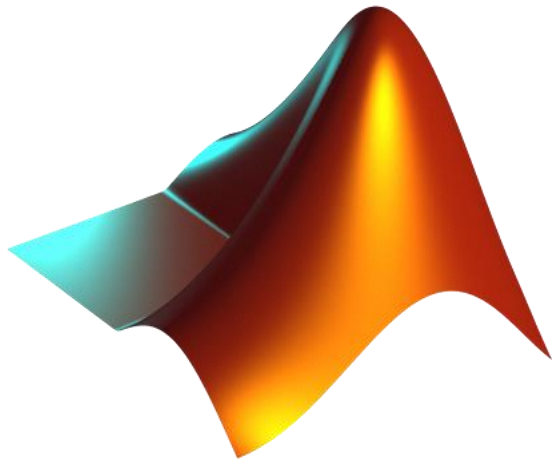


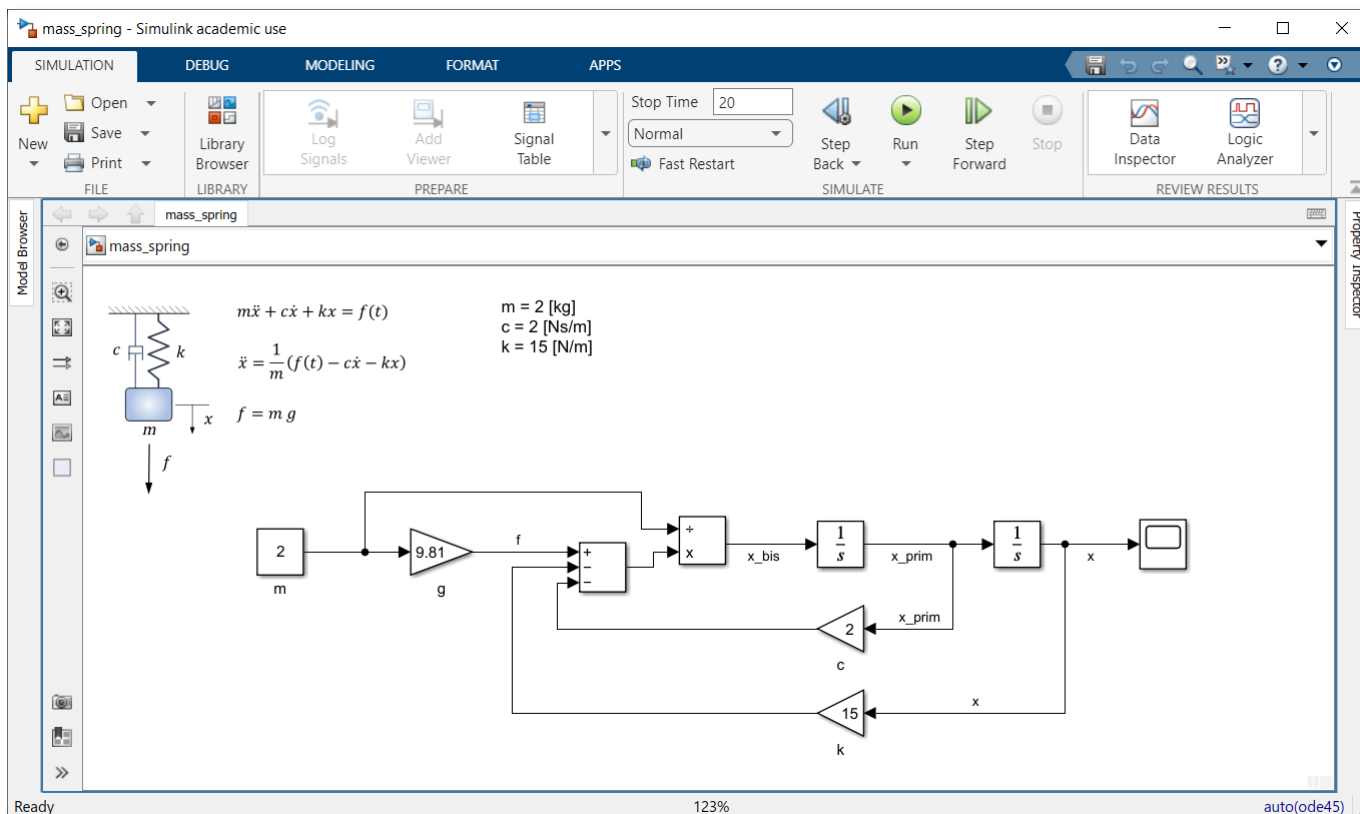
Programowanie w zastosowaniach inżynierskich

