

## 付録

## B

## Simulink の基本的な操作

Simulink とは MATLAB に含まれるシミュレータであり、ブロックを結線するだけで簡単に数値シミュレーションを行うことができる。また、本書では説明を省略するが、Simulink モデルを C コードに自動変換することで、Arduinoなどを介して実験装置を動かすこともできる。ここでは、シミュレータとしての Windows 版 Simulink (R2022a) の基本的な操作について説明する。

## B.1 Simulink の起動とライブラリ

### ■ Simulink の起動

MATLAB ウィンドウで「ホーム/Simulink」を選択するか、コマンドウィンドウで

```
>> simulink ↵
```

と入力すると、図 B.1 に示すように、Simulink スタートページが別ウィンドウで開く。つぎに、「空のモデル」を選択すると、Simulink モデルのウィンドウが開き、「Simulink スタートページ」が自動的に閉じられる。また、Simulink モデルのウイン



図 B.1 Simulink の起動

ドウで「ライブラリブラウザ」のボタンをクリックすると、様々な Simulink ブロックが含まれる Simulink ライブラリブラウザのウィンドウが開く。

## ■ Simulink ライブラリ

図 B.1 に示した Simulink ライブラリブラウザの左側には、利用可能なライブラリ (ファイル群) が表示される。利用したいライブラリにマウスカーソルを合わせ、左ボタンをクリックすると、ライブラリに含まれる Simulink ブロックの一覧が Simulink ライブラリブラウザの右側に表示される。一方、利用したいライブラリにマウスカーソルを合わせ、右ボタンで「\*\*\* ライブラリを開く」を選択すると、別ウィンドウに選択したライブラリが表示される。たとえば、Continuous に含まれる Simulink ブロックを表示させると、図 B.2 のようになる。

Simulink ライブラリに含まれるブロック群を表 B.1 にまとめる。また、本書で利用する Simulink ライブラリに含まれている Simulink ブロック (抜粋) を図 B.3 ~ B.9 に示す。

