

OpenModelica - lab 1

Paweł Łojek

1. Instalacja

Maszyna wirtualna z najnowszą wersją OpenModelica została przekazana przez prowadzącego, jeśli nie macie Państwo do niej dostępu, można pobrać starszą wersję ze strony https://build.openmodelica.org/omc/builds/virtual_machines/old_releases/1.9.6-x86_64/ Maszynę należy zaimportować za pomocą oprogramowania *VirtualBox* lub *VMware Player*.

Hasło do wirtualnej maszyny jest takie samo jak login – w przypadku maszyny od prowadzącego będzie to *modelica*, a w przypadku maszyny pobranej ze strony dewelopera oprogramowania – *openmodelica*

Oprócz tego możliwe jest zainstalowanie *OpenModelica*. Pliki instalacyjne/informacje na temat instalacji można znaleźć na stronie <https://openmodelica.org/download/>

Uwaga: w systemie macOS można zainstalować OpenModelica przez *MacPorts*, jednak program nie działa poprawnie i znacznie wygodniejsze jest wykorzystanie wirtualnej maszyny.


2. Podstawy

Po uruchomieniu maszyny wirtualnej i zalogowaniu się, na pulpicie można znaleźć następujące pliki:

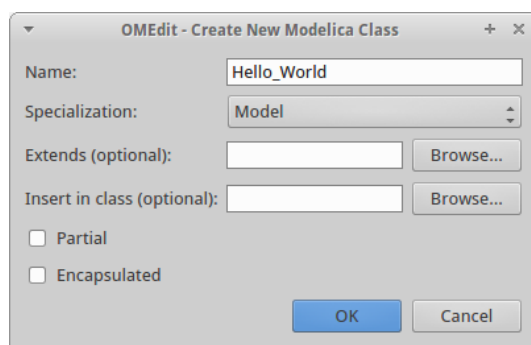
OMEdit – program z interfejsem graficznym który będzie wykorzystywany na zajęciach,

OMShell – konsola języka Modelica,

OMNotebook – „zeszyt” zawierający interaktywny samouczek, pozwala na tworzenie nowych zeszytów.






Program uruchamia się za pomocą ikony *OMEdit*. Nowy plik można utworzyć za pomocą przycisku , lub klikając na **File** » **New Modelica Class**. W oknie tworzenia nowej klasy należy tylko nadać jej nazwę (rys. 1).

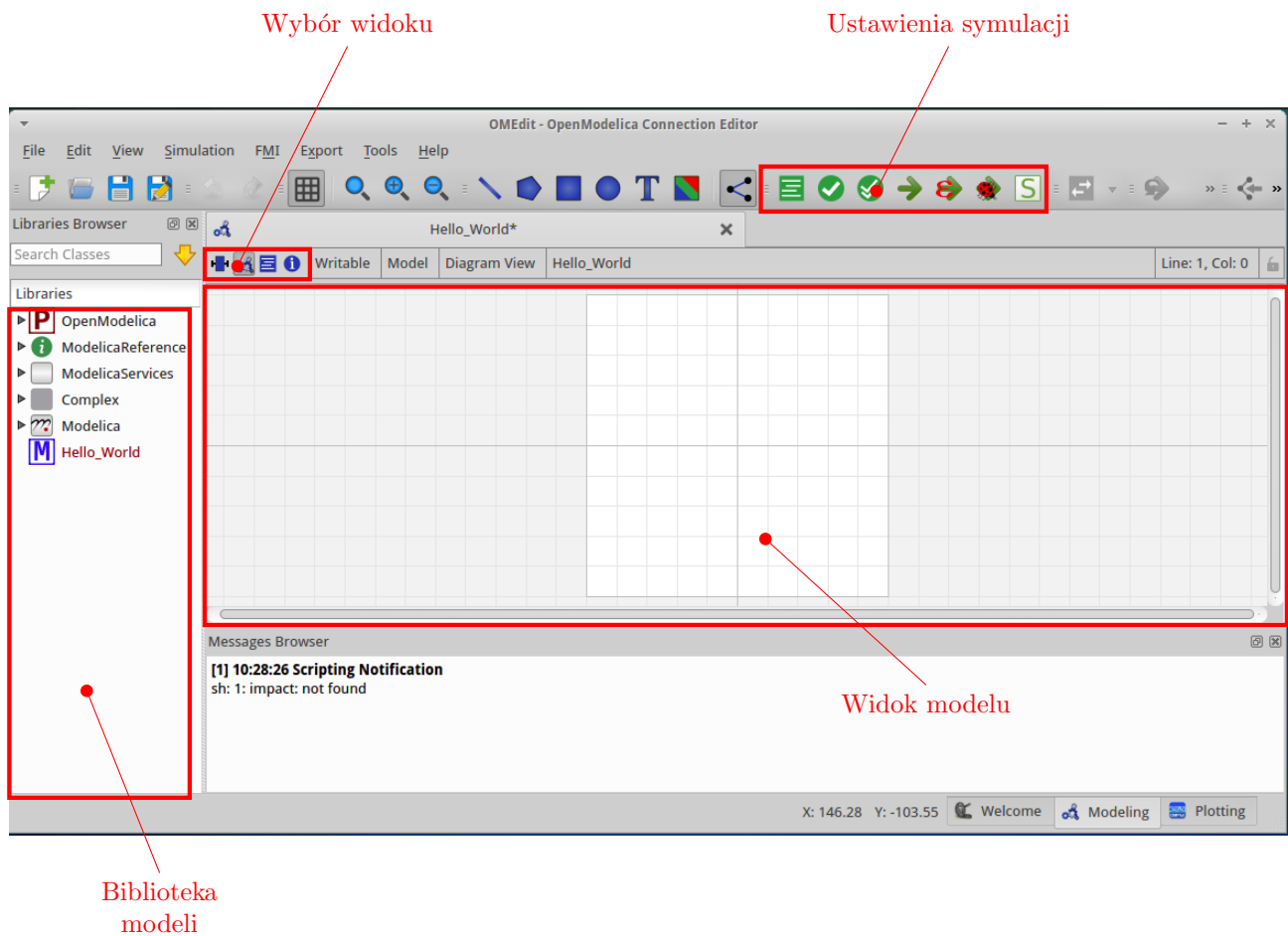
Po uruchomieniu i utworzeniu nowego modelu można zobaczyć interfejs pokazany na rys. 2.