Dodatkowe narzędzia pakietu MATLAB

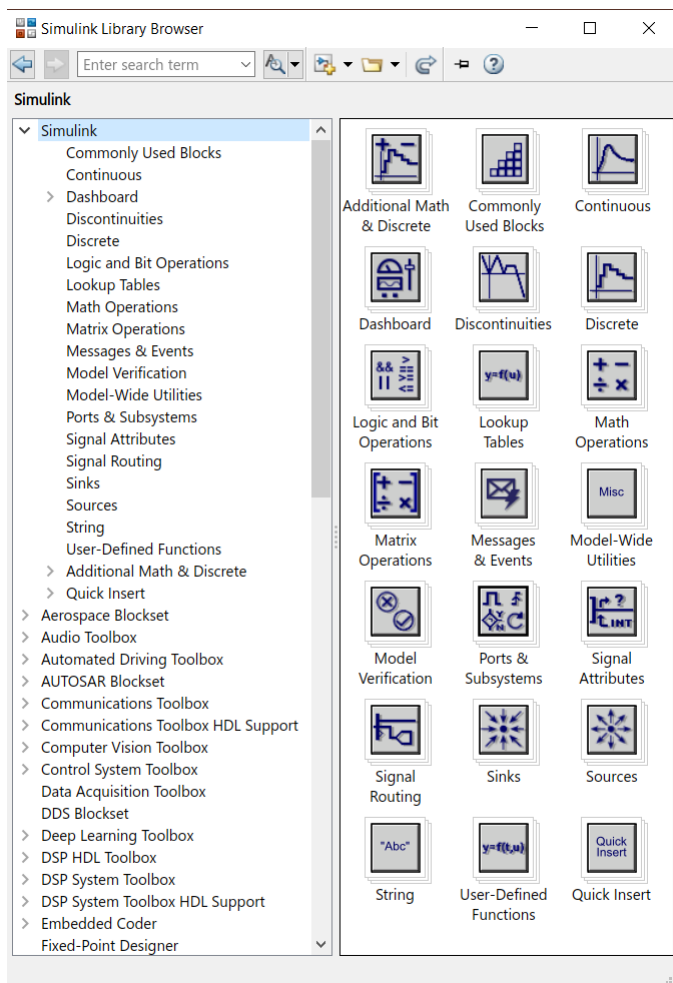
Modelowanie obiektów dynamicznych – Simulink
Obliczenia symboliczne



Simulink jest zintegrowanym ze środowiskiem Matlab interaktywnym narzędziem graficznym przeznaczonym do projektowania i symulacji złożonych systemów z wykorzystaniem modeli blokowych (metodyka *Model-Based Design*) bez konieczności tworzenia kodu. Model składa się z bloków (symbole graficzne) reprezentujących operacje oraz sygnałów (linie łączące bloki), które są przez nie przetwarzane.



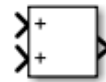
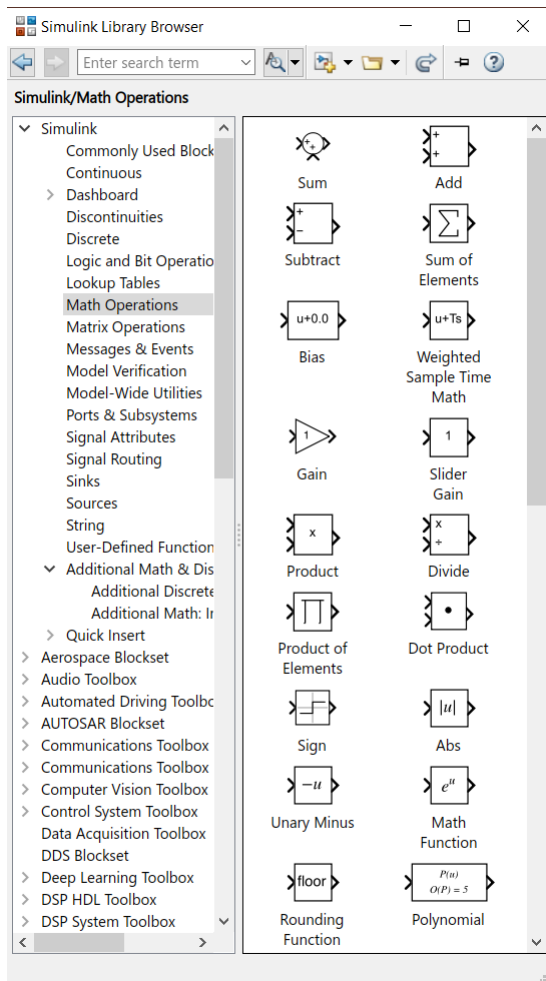
Simulink – przegląd podstawowych bloków



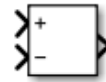
Wybrane kategorie z pakietu Simulink

- **Continuous** – modelowanie układów ciągłych,
- **Discrete** – modelowanie układów dyskretnych,
- **Logic and Bit Operations** – operacje logiczne i bitowe,
- **Math Operations** – podstawowe operacje i funkcje matematyczne,
- **Matrix Operations** – operacje macierzowe,
- **Signal Routing** – sterowanie przebiegiem sygnałów,
- **Sinks** – wizualizacja wartości sygnałów,
- **Sources** – źródła sygnałów,
- **User-Defined Functions** – funkcje definiowane przez użytkownika.

Math Operations



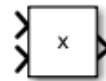
Add – dodaje sygnały wejściowe



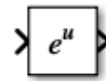
Subtract – odejmuje sygnały wejściowe



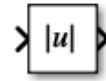
Gain – mnoży sygnał wejściowy przez stałą



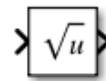
Product – mnoży lub dzieli sygnały wejściowe



Math function – wybrana funkcja (exp, log, pow,...)



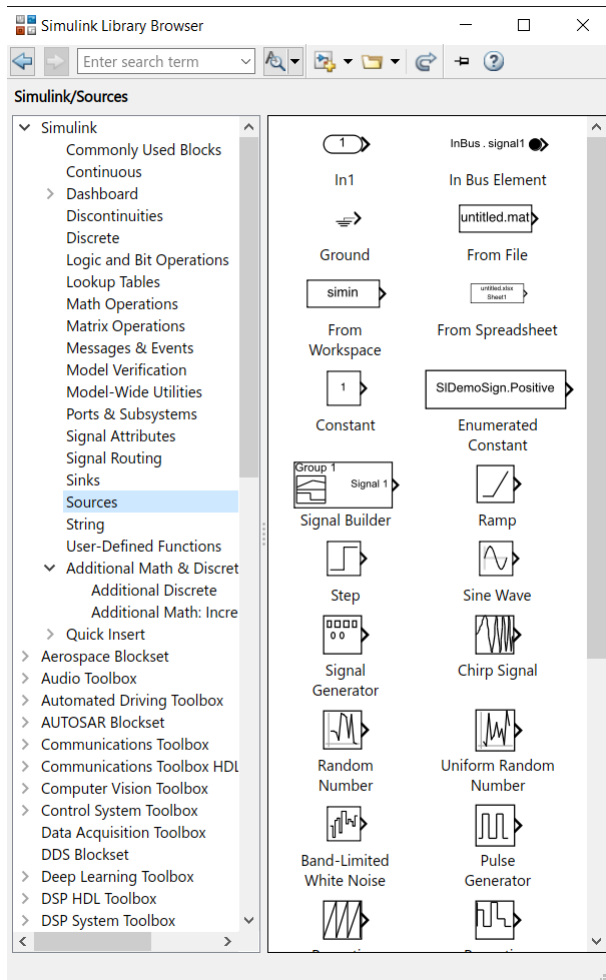
Abs – wartość bezwzględna



Sqrt – pierwiastek kwadratowy



Trigonometric Function – funkcje trygonometryczne



Constant – stała wartość



Ramp – wartość rosnąca jednostajnie



Step – skok w określonej chwili czasu



Sine Wave – sygnał sinusoidalny



Random number – wartości losowe o rozkładzie Gaussa)



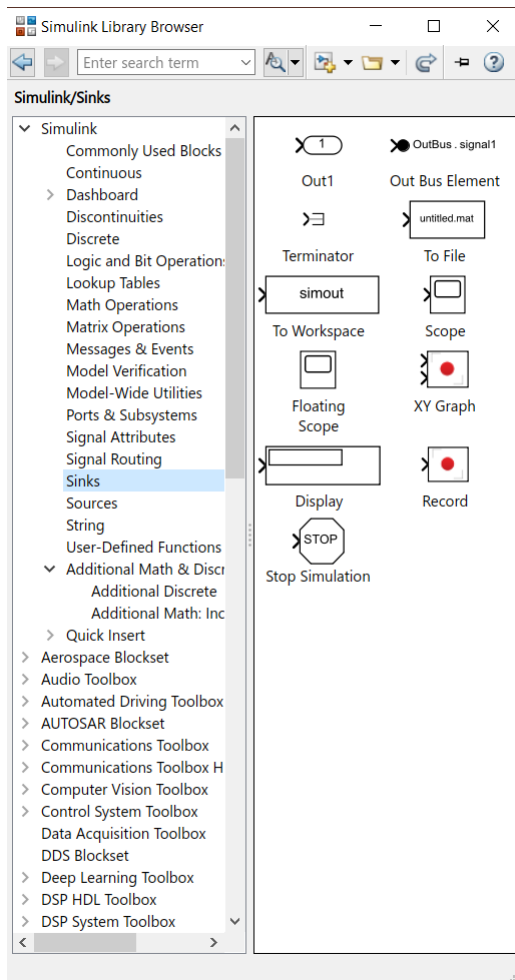
Uniform Random Number – wartości losowe o rozkładzie jednostajnym



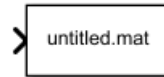
Pulse Generator – sygnał pulsujący



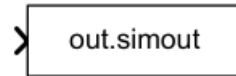
Repeating Sequence – powtarzalna sekwencja



Terminator – kończy niepołączony sygnał wyjściowy



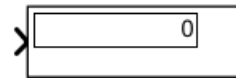
To File – zapisuje wartości wejściowe do zmiennej w pliku MAT



To Workspace – zapisuje wartości wejściowe do zmiennej w obszarze roboczym Matlaba



Scope – wykreśla sygnał wejściowy



Display – wyświetla aktualną wartość sygnału wejściowego



Record – zapisuje i przechowuje wartości sygnałów wejściowych



Stop Simulation – zatrzymuje symulację gdy wartość wejściowa jest różna od 0

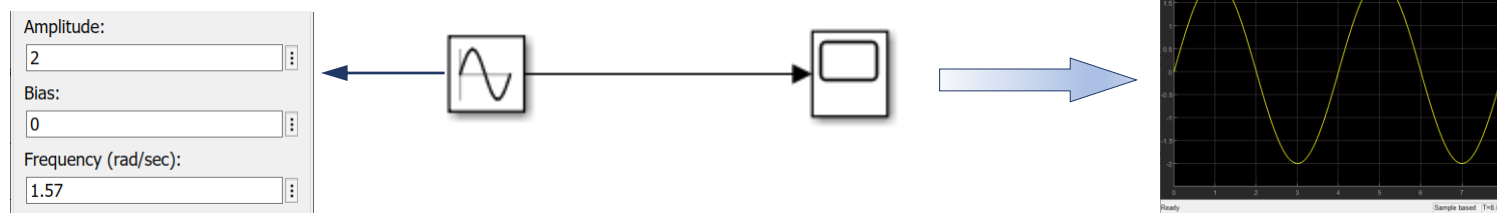
Konfigurowanie bloków

Każdy blok ma określony zestaw parametrów (menu podręczne → Block Parameters lub podwójne kliknięcie). Dodatkowo każdy blok może być formatowany (czcionka, kolor, orientacja, ikona, itp.), można określić jego nazwę oraz zmienić rozmiary.

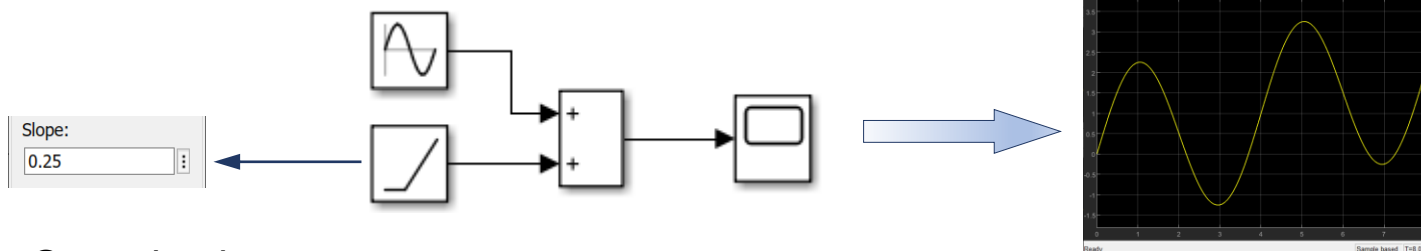
The image illustrates the configuration of a block in a software environment. On the left, the 'Block Parameters: Add' dialog box is shown, with the 'List of signs' field containing '+-|+'. Below this, a legend explains the signs: '+' for signal without sign change, '-' for sign change, and '|' for spacing. On the right, the software's ribbon interface is shown with several options highlighted by red boxes and arrows pointing to a 'Suma sygnałów' block. The highlighted options are: 'Arial' (font size 12) for 'Rozmiar czcionki', 'Background', 'Foreground', and 'Shadow' for 'Kolory i cień', 'Horizontal Flip' and 'Vertical Flip' for 'Odbicie w poziomie i obrót', and 'Flip Name' for 'Polozenie nazwy'. The 'Suma sygnałów' block is shown in two orientations: a blue-bordered 'Add' block and a grey-bordered 'Suma sygnałów' block.

Zadanie: Wygenerować sygnał sinusoidalny o amplitudzie 2 i częstotliwości 1.57 rad/s, stale rosnący o 0.25 na sekundę, zaburzony sygnałem sinusoidalnym o amplitudzie 0.1 i częstotliwości 31 rad/s.

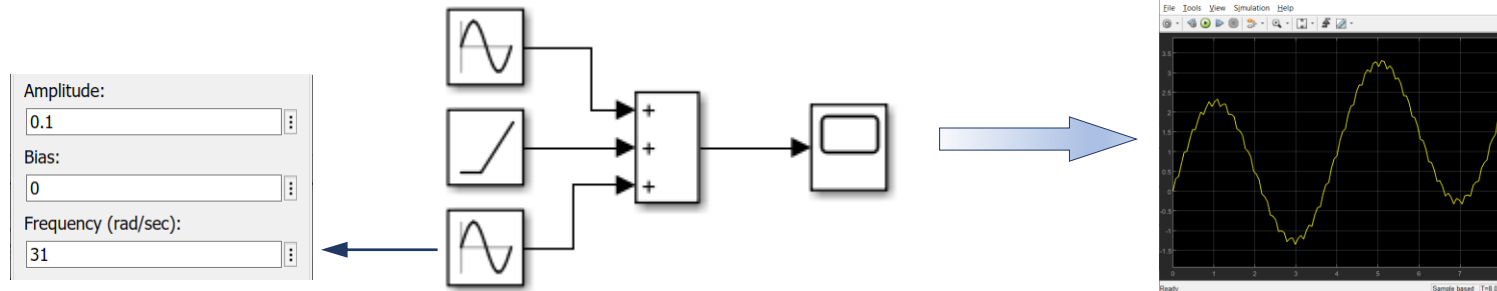
Krok 1: Sygnał sinusoidalny



Krok 2: Rosnący sygnał sinusoidalny



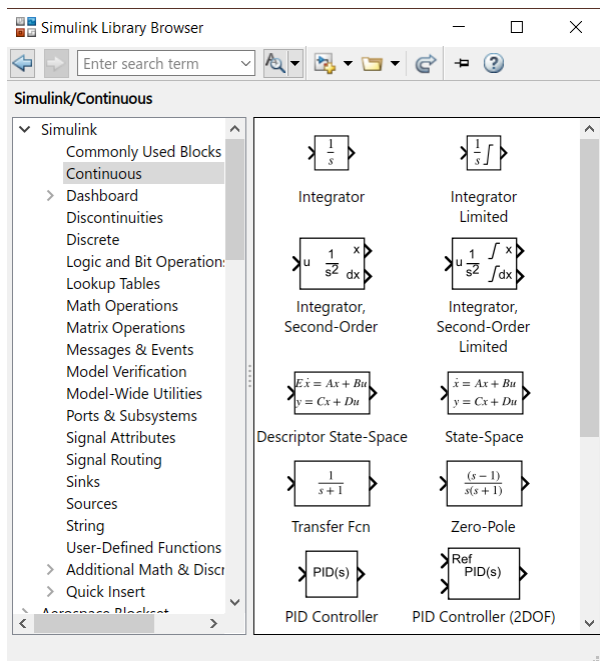
Krok 3: Sygnał zaburzony



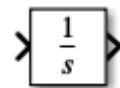
Układ dynamiczny to układ, którego stan zmienia się w czasie. Układy tego typu są modelowane matematycznie za pomocą układów równań różniczkowych.

Równanie różniczkowe jest równaniem, które określa zależności pomiędzy pewną nieznaną funkcją a jej pochodnymi.

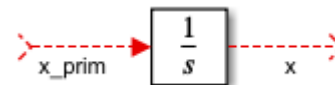
Pochodna jest miarą tempa zmian wartości funkcji względem zmian jej argumentów (definicja nieformalna).



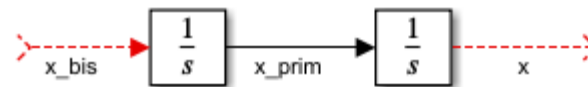
Blok całkujący



Integrator – całkuje sygnał wejściowy (operacja odwrotna do różniczkowania)



Uwaga: pojedynczy blok całkujący na podstawie pochodnej rzędu n wyznacza pochodną rzędu $n - 1$.



Warunek początkowy jest parametrem bloku całkującego.

Modelowanie układów dynamicznych

Tworzenie modelu układu dynamicznego (ciągłego)

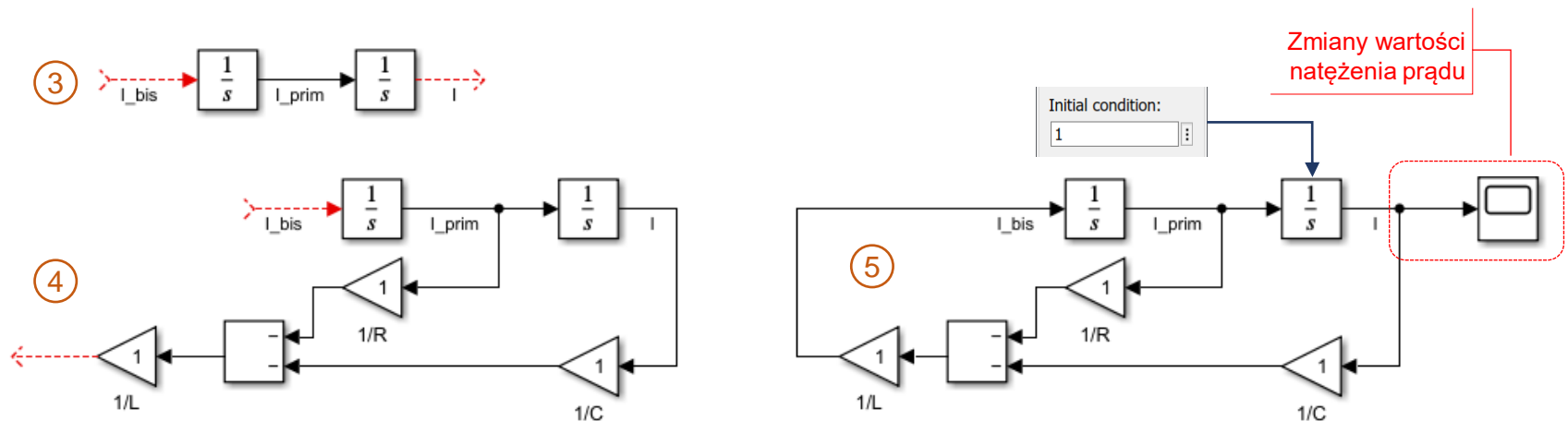
1. Ustal rząd równania (najwyższy rząd pochodnej).
2. Przekształć równanie przenosząc pochodną najwyższego rzędu na lewą stronę, a pozostałe składniki na prawą.
3. Ustal liczbę niezbędnych bloków całkujących (zazwyczaj równa rzędowi równania), wprowadź je do modelu, dodaj sygnały.
4. Zbuduj prawą stronę równania różniczkowego.
5. Wynik połącz z wejściem pierwszego bloku całkującego, ustal warunki początkowe.

Uwaga:

$$\frac{dx}{dt} = \dot{x}, \quad \frac{d^2x}{dt^2} = \ddot{x}$$

Przykład (zmiany natężenia prądu w obwodzie RLC)

$$L \frac{d^2 I}{dt^2} + R \frac{dI}{dt} + \frac{1}{C} I = 0 \Rightarrow \frac{d^2 I}{dt^2} = \frac{1}{L} \left(-\frac{1}{R} \frac{dI}{dt} - \frac{1}{C} I \right) \Rightarrow \ddot{I} = \frac{1}{L} \left(-\frac{1}{R} \dot{I} - \frac{1}{C} I \right) \quad \textcircled{2}$$

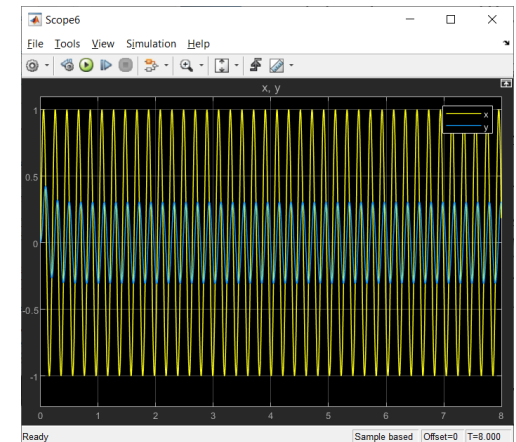
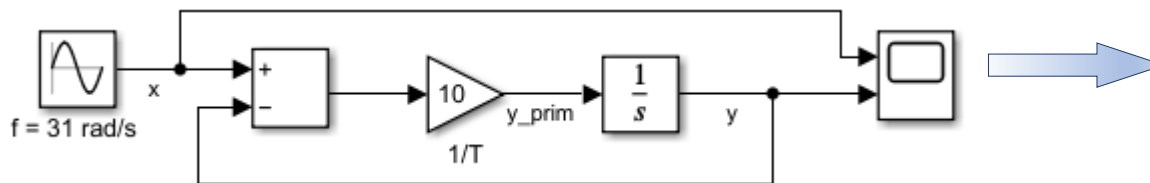
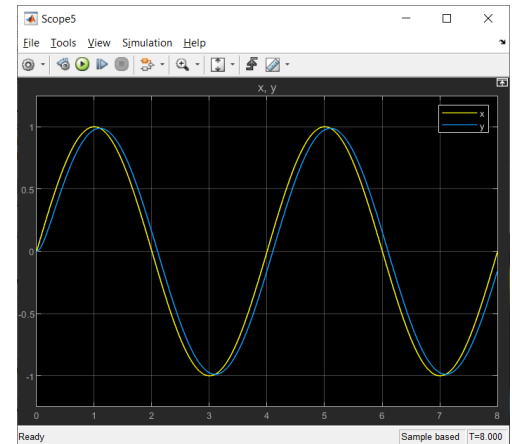
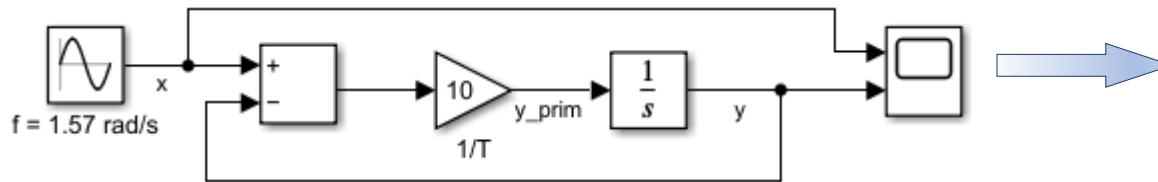


Filtr dolnoprzepustowy RC

Filtr dolnoprzepustowy (ang. *low-pass filter*) jest układem przetwarzającym sygnały, który przepuszcza sygnały o częstotliwości poniżej tzw. częstotliwości odcięcia (pasmo przepustowe) i tłumi składowe o częstotliwościach wyższych (pasmo zaporowe).

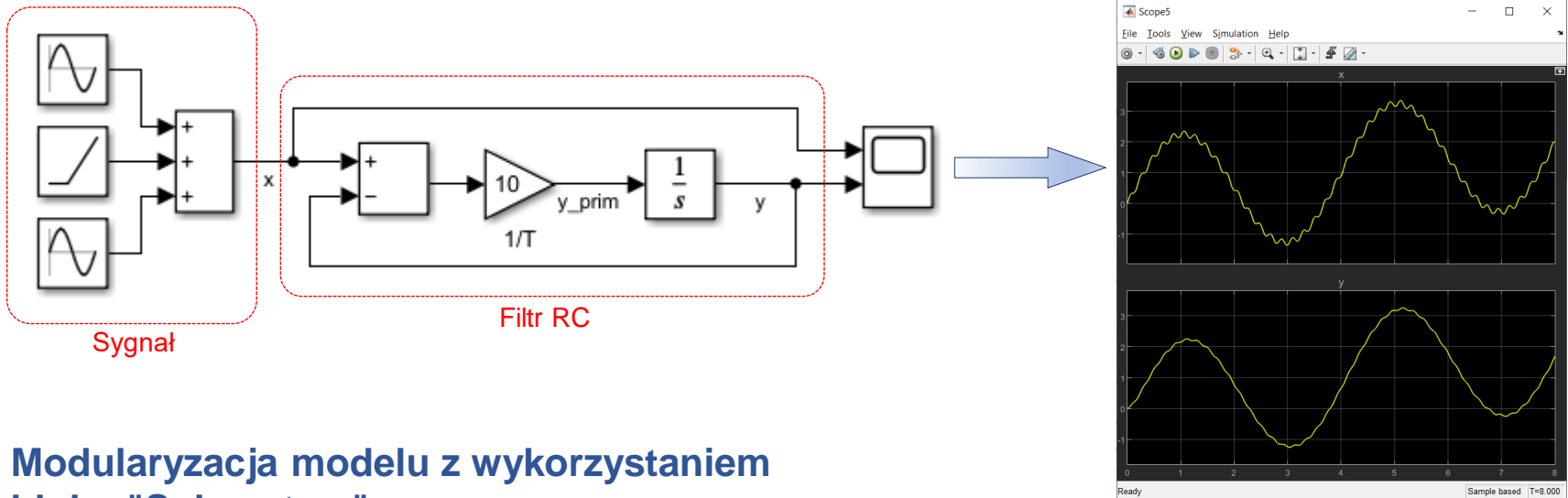
$$y + T \frac{dy}{dt} = x \quad \Leftrightarrow \quad \dot{y} = \frac{1}{T} (x - y)$$

x – sygnał wejściowy, y – sygnał wyjściowy,
 T - stała czasowa (parametr filtra)

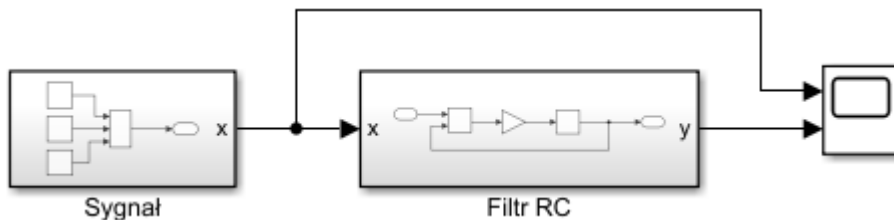


Filtr dolnoprzepustowy RC

Test filtra RC dla zaszumionego sygnału sinusoidalnego (przykład s.8).

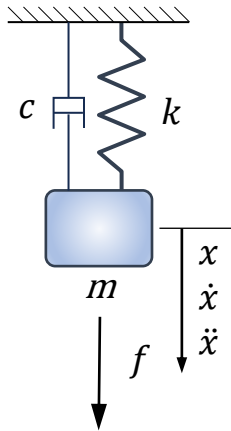


Modularyzacja modelu z wykorzystaniem bloku "Subsystem"



Uwaga: blok Subsystem znajduje się w grupie Simulink->Ports & Subsystems. Blok można utworzyć z istniejących elementów zaznaczając fragment modelu i wybierając opcję "Create Subsystem from Selection" z menu podręcznego.

Układ masa-sprężyna



m – masa ciała

k – współczynnik sprężystości

c – współczynnik tłumienia

x, \dot{x}, \ddot{x} – położenie, prędkość i przyspieszenie ciała

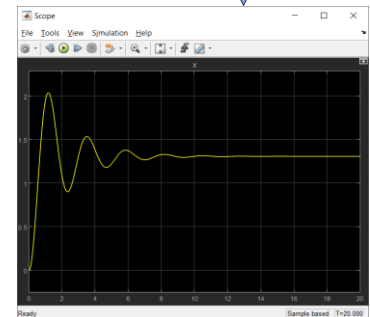
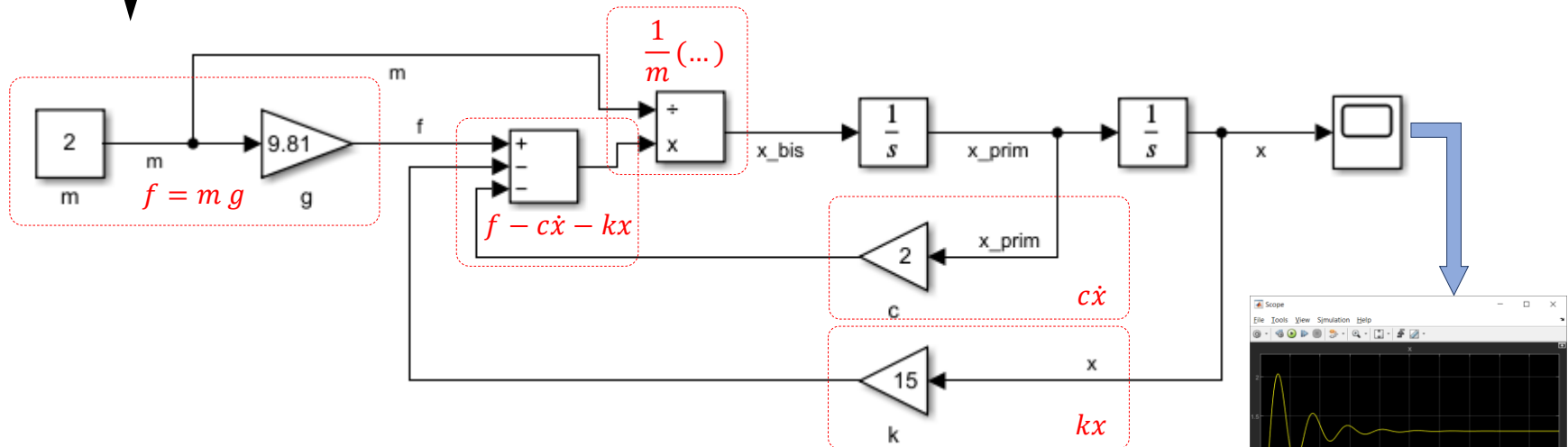
f – siła zewnętrzna (w przykładzie grawitacja)

$$m\ddot{x} + c\dot{x} + kx = f(t)$$

$$\ddot{x} = \frac{1}{m}(f(t) - c\dot{x} - kx)$$

$$f = mg$$

$$x = 0, \dot{x} = 0$$



Uwaga: model przedstawia zachowanie układu po zawieszeniu masy "m" na sprężynie, na którą nie działa żadna siła; symulacja rozpoczyna się w chwili zwolnienia masy, która zaczyna opadać pod wpływem siły grawitacji.

Obliczenia numeryczne i symboliczne

Obliczenia numeryczne – metody rozwiązywania problemów matematycznych wykorzystujące wyłącznie działania na liczbach. Uzyskiwane wyniki na ogół są przybliżone, jednak zazwyczaj dokładność obliczeń może być dobrana do potrzeb.

Obliczenia symboliczne – metody rozwiązywania problemów matematycznych wykorzystujące operacje na symbolach. Uzyskiwane rozwiązania mają postać równań lub funkcji opisujących dokładne rozwiązanie problemu.

Rozwiązanie numeryczne

Rozwiązanie symboliczne

równanie algebraiczne $x^2 + 2x + a = 0$

x niewiadoma, a parametr

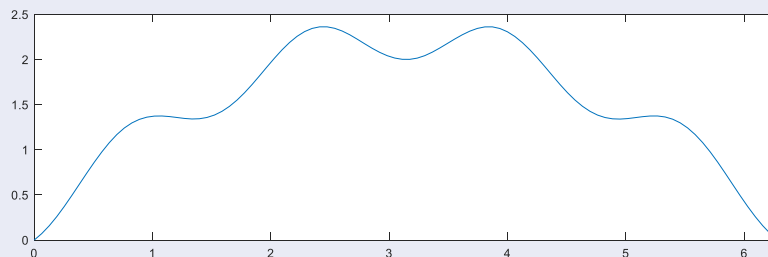
dla $a = 5$: $x_1 \approx -3,44949$, $x_2 \approx 1,44949$

$$x = \pm\sqrt{1-a} - 1$$

równanie różniczkowe $\dot{y}(t) = \sin(at) + \cos(bt)$, $y(0) = 0$

$y(t)$ niewiadoma funkcja, a, b parametry

dla $a = 4$ i $b = 0.5$ w czasie $t \in [0, 2\pi]$



$$y(t) = \frac{1}{a} - \frac{\cos(at)}{a} + \frac{\sin(bt)}{b}$$

Symbolic Math Toolbox udostępnia zestaw funkcji do przeprowadzania obliczeń symbolicznych. Najważniejsze zastosowania:

- algebra liniowa,
- rachunek różniczkowy i całkowy,
- rozwiązywanie równań algebraicznych i różniczkowych,
- upraszczanie i transformacja wyrażeń matematycznych.

Tworzenie obiektów symbolicznych

<i>Instrukcja</i>	<i>Przykład</i>	<i>Opis</i>
<code>syms n1 n2 ... nN</code>	<code>syms a b c</code>	Tworzy skalarne zmienne symboliczne o nazwach określonych przez $n1, n2, \dots, nN$
<code>syms n1 n2 ... nN set</code>	<code>syms q w real</code>	Tworzy skalarne zmienne symboliczne przypisując im ograniczenia wyspecyfikowane przez <i>set</i> , dozwolone wartości: <i>real, positive, integer, rational</i>
<code>syms fun(v1,v2,...,vN)</code>	<code>syms f(x,y)</code>	Tworzy symboliczną funkcję <i>fun</i> oraz symboliczne zmienne $v1, v2, \dots, vN$ reprezentujące jej argumenty
<code>n = sym('n')</code>	<code>x=sym('x')</code>	Tworzy skalarną zmienną symboliczną o nazwie <i>n</i>
<code>n = sym('n', set)</code>	<code>q=sym('q', real)</code>	Tworzy skalarną zmienną symboliczną z ograniczeniami
<code>v = sym(num)</code>	<code>f=sym(1/7)</code>	Konwertuje wartość numeryczną <i>num</i> na symboliczną wartość numeryczną

Wyrażenie symboliczne jest dowolnym wyrażeniem zawierającym zmienne symboliczne. Do budowy takich wyrażen mogą być wykorzystane standardowe operatory i wbudowane funkcje Matlab-a.

Manipulowanie wyrażeniami symbolicznymi

<i>Funkcja</i>	<i>Przykład</i>	<i>Opis</i>
<code>collect(exp, v)</code>	<code>collect((x-5)^2+(y-2)^2, x)</code> $x^2 - 10x + (y-2)^2 + 25$	Grupuje współczynniki zmiennej v
<code>expand(exp)</code>	<code>expand(sin(x+y))</code> $\cos(x) * \sin(y) + \cos(y) * \sin(x)$	Rozwija wyrażenie exp
<code>fplot(exp, r)</code>	<code>fplot(2*x^2+5, [-3, 3])</code>	Wykreśla funkcję jednej zmiennej opisaną przez exp w przedziale r (dwuelementowy wektor)
<code>poly2sym(c, var)</code>	<code>poly2sym([2, 3, 0, 1], y)</code> $2*y^3 + 3*y^2 + 1$	Tworzy wielomian zmiennej var o współczynnikach określonych wektorem c
<code>simplify(exp)</code>	<code>simplify((x^2+5*x+6)/(x+2))</code> $x+3$	Upraszcza wyrażenie exp
<code>subs(exp, v1, v2)</code>	<code>subs(1/2+sin(y), y, pi/4)</code> $2^{(1/2)}/2 + 1/2$	Podstawia za zmienną symboliczną $v1$ wartość $v2$ (obiekt symboliczny, liczba) w wyrażeniu exp
<code>vpa(exp, d)</code>	<code>vpa(sin(x)+1/7, 5)</code> $\sin(x) + 0.14286$	Wyznacza wartości stałych symbolicznych z dokładnością do d miejsc po kropce

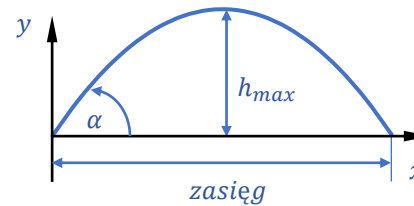
Rozwiązywanie równań algebraicznych

```
S = solve(eqn, var)
```

funkcja rozwiązuje równanie opisane wyrażeniem *eqn* ze względu na zmienną *var*.

Przykład. Rzut ukośny

```
>> syms x(t) y(t) v0 a g
>> x(t) = v0*t*cos(a);
>> y(t) = v0*t*sin(a)-g*t^2/2;
>> T = solve(y==0,t)
T =
    0
    (2*v0*sin(a))/g
>> T = T(2);
>> z = subs(x, t, T)
z =
    (2*v0^2*cos(a)*sin(a))/g
>> h = subs(y, t, T/2)
h =
    (v0^2*sin(a)^2)/(2*g)
```



$$x(t) = v_0 t \cos \alpha$$
$$y(t) = v_0 t \sin \alpha - \frac{g t^2}{2}$$

```
% uproszczenie wyrażenia na "z"
```

```
>> z = simplify(z)
```

```
z =
    (v0^2*sin(2*a))/g
```

```
% wyrażenie w czytelnej formie
```

```
>> pretty(z)
```

$$\frac{v_0^2 \sin(2 a)}{g}$$

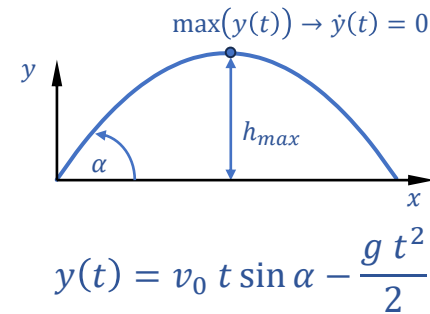
```
Df = diff(f, var, n)
```

wyznacza pochodną funkcji f rzędu n ze względu na zmienną var

Przykład. Rzut ukośny (alternatywne wyznaczenie h_{max})

```
>> syms y(t) v0 a g
>> y(t) = v0*t*sin(a)-g*t^2/2;
>> T = solve(diff(y,t)==0,t)
T =
    (v0*sin(a))/g
>> h = y(T)
h =
    (v0^2*sin(a)^2)/(2*g)
>> pretty(h)
```

$$\frac{v_0^2 \sin^2(a)}{2g}$$



Rozwiązywanie równań różniczkowych

`S = dsolve(eqn, cond)`

Funkcja rozwiązuje równanie różniczkowe `eqn` z warunkami początkowymi `cond`

Przykład. Układ masa-sprężyna

```
>> syms x(t)
```

```
>> syms m c k g positive
```

```
>> E = m*diff(x,t,2)+c*diff(x,t)+k*x == m*g
```

```
>> Dx(t) = diff(x(t),t);
```

```
>> X(t) = dsolve(E, [x(0)==0, Dx(0)==0]);
```

```
>> X(t) = subs(X,[m,c,k,g], [2,2,15,9.81]);
```

```
>> fplot(X(t), [0,20])
```