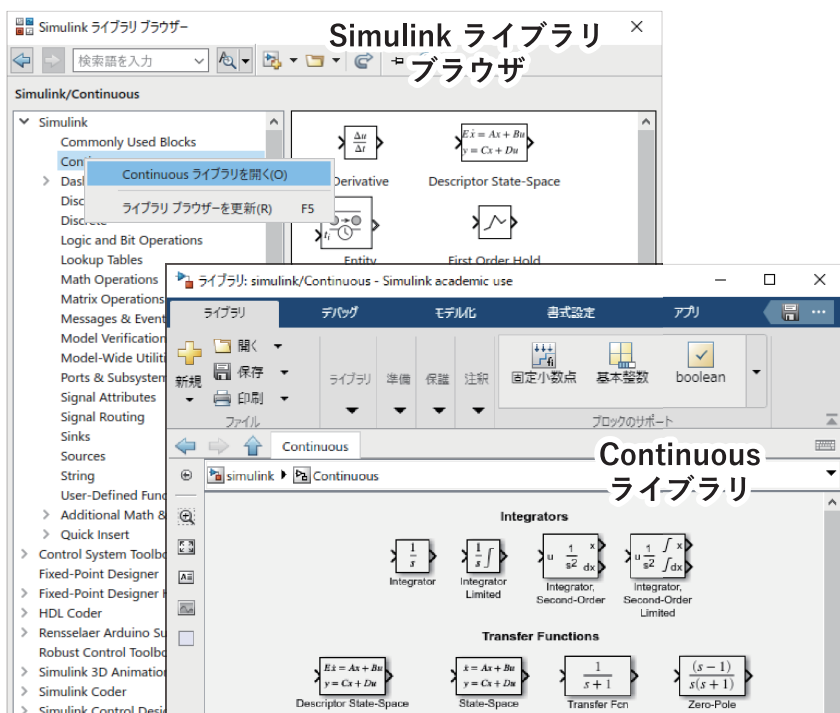


図 B.2 Continuous に含まれる Simulink ブロックの表示

表 B.1 Simulink ライブラリ

ライブラリ	説明
Commonly Used Blocks	一般的によく使用されるブロック群
Continuous	連続時間の伝達関数表現や状態空間表現、微分要素、積分要素などが含まれるブロック群
Dashboard	シミュレーションを操作するためのブロック群
Discontinuities	飽和関数、不感帯などの不連続関数が含まれるブロック群
Discrete	離散時間の伝達関数表現や状態空間表現、微分要素、積分要素などが含まれるブロック群
Logic and Bit Operations	論理演算やビット演算を行うブロック群
Lookup Tables	データの補間などの操作に関するブロック群
Math Operations	加減算器、比例要素、ゲインなどが含まれるブロック群
Messages & Events	メッセージベースの通信をモデル化するためのブロック群
Model Verification	モデルの検証に関するブロック群
Model-Wide Utilities	種々の用途を持つブロック群
Ports & Subsystems	入力・出力端子やサブシステムに関連するブロック群
Signal Attributes	型変換など信号属性を変更するブロック群
Signal Routing	複数信号のベクトル化、ベクトル信号の要素化、信号の分岐などのように信号の経路を指定するブロック群
Sinks	信号を表示したり、ファイルやワークスペースに信号をデータとして受け渡すブロック群
Sources	ステップ関数、正弦波などの信号を生成するブロック群
User-Defined Functions	カスタム関数のブロック群
Additional Math & Discrete	その他の数学関数と離散時間システムに関するブロック

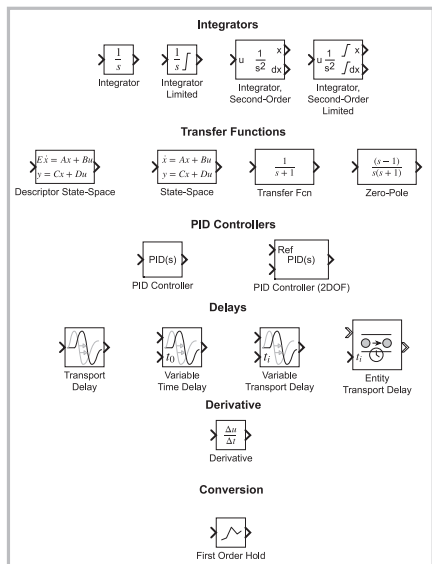


図 B.3 Continuous

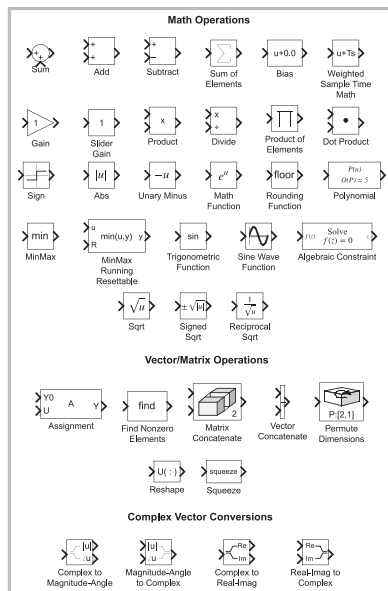


図 B.4 Math Operations

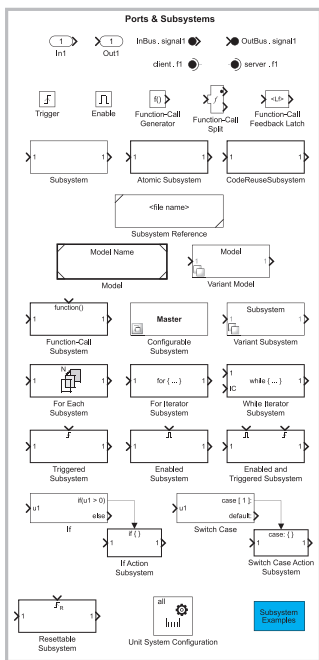


図 B.5 Ports & Subsystems

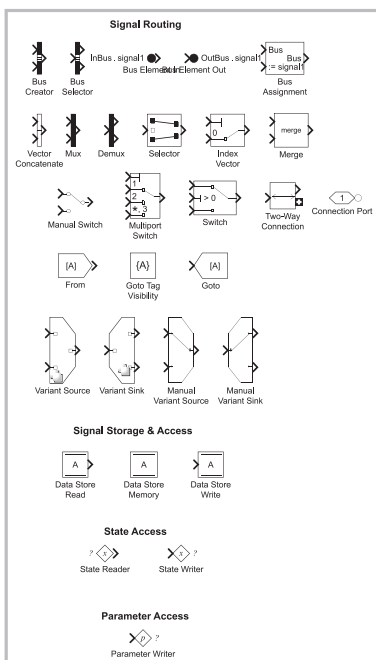


図 B.6 Signal Routing

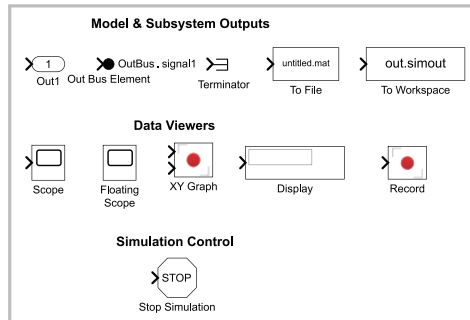


図 B.7 Sinks

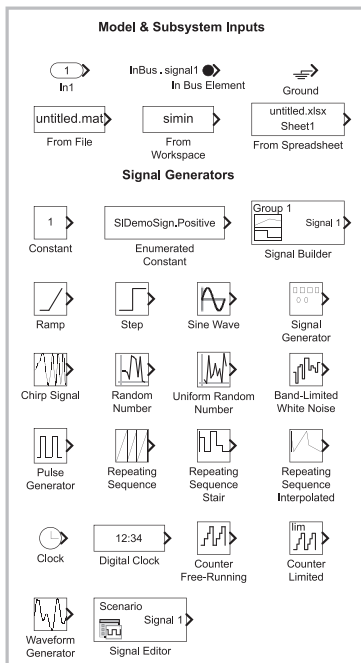


図 B.8 Sources

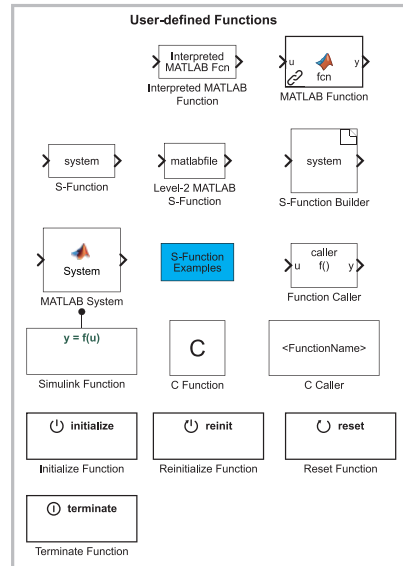


図 B.9 User-Defined Functions

## ■ ヘルプ機能

Simulink ブロックの使用方法を調べたい場合、Simulink ブロックを選択してダブルクリックをし、「ブロックパラメータ」のウィンドウを開く。そして、このウィンドウの最下部にある「ヘルプ」を選択するとヘルプブラウザが起動し、図 B.10 に示すように Simulink ブロックの説明が表示される。

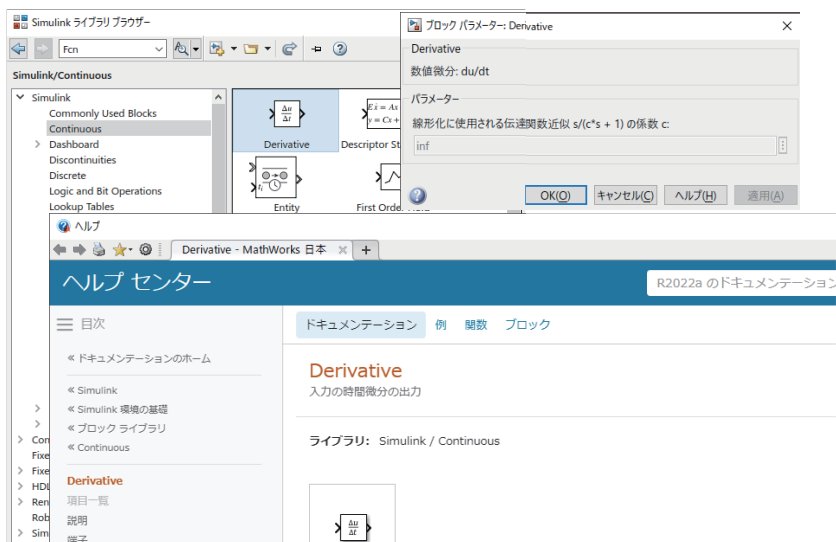


図 B.10 Simulink ブロックのヘルプ機能

## B.2 Simulink モデルの作成と実行

### ■ ソルバの設定

Simulink モデルの空白部分でマウスを右クリックして「モデルコンフィギュレーションパラメータ (Ctrl + E)」を選択するか、Simulink モデルのメニュー「モデル化/モデル設定」を選択すると、コンフィギュレーションパラメータのウィンドウが開くので、微分方程式の数値解法に関する設定を行うことができる。たとえば、図 B.11 のように「ソルバ」を設定すると、サンプリング周期 0.001 秒で 5 秒間、4 次のルンゲ・クッタ法により微分方程式を数値的に解くことができる。

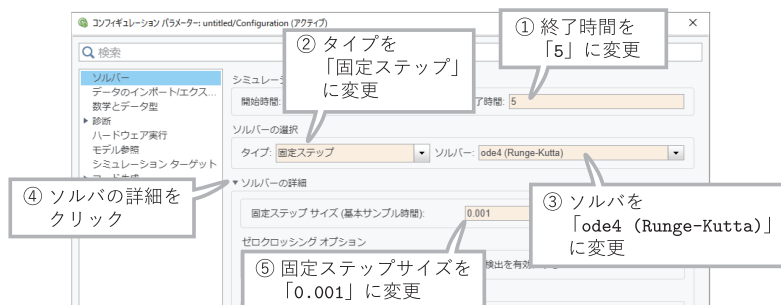


図 B.11 「コンフィギュレーションパラメータ」の「ソルバ」の設定

## ■ Simulink ブロックの配置

Simulink モデルに Simulink ブロックを配置するには、図 B.12 のように、Simulink ライブラリブラウザもしくは Simulink ライブラリで Simulink ブロックにマウスカーソルを合わせ、左ボタンをクリックしたまま Simulink モデルに移動すればよい<sup>†1</sup>。

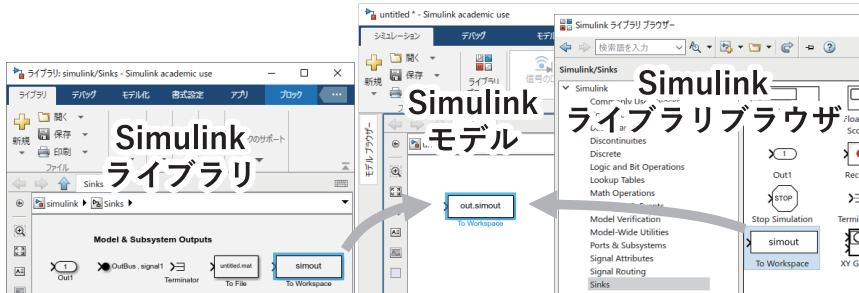


図 B.12 Simulink ブロックの配置

## ■ Simulink モデルの作成例

ライブラリ Sources, Sinks, Continuous, Math Operations から Simulink ブロックを Simulink モデルに移動し、必要に応じて Simulink ブロックを回転させる。つぎに、Simulink ブロックのブロックパラメータを表 B.2 のように変更する。なお、Simulink ブロック“Sum”のパラメータ「符号リスト」の意味については、表 B.3 を参照されたい。最後に、Simulink ブロックを結線し、“sample.slx”という名前で保存する。以上の手順を示すと、つぎのようになる。

表 B.2 Simulink ブロックのパラメータ設定

Simulink ブロック	変更するパラメータ
To Workspace	変数名：y, 保存形式：配列
To Workspace1	変数名：t, 保存形式：配列
To File	ファイル名：datafile, 変数名：output
Sum, Sum1	符号リスト： +-
Gain	ゲイン：5

<sup>†1</sup> Simulink モデルの空白部分にマウスカーソルを合わせて左クリックし、配置したいブロック名をキーボードから入力することで、Simulink ブロックを表示することもできる (R2015a 以降のバージョン)。使用したい Simulink ブロックがわかっている場合は、このようにした方が効率がよい。

表 B.3 Simulink ブロック “Sum” のパラメータ設定

符号リスト	+-	+ -	+--+
結果			

**ステップ 1** Simulink ライブラリブラウザから Simulink ブロックを移動する (図 B.13).

**ステップ 2** Simulink ブロック “Gain1” を選択した後、ショートカット「Ctrl + R」を 2 回操作し、時計まわりに 180 度回転する (図 B.14).

**ステップ 3** Simulink ブロック “To Workspace”, “To Workspace1”, “To File”, “Sum”, “Sum1” をダブルクリックし、ブロックパラメータを表 B.2 のように変更する (図 B.15).

**ステップ 4** Simulink ブロックの出力側にマウскарソルを合わせ、左ボタン (もしくは右ボタン) を押したまま線を伸ばし、別の Simulink ブロックの入力側と結線する (図 B.16). マウスの右ボタンにより線の途中から矢印を引き出すこ

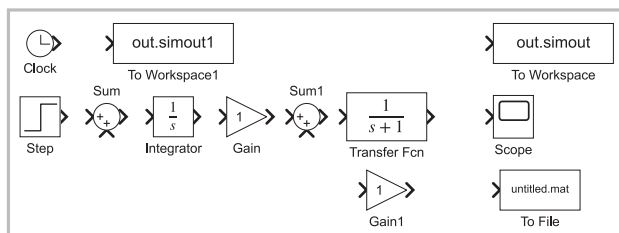


図 B.13 ステップ 1

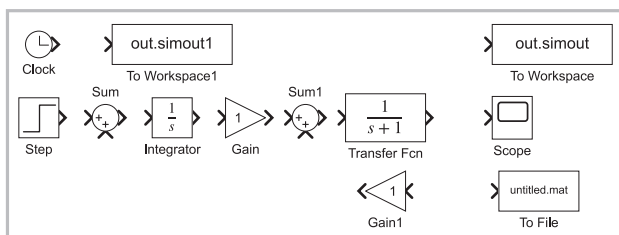


図 B.14 ステップ 2



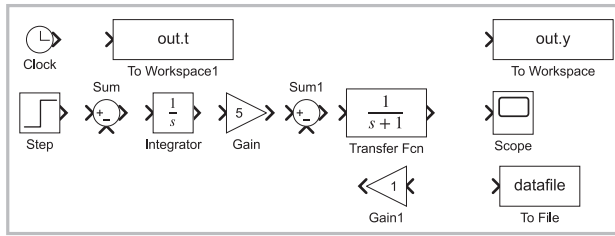


図 B.15 ステップ 3

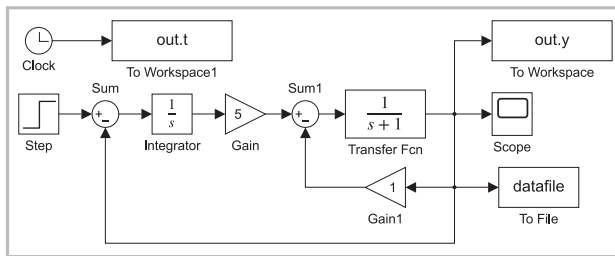


図 B.16 ステップ 4

とで、引き出し線を作成できる。

**ステップ 5** 「コンフィギュレーションパラメータ」の「ソルバ」を図 B.11 のように設定し、Simulink モデルを作業フォルダに“sample.slx”という名前で保存する。

## ■ Simulink モデルの実行

カレントディレクトリを Simulink モデル “sample.slx” が保存されたフォルダとした後、以下のいずれかの方法によりシミュレーションを実行することができる。

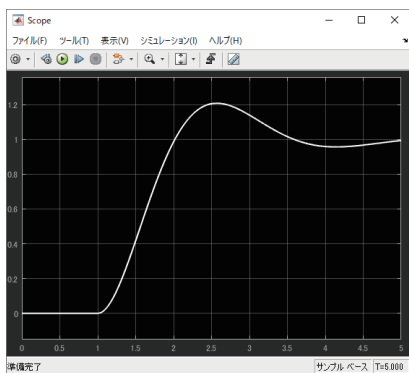
- 方法 1：Simulink モデルの▶をクリックする。
- 方法 2：Simulink コマンドウィンドウで

```
>> sim('sample');
```

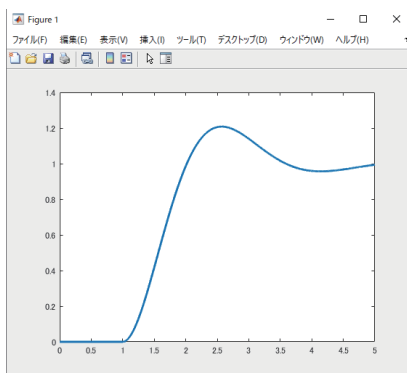
のように入力する。

シミュレーション結果は以下のいずれかの操作により表示することができる。

- Simulink ブロック “Scope” をダブルクリックすると、図 B.17 (a) のようにシミュレーション結果が表示される。



(a) “Scope”による表示



(b) 関数 plot による描画

図 B.17 Simulink モデルの実行結果の表示

- Simulink ブロック “To Workspace” によりワークスペースに `out` という変数名で構造体のデータが保存されている<sup>†2</sup>。コマンドウィンドウで

```
>> figure(1); plot(out.t,out.y,'LineWidth',2) ↵
```

と入力することにより、図 B.17 (b) のシミュレーション結果がフィギュアウィンドウに表示される。

- Simulink ブロック “To File” では、変数 `output` の保存形式はデフォルトの「時系列」としているため、時間データは `output.Time`、時間応答のデータは `output.Data` に格納されている。そのため、コマンドウィンドウで

```
>> load('datafile') ↵ ..... mat ファイル “datafile.mat” の読み込み
>> output ↵
    timeseries

一般的なプロパティ:
    Name: ''
    Time: [5001x1 double]
    TimeInfo: [1x1 tsdata.timemetaddata]
    Data: [5001x1 double]
    DataInfo: [1x1 tsdata.datametaddata]

その他のプロパティ, メソッド

>> figure(1); plot(output.Time,output.Data,'LineWidth',2) ↵ ..... グラフの描画
```

と入力することにより、図 B.17 (b) のシミュレーション結果が描画される。

<sup>†2</sup> R2019a 以降のバージョンでは、Simulink ブロック “To Workspace” の「保存形式」を「配列」に設定した場合でもデータが構造体として保存される。

## ■ Simulink モデルの出力データの形式 (構造体の回避)

先に説明したように、R2019a 以降のバージョンの標準の設定では、Simulink ブロック “To Workspace” の「保存形式」を「配列」に設定した場合でも、出力データがワークスペースに変数 `out` の構造体として保存される。構造体としてまとめられることを回避するには、図 B.18 のように、「コンフィギュレーションパラメータ」の「データのインポート/エクスポート」の設定を変更し、Simulink モデルを作業フォルダに “sample2.slx” という名前で保存すればよい。このように設定すると、図 B.19 のように Simulink ブロック “To Workspace”, “To Workspace1” の表示がそれぞれ `out.y`, `out.t` から `y`, `t` のように切り替わる。したがって、Simulink モデル “sample2.slx” によりシミュレーションを実行した後、コマンドウィンドウで

```
>> figure(1); plot(t,y,'LineWidth',2) <|
```

と入力すれば、図 B.17 (b) のシミュレーション結果が描画される。

本書では、図 B.18 のように「データのインポート/エクスポート」の設定を変更した例を示す。ただし、R2018b 以前のバージョンではこのような設定変更の必要はない。

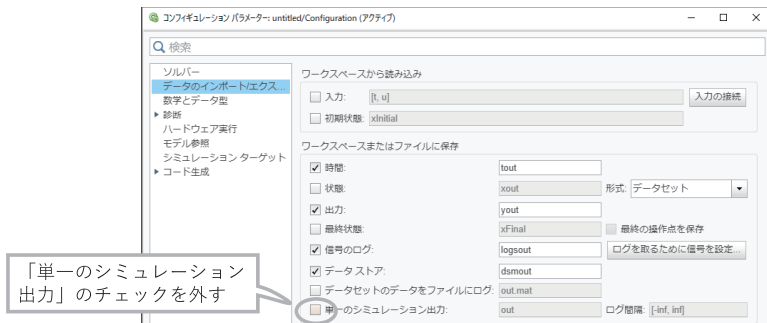


図 B.18 「コンフィギュレーションパラメータ」の「データのインポート/エクスポート」の設定

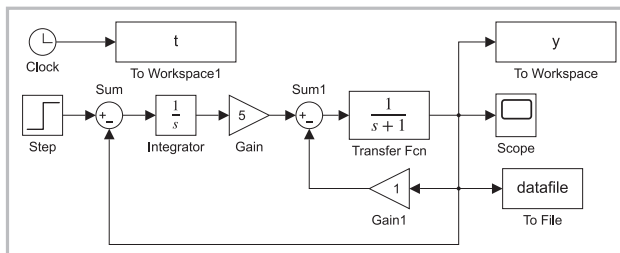


図 B.19 構造体の回避

## ■ Simulink モデルの図としての保存

作成した Simulink モデルは関数 `print` により図として保存することができる。たとえば、先のステップ 4 で作成した図 B.16 の Simulink モデル “`sample.slx`” を PDF ファイル “`model.pdf`” として保存するためには、Simulink モデルが保存されているフォルダをカレントディレクトリとした後、コマンドウィンドウで

```
>> sample; print('-s','model','-dpdf') ↵
```

または

```
>> sample; print -s -dpdf model ↵
```

のいずれかを入力する。同様に、`-dpdf` の代わりに `-djpeg` とすれば “`model.jpg`” という名前の JPEG ファイル、`-dpng` とすれば “`model.png`” という名前の PNG ファイル、`-dmeta` とすれば “`model.emf`” という名前の拡張メタファイル (Windows のみ) で保存される<sup>†3</sup>。

---

<sup>†3</sup> R2015b 以降のバージョンでは、Simulink モデルを EPS ファイルで保存することはできなくなった。