Rysunek 1. Nowy model OpenModelica

Najważniejsze przyciski zostały opisane poniżej

-  – tryb graficzny
-  – tryb tekstowy
-  – sprawdzenie modelu
-  – przeprowadzenie symulacji
-  – ustawienia symulacji



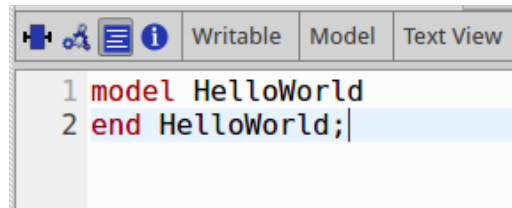
Rysunek 2. Interfejs programu OMEdit

2.1. Równanie różniczkowe zwyczajne

OpenModelica zostanie wykorzystana do rozwiązania równania różniczkowego zwyczajnego:

$$\dot{x} = -a \cdot x(t) \quad (1)$$

dla warunku początkowego $x(0) = 2$, $a = 1$ i $t \in < 0, 5 >$ za pomocą metody Rungego-Kutty. Na początku trzeba utworzyć nowy model, i nadać mu nazwę, na przykład *HelloWorld*. Następnie trzeba przejść do trybu tekstowego (rys. 3)



Rysunek 3. Nowy model (tryb tekstowy)

Następnie trzeba uzupełnić kod o definicję zmiennej x , parametru a i o równanie które chce się rozwiązać.

Zmienną w równaniu definiuje się za pomocą komendy `Real x(start=2)`. Każda zmienna w języku Modelica ma atrybut `start`, którego wartość domyślna wynosi 0. Chcąc nadać własne warunki początkowe, należy go nadpisać. **Wszystkie zmienne definiowane w OpenModelica są funkcjami czasu, który jest przypisany do zmiennej time.**

Stały atrybut a definiuje się za pomocą komendy `parameter Real a = 1;`. Przedrostek `parameter` oznacza, że jest to stała, która może się zmieniać między symulacjami.


Ostatnią rzeczą, którą należy zdefiniować jest równanie (1). Może to zostać zrobione za pomocą komendy `equation`:

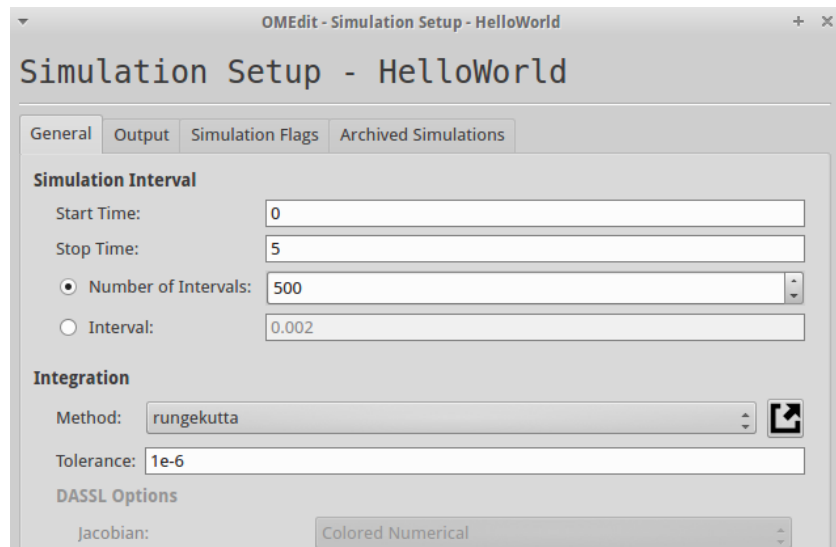
```
equation
  der(x)=-a*x;
```

gdzie `der(x)` oznacza $\frac{dx}{dt}$.

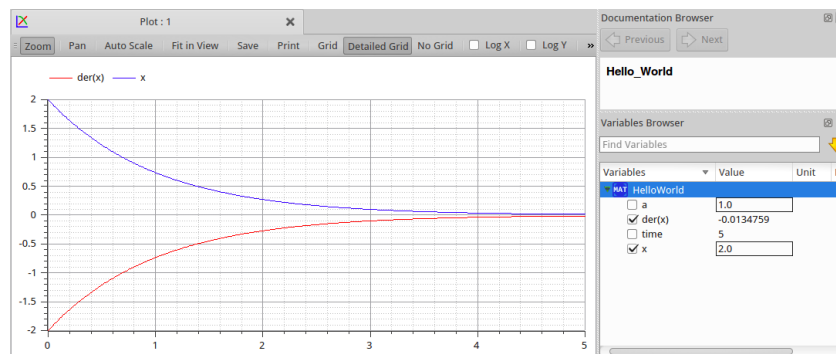
Ostatecznie cały program powinien wyglądać następująco:

```
model HelloWorld
  Real x(start=2);
  parameter Real a = 1;
  equation
    der(x)=-a*x;
end HelloWorld;
```

Czas symulacji jest definiowany w ustawieniach () , podobnie jak metoda rozwiązywania (rys. 4). Po wpisaniu czasu, wybraniu metody i ewentualnej zmianie innych ustawień można uruchomić symulację.



Rysunek 4. Ustawienia symulacji



Rysunek 5. Wyniki symulacji

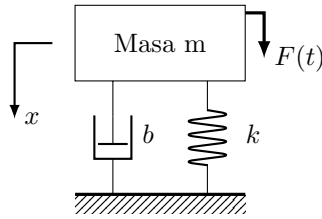
2.2. Ćwiczenie

Rozwiązać równanie

$$\frac{dy}{dt} = t^2 - y \cdot \cos(y^2) \quad (2)$$

dla warunku początkowego $y(0) = 0$ i $t \in \langle -2, 2 \rangle$ za pomocą metody Rungego-Kutty.

3. Układ mechaniczny o jednym stopniu swobody



Rysunek 6. Układ mechaniczny o jednym stopniu swobody

Zostanie rozwiązany dwoma sposobami układ składający się z masy, sprężystości i tłumienia, opisany równaniem:

$$m\ddot{x} + b\dot{x} + kx = F(t) \quad (3)$$

dla wymuszenia $F = A \cdot \sin(2\pi \cdot f \cdot t)$, czasu $t \in \langle 0, 30 \rangle$, warunków początkowych $x = 0m, \dot{x} = 10\frac{m}{s}$ i parametrów w tabeli 1.

Tabela 1. Parametry symulacji

Parametr	Wartość	Jednostka
masa m	330	kg
sprężystość k	20	$\frac{N}{m}$
tłumienie b	$2 \cdot 0.4\sqrt{k \cdot m}$	$\frac{N \cdot s}{m}$
częstotliwość f	73	Hz
amplituda A	45	kN

3.1. Tryb tekstowy

Pierwszym sposobem rozwiązania tego układu będzie wykorzystanie interfejsu tekstowego. Zmiennymi modelu będą $x, \dot{x} = \frac{dx}{dt}$ i wymuszenie, stałymi - m, b, k, f, A . Model opisują trzy równania – równanie ruchu, równanie wymuszenia i podstawienie $dx = \dot{x} = \frac{dx}{dt}$.

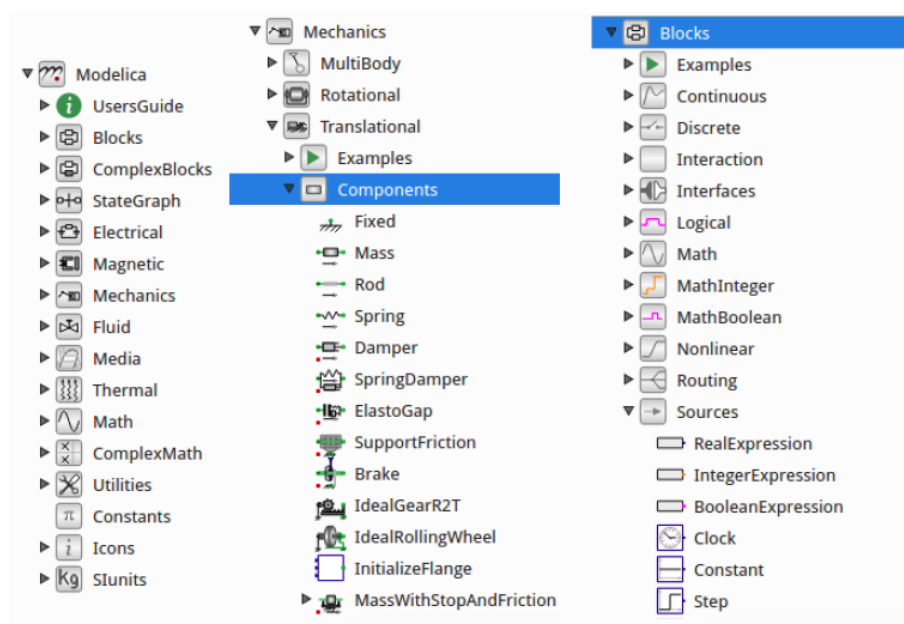
```

model massdamp
  parameter Real m=330, k=20, b=2*0.4*sqrt(k*m), f=73, A=45;
  Real x(start=0), dx(start=10);
  Real F;
  equation
    F=A*sin(2*3.14*f*time);
    m*der(dx)+b*dx+k*x=F;
    dx=der(x);
end massdamp;
    
```

3.2. Tryb graficzny (modele blokowe)

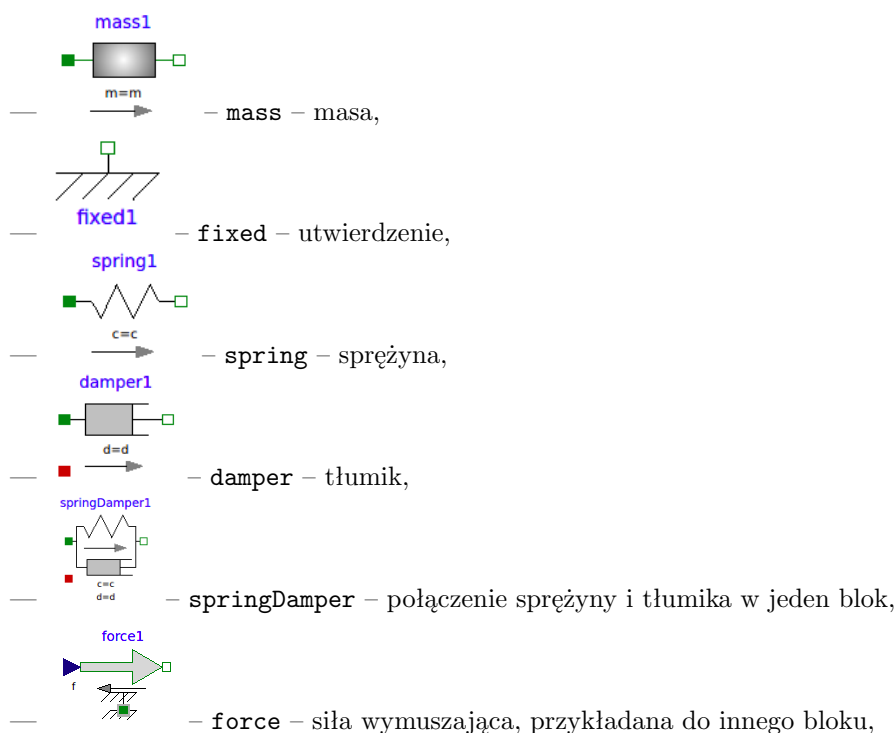
Głównym sposobem modelowania w OpenModelice jest wykorzystanie schematów blokowych, takich jak na rys. 6 i odwzorowanie ich w programie.

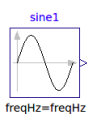
Polega to na wyborze odpowiednich elementów z biblioteki obiektów i modeli i przeniesienie ich do do widoku modelu. Modelica pozwala na modelowanie różnych obiektów rzeczywistych, mechanicznych, elektrycznych, mechatronicznych, etc. Przy badanym układzie interesują nas jedynie obiekty mechaniczne. Wszystkie podstawowe modele znajdują się w zakładce *Modelica* (rys. 7).



Rysunek 7. Podstawowe modele w programie Modelica, modele mechaniczne, wymuszenia

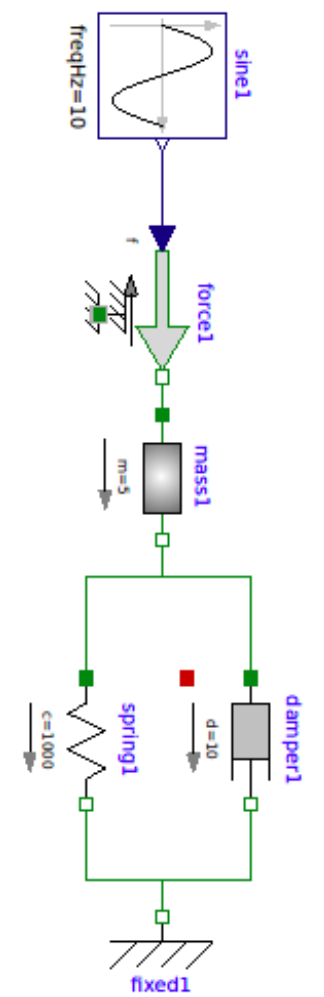
Do zbudowania modelu masy na sprężynie będą nam potrzebne następujące elementy, można je znaleźć w zakładkach `Modelica` `>` `Mechanics` `>` `Translational` i `Blocks` `>` `Sources`:





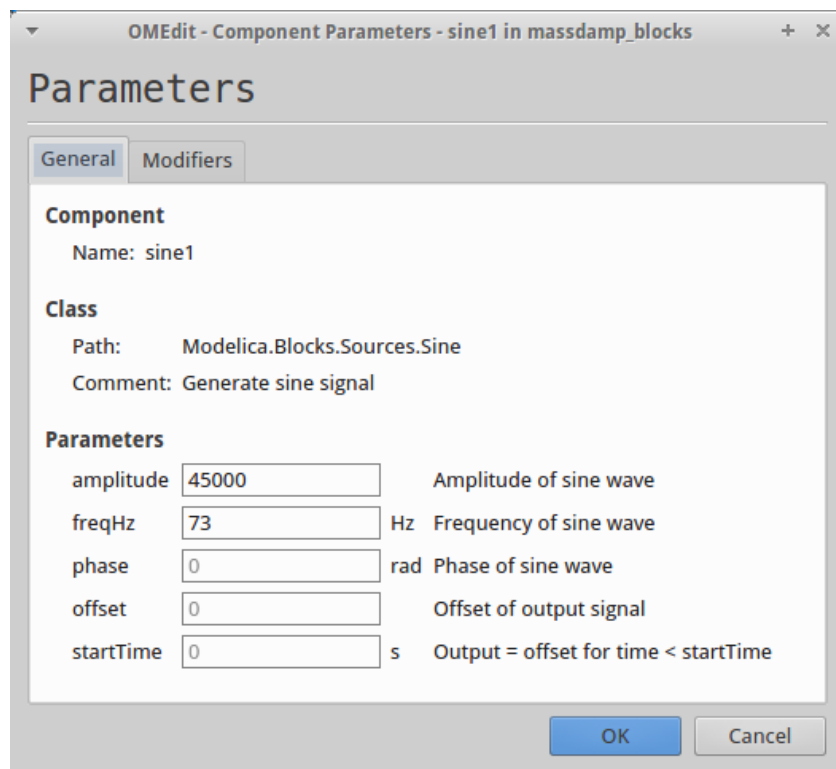
— **sine** – sygnał wymuszający.

Po odnalezieniu ich w bibliotece należy je przeciągnąć do głównego okna i połączyć ze sobą. Bloki można obrócić za pomocą skrótu klawiszowego **ctrl**+**R**. Finalