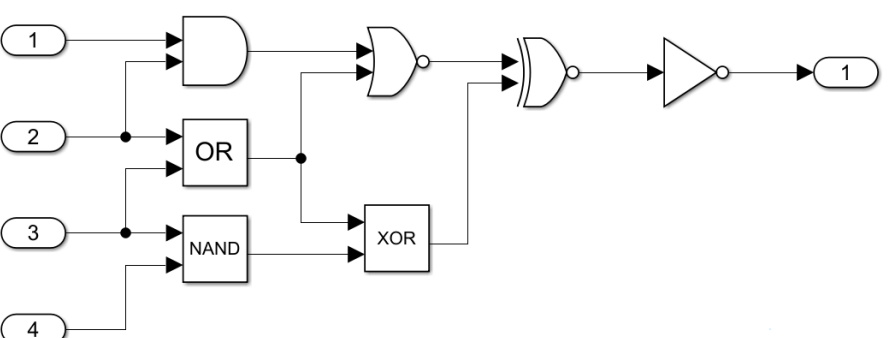
規約		
サブ ID	記述内容	カスタムパラメーター
i	<p>[Product]の{乗算}を”要素単位 (.*)” に設定して使用する 場合、約数入力({入力数}で「/」を設定した入力ポート)に ゼロを入力しない。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—
j	<p>[Product]の{乗算}を”行列(*)” に設定して使用する 場合、約数入力({入力数}で「/」を設定した入力ポート)に 特異値行列を入力しない。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—

根拠	
サブ ID	内容
a1	<ul style="list-style-type: none"> <li>不正な値を入力すると結果が処理系依存になるため、意図しない動作となる可能性がある。</li> <li>明示的にモデルで不正な値が入力されないように処理することで、可読性が向上し設計意図も明確になる。</li> </ul>
a2	<ul style="list-style-type: none"> <li>設定により、不正な値が入力された時の意図しない動作を防止する。ただし、不正な値が入力された時の演算結果が意図した結果であるかは別途検証が必要である。</li> </ul>
b	<ul style="list-style-type: none"> <li>コード生成時に最適化され、コードまでトレースできないブロックとなる可能性がある。</li> </ul>
c、d、w、f、g、 h、i、j	<ul style="list-style-type: none"> <li>不正な値を入力すると結果が処理系依存になるため、意図しない動作となる可能性がある。</li> </ul>

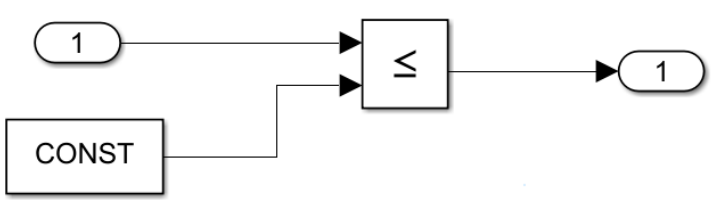
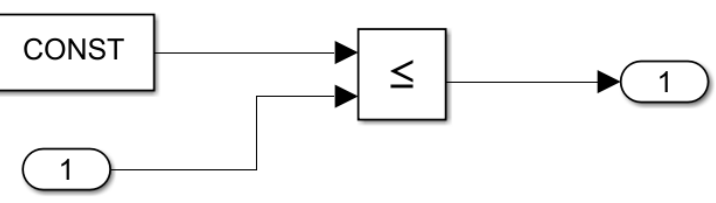
## 3.6.7. b\_0026: Fcn ブロックの使用方法

規約 ID	b_0026	
規約名	Fcn ブロックの使用方法	
参考規約 ID	jc_0622	
規約		
サブ ID	記述内容	カスタムパラメーター
a	[Fcn]の演算にて優先順位が異なる演算子が混在する場合は、括弧を付け優先順位を明示します。	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>優先順位が異なる演算子を使用する場合、括弧を用いて計算順序を明確化しないと可読性が落ちる、もしくは間違っって解釈して演算を間違える可能性がある。</li> </ul>	

3.6.8. b\_0027: 論理演算ブロックのアイコン形状

規約 ID	b_0027	
規約名	論理演算ブロックのアイコン形状	
参考規約 ID	jc_0621	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[Logical Operator]の{アイコン形状}は、” 四角形” にする。</p> <p>【例】</p>  <p>【誤例】</p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・ 同一の機能を複数の表現で記述できる場合には、表現を統一することで可読性が向上する。</li> <li>・ {アイコン形状}が「特徴」の場合、それぞれの形状が似ているため、見間違いリスクが「四角形」に比べて増加する。</li> </ul>	

3.6.9. b\_0028: Relational Operator の使用方法

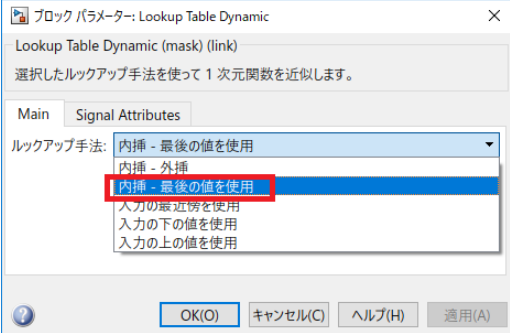
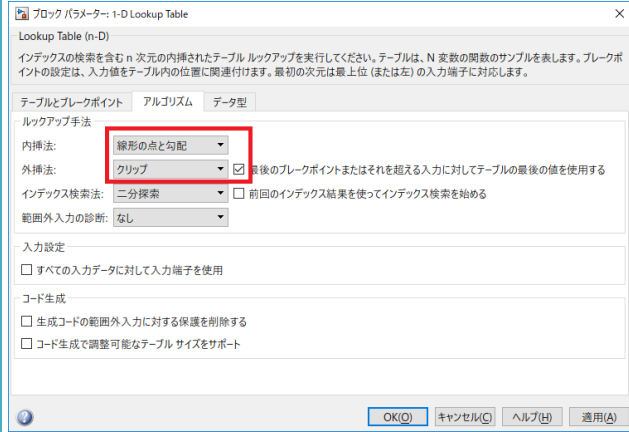
規約 ID	b_0028	
規約名	Relational Operator の使用方法	
参考規約 ID	jc_0131	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[Relational Operator]を信号と定数値の比較に使用する場合は、定数値を第二入力とする。</p> <p>【例】</p>  <p>【誤例】</p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>定数値との比較方法を統一することで、可読性が向上する。</li> </ul>	

3.6.10. b\_0029: Simulink における浮動小数点型の比較

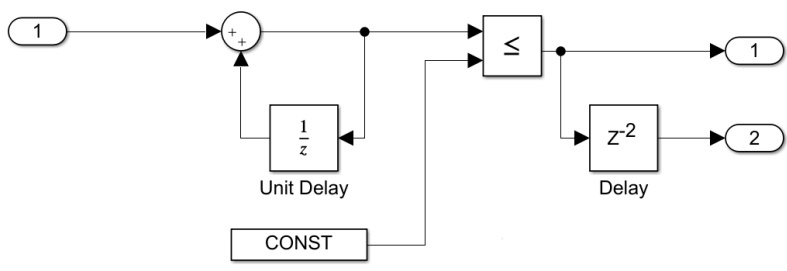
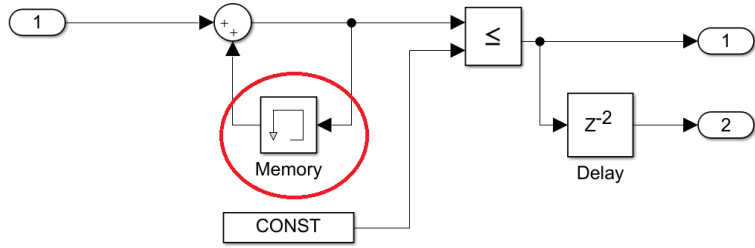
規約 ID	b_0029	
規約名	Simulink における浮動小数点型の比較	
参考規約 ID	jc_0800	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>浮動小数点型の等価比較演算==、~=は使用しない。</p> <p>【例】</p> <p>【誤例】</p>	—
根拠		
サブ ID	内容	
a	<p>・浮動小数点の性質上、値に誤差を含むため、意図した等価比較結果にならない可能性がある。</p>	



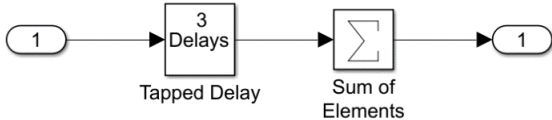
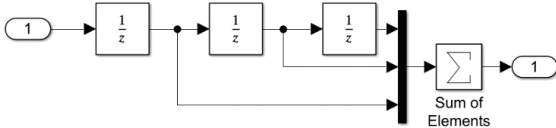
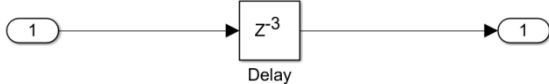
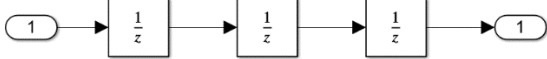
3.6.11. b\_0030: Lookup Table 系ブロックの使用法

規約 ID	b_0030	
規約名	Lookup Table 系ブロックの使用法	
参考規約 ID	jc_0626	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>Lookup Table Dynamic ブロックの{ルックアップ手法}は「内挿 - 最後の値を使用」に設定する。</p> <p><b>【例】</b></p> 	—
b	<p>1-D Lookup Table、2-D Lookup Table および n-D Lookup Table {内挿法}は「線形の点と勾配」、{外挿法}は「クリップ」に設定し、{最後のブレイクポイントまたはそれを超える入力に対してテーブルの最後の値を使用する}にチェックを入れる。</p> 	—
根拠		
サブ ID	内容	
ab	<p>・ Lookup Table 系ブロックに想定外の値が入力された場合、外挿法によって出力値を推測すると、実際には取り得ない値になったり、Lookup Table 系ブロックの出力がオーバーフローしたりする可能性がある。</p>	

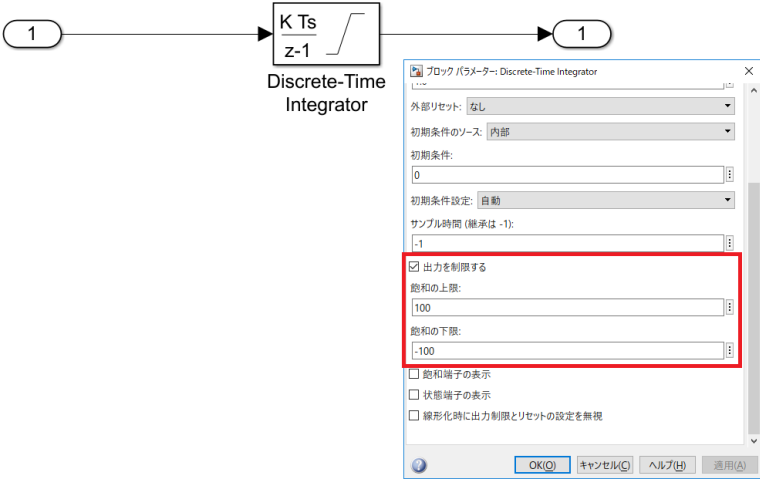
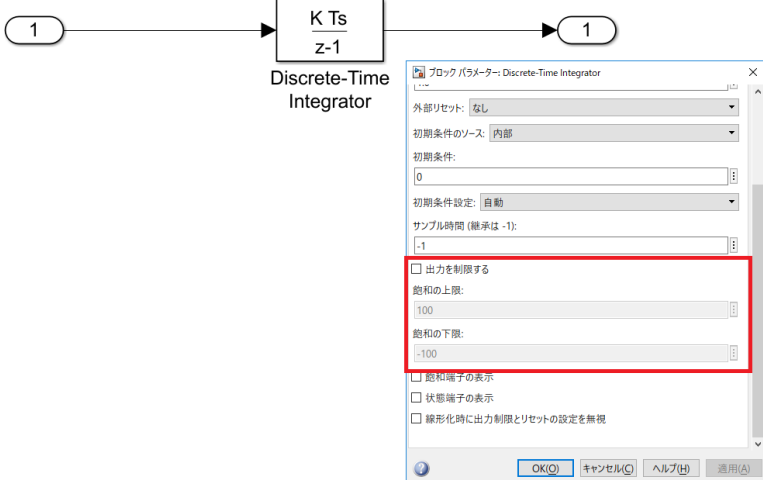
3.6.12. b\_0031: 離散系遅延ブロックの使い分け

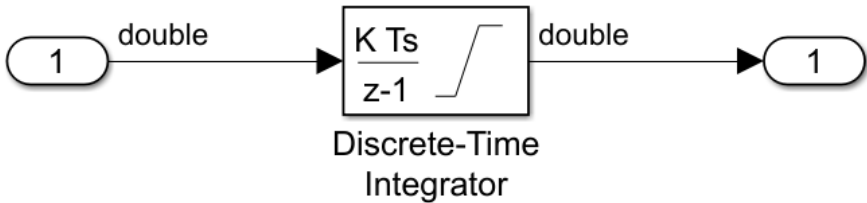
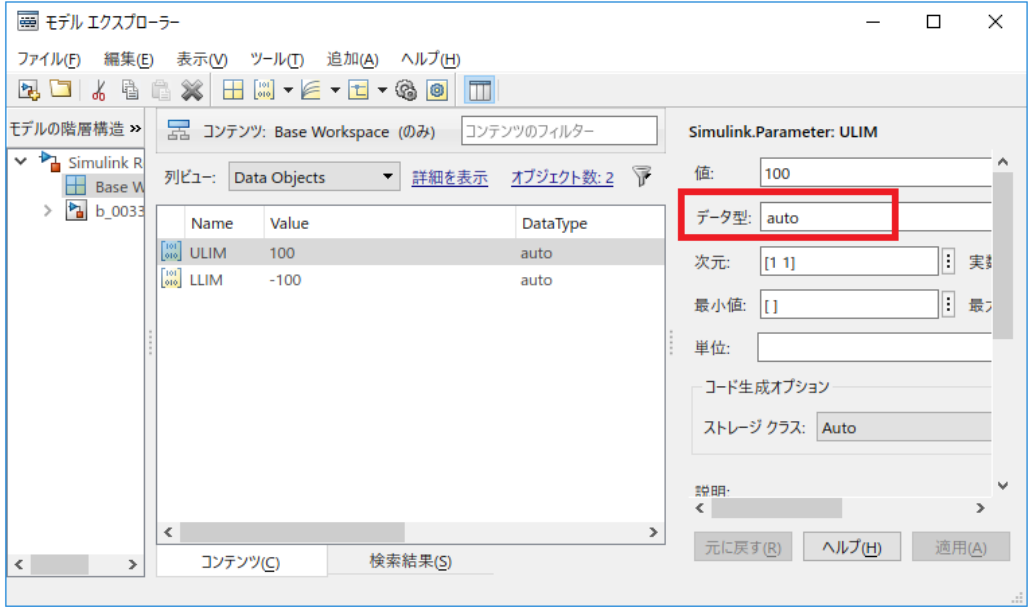
規約 ID	b_0031	
規約名	離散系遅延ブロックの使い分け	
参考規約 ID	jc_0623	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>離散系のモデル又はサブシステム内では、[Unit Delay] もしくは [Delay] を使用する。</p> <p>【例】</p>  <p>【誤例】</p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・使用方法を揃えることで可読性が向上する。</li> </ul>	

3.6.13. b\_0032: Tapped Delay ブロック/Delay ブロックの使用方法

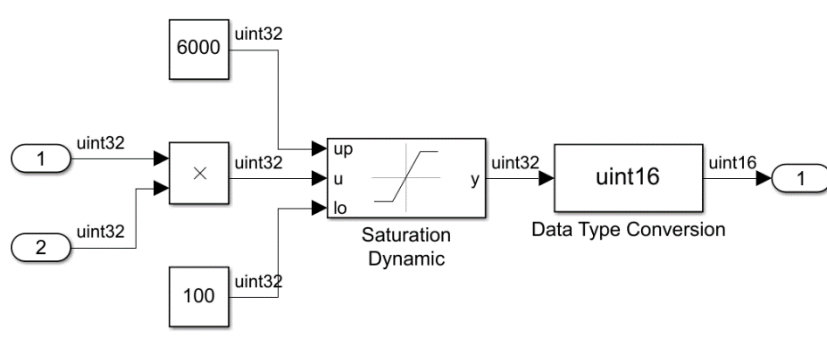
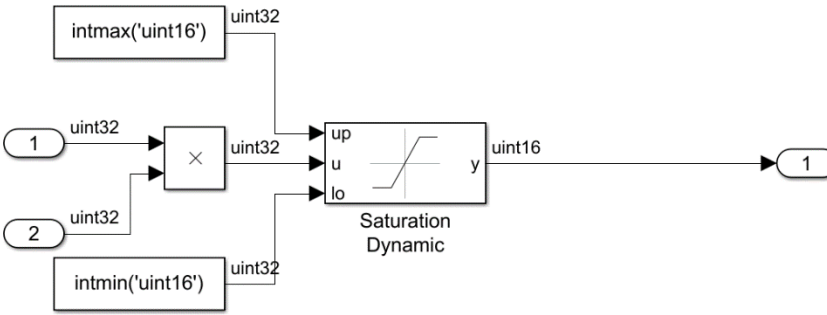
規約 ID	b_0032	
規約名	Tapped Delay ブロック/Delay ブロックの使用方法	
参考規約 ID	jc_0624	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>複数回前の過去値を保持する場合、保持している全ての値をベクトル信号として生成するには[Tapped Delay]を使用する。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—
b	<p>複数回前の過去値を保持する場合、保持している最も古い信号の値のみを取得するには[Delay]を使用します。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>• Tapped Delay ブロックは、過去の値を持つ配列が設定され、コードの可読性が向上し、コード効率面でも有利になる。</li> </ul>	
b	<ul style="list-style-type: none"> <li>• モデルの可読性が向上し、コード効率面でも有利になる。</li> </ul>	

3.6.14. b\_0033: Discrete-Time Integrator ブロックの使用方法

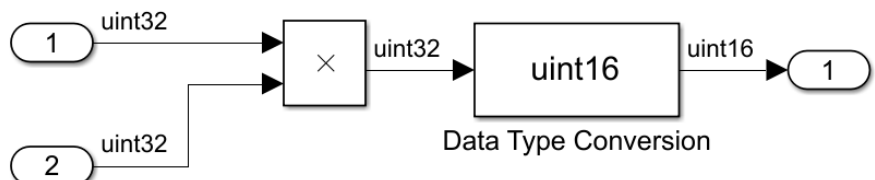
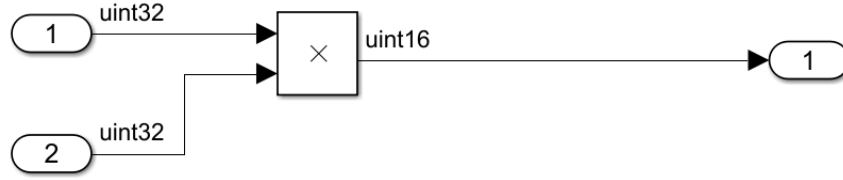
規約 ID	b_0033	
規約名	Discrete-Time Integrator ブロックの使用方法	
参考規約 ID	jc_0627	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[Discrete-Time Integrator] の {飽和の上限} と {飽和の下限} を設定する。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—

規約		
サブ ID	記述内容	カスタムパラメーター
b	<p>[Discrete-Time Integrator]の{飽和の上限}と{飽和の下限}を <code>mpt.Parameter</code> 等で設定する場合、そのパラメーターのコード生成用の設定は、{データ型}を「auto」にする。</p> <p>【例】</p>  	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>• ブロックの出力がオーバーフローするのを避け、このブロックの出力を使う他の演算ブロックが想定外の結果になることを防止する。</li> </ul>	
b	<ul style="list-style-type: none"> <li>• <code>auto</code>、<code>single</code>、<code>double</code> 以外のデータ型を設定するとモデルコンパイル(シミュレーション)時にエラーが発生するため、使用方法を揃える。</li> </ul>	

3.6.15. b\_0034: Saturation ブロックの使用方法

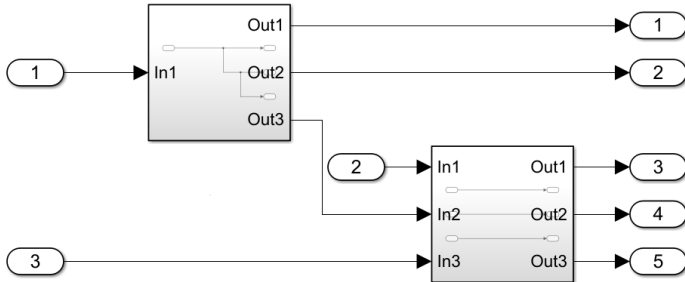
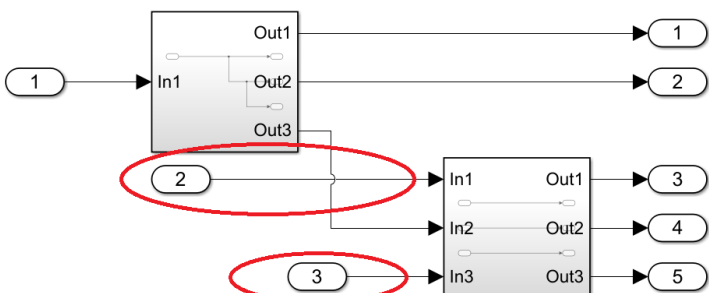
規約 ID	b_0034	
規約名	Saturation ブロックの使用方法	
参考規約 ID	jc_0628	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[Saturation]または[Saturation Dynamic]は、物理量を制限する目的で使用する。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> <p>上限および下限にデータ型の最大値、最小値を設定して、型変換を目的として使用している。</p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・使用方法を揃えることで可読性および保守性が向上する。</li> </ul>	

3.6.16. b\_0035: 型変換を実施する場合の使用方法

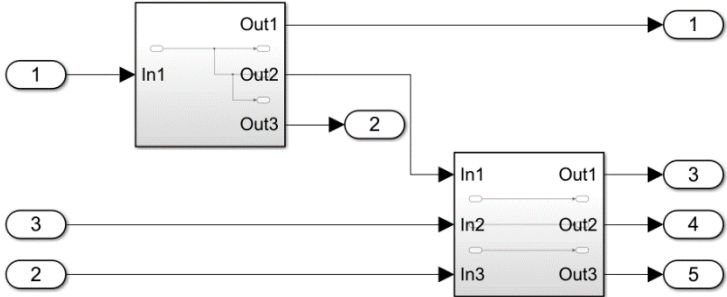
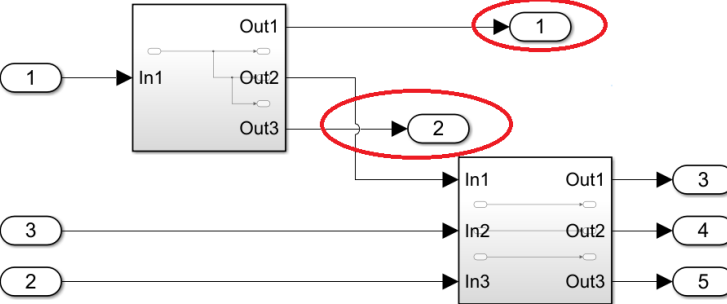
規約 ID	b_0035	
規約名	型変換を実施する場合の使用方法	
参考規約 ID	jc_0651	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>ブロックの出力信号のデータ型を、初期設定で決まる型以外に変える場合は[Data Type Conversion]を使用する。</p> <p>【例】</p>  <p>【誤例】</p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・演算と型変換を分ける事によって、実行順序とどの演算までがどのデータタイプを使用するかがブロックの構成上で明確化できる。</li> </ul>	

3.7. その他のブロック

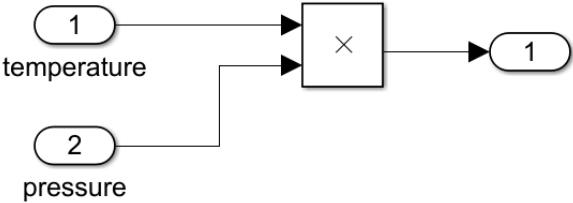
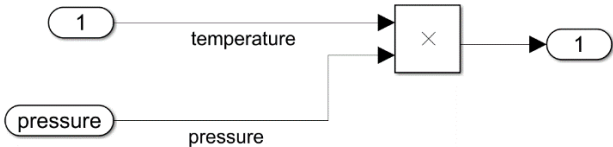
3.7.1. b\_0036: Inport ブロック / Outport ブロックの使用方法

規約 ID	b_0036	
規約名	Inport ブロック / Outport ブロックの使用方法	
参考規約 ID	db_0042	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[Inport]はダイアグラムの左側に配置する。</p> <p>【例外】 信号線が交差する場合は移動可能とする。</p> <p>【例】</p>  <p>【誤例】</p> 	—



規約		
サブ ID	記述内容	カスタムパラメーター
b	<p>[Output]はダイアグラムの右側に配置する。</p> <p>【例外】 信号線が交差する場合は移動可能とする。</p> <p>【例】</p>  <p>【誤例】</p> 	—
根拠		
サブ ID	内容	
a、b	・使用方法を揃えることで可読性が向上する。	

3.7.2. b\_0037: Inport ブロック / Outport ブロックのアイコン表示

規約 ID	b_0037	
規約名	Inport ブロック / Outport ブロックのアイコン表示	
参考規約 ID	jc_0081	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[Inport]、[Outport]の{アイコン表示}は「端子番号」にする。</p> <p>【例】</p>  <p>【誤例】</p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・サブシステム外部で[Inport]または[Outport]の順番を、端子番号で確認するため、また、[Inport]または[Outport]にブロック名を表示するために、[Inport]、[Outport]の{アイコン表示}は”端子番号”にする。</li> <li>・ブロック名を表示し、[Inport]または[Outport]に接続される信号線の名前と、ブロック名を同一にする事で、階層化されたサブシステムにおいて、接続ミスを防ぎます。</li> <li>・ブロック表示サイズが統一されるため、可読性が向上する。</li> </ul>	

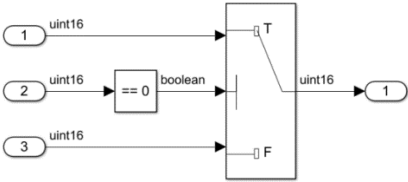
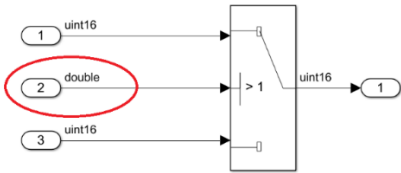
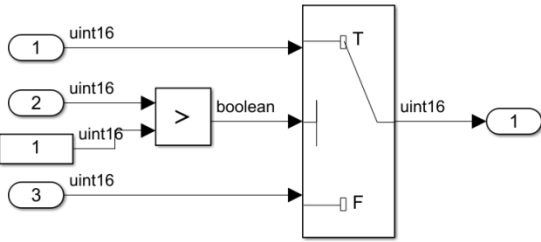
## 3.7.1. b\_0038: Goto/From の範囲

規約 ID	b_0038	
規約名	Goto/From の範囲	
参考規約 ID	na_0011	
規約		
サブ ID	記述内容	カスタムパラメーター
a	[Goto]の{タグの可視性}は「ローカル」にする。	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・ [Goto]と対応する[From]が異なる階層にあると、接続関係が解りにくくなる。</li> <li>・ [Goto]と対応する[From]が異なる階層にあると、バーチャルサブシステムを Atomic サブシステムに変更したときに、シミュレーションエラーが発生する可能性がある。</li> </ul>	

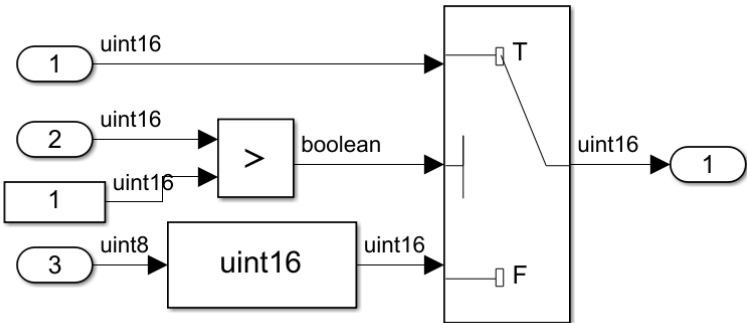
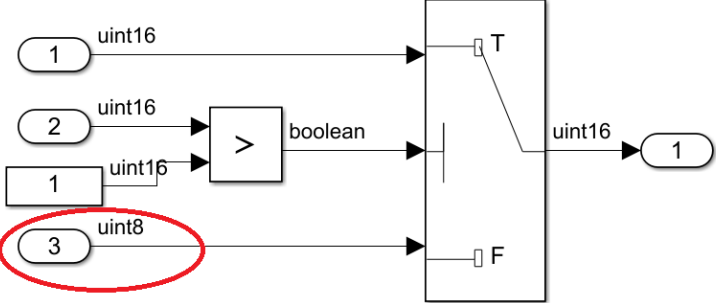
## 3.7.2. b\_0039: Data Store Memory ブロックの定義方法

規約 ID	b_0039	
規約名	Data Store Memory ブロックの定義方法	
参考規約 ID	jc_0161	
規約		
サブ ID	記述内容	カスタムパラメーター
a	[Data Store Memory]は、使用する最小スコープレベルで定義する。	—
b	[Data Store Memory]には、実行、コード生成に必要となるデータのみを定義する。	—
根拠		
サブ ID	内容	
a	・ 使用箇所を限定することで、信頼性が向上する。	
b	・ 未使用の[Data Store Memory](データ)が存在すると保守性、運用性に影響がある可能性がある。	

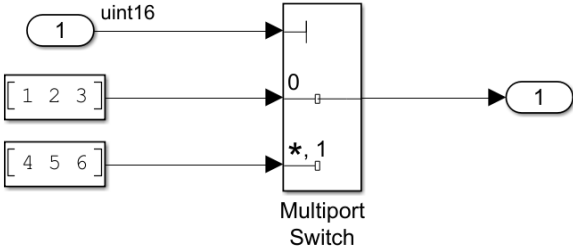
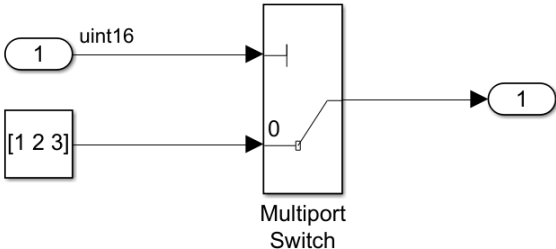
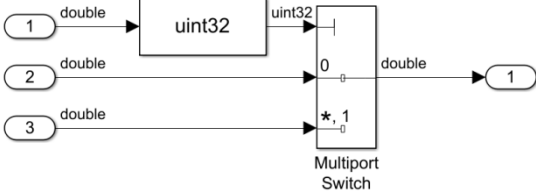
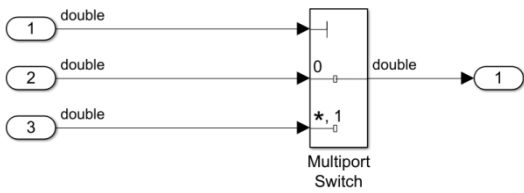
3.7.3. b\_0040: Switch ブロックの使用方法

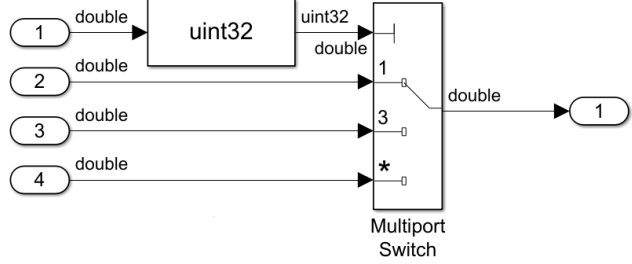
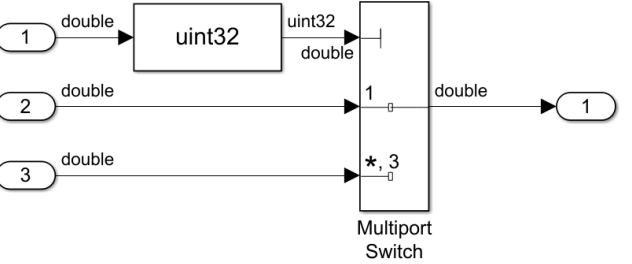
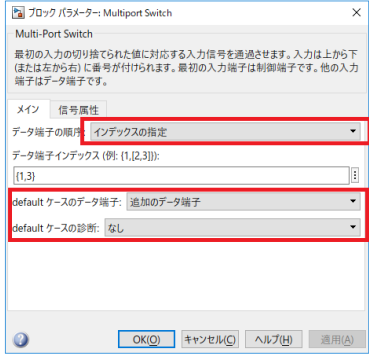
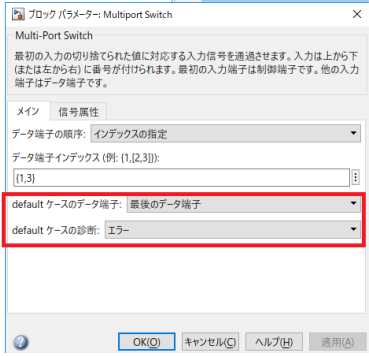
規約 ID	b_0040	
規約名	Switch ブロックの使用方法	
参考規約 ID	jc_0141	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[Switch]の2番目の入力(条件) は論理型を入力する。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—
b	<p>[Switch]の{1番目の入力}が通過する条件は「u2~=0」にする。</p> <p><b>【例】</b></p> <p>Switch ブロック外で演算式を示している。</p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・ 論理型と数値型を同じ扱いにして計算すると、設計意図が不明になり、改修時に意図を間違え不具合の要因になる。</li> </ul>	
b	<ul style="list-style-type: none"> <li>・ Switch ブロックの中に演算式を書くのではなく、Simulink ブロックで構成することで、可読性が向上する。</li> </ul>	

3.7.4. b\_0041: 切替機能を持つブロックの入出力データ型

規約 ID	b_0041	
規約名	切替機能を持つブロックの入出力データ型	
参考規約 ID	jc_0650	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>切替機能を持つブロック ([Switch]、[Multiport Switch]、[Index Vector])は、データ端子と出力ポートのデータ型を同じにする。</p> <p>【例】</p>  <p>【誤例】</p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・ 暗黙の型変換を防止する。</li> </ul>	

3.7.5. b\_0042: Multiport Switch ブロックの使用方法

規約 ID	b_0042	
規約名	Multiport Switch ブロックの使用方法	
参考規約 ID	jc_0630	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>[Multiport Switch]の{データ端子数}は” 2” 以上にする。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—
b	<p>[Multiport Switch]の制御端子への入力は、符号なし整数にする。</p> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	—

規約		
サブ ID	記述内容	カスタムパラメーター
c	<p>[Multiport Switch]の{データ端子の順序}を「インデックスの指定」とした場合は、下記の設定を行う。</p> <ul style="list-style-type: none"> <li>・{default ケースのデータ端子}を「追加のデータ端子」</li> <li>・{default ケースの診断}を「なし」</li> </ul> <p><b>【例】</b></p>  <p><b>【誤例】</b></p> 	<p>—</p>  
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>・データ端子数が1つの場合は、ベクトルの中からスカラを取り出すブロックに変わるため、意図しない出力になる可能性がある。</li> </ul>	
b	<ul style="list-style-type: none"> <li>・制御端子は0以上の整数値が期待する入力範囲のため、符号付き、もしくは、非整数の信号が制御端子に接続されると、接続ミスのように見える。</li> <li>・実際に負の値、非整数値が入力されたときに、意図しないデータ端子が選択される可能性がある。</li> </ul>	
c	<ul style="list-style-type: none"> <li>・“インデックスの指定”の場合、制御端子に指定したインデックス以外の値が入力されると、指定したインデックスの最後の値と同じ扱いとなります。そのため、意図しないデータ端子が選択される可能性がある。</li> </ul>	



#### 4. Stateflow

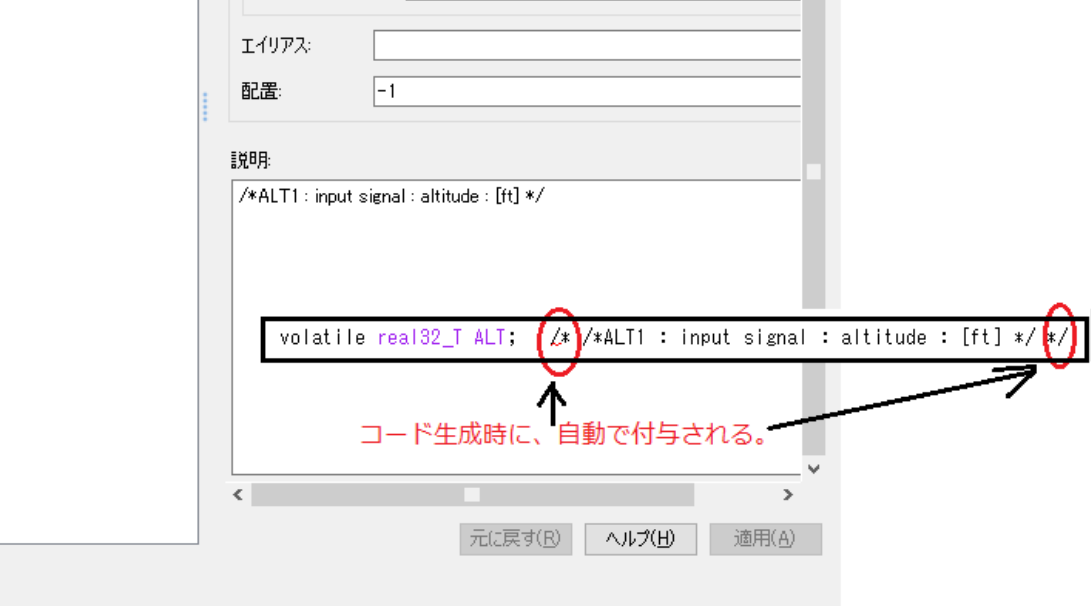
本ガイドライン作成時に「Stateflow」の使用経験が少なかったため、規定作成は見送ることとした。

5. その他

5.1.1. d\_0001: 列挙型の規定値の定義

規約 ID	d_0001	
規約名	列挙型の規定値の定義	
参考規約 ID	na_0031	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>列挙型の既定値は、「getDefaultValue」を用いて明示的に定義する。</p> <p><b>【例】</b></p> <pre> 2 3 classdef (Enumeration) d_0001 &lt; Simulink.IntEnumType 4     enumeration 5         Red(0) 6         Yellow(1) 7         Blue(2) 8     end 9     methods (Static = true) 10        function retVal = get DefaultValue() 11            retVal = BasicColors.red; 12        end 13    end 14 end 15 </pre> <p><b>【誤例】</b></p> <pre> 2 3 classdef (Enumeration) d_0001_2 &lt; Simulink.IntEnumType 4     enumeration 5         Red(0) 6         Yellow(1) 7         Blue(2) 8     end 9 end 10 </pre>	—
根拠		
サブ ID	内容	
a	<p>・列挙型に明示的な既定値を設定しない場合、enumeration の最初に記載した文字列が既定値として定義されるため、意図した値にならないことがある。</p>	

## 5.1.2. d\_0002: コメント記号/\*、\*/の使用禁止

規約 ID	d_0002	
規約名	コメント記号/*、*/の使用禁止	
参考規約 ID	jc_0081	
規約		
サブ ID	記述内容	カスタムパラメーター
a	<p>コード生成時にコメント記号が付与される以下の箇所には、コメント記号/*、*/を使用しない。</p> <ul style="list-style-type: none"> <li>• cgt ファイル内</li> <li>• mpt Signal の[説明]</li> <li>• mpt Parameter の[説明]</li> </ul> <p><b>【誤例】</b></p> 	—
根拠		
サブ ID	内容	
a	<ul style="list-style-type: none"> <li>• コード生成時に自動的にコメント記号/*、*/が付与されるため、意図せずコメントが入れ子になり、意図と異なる動作になる可能性がある。</li> </ul>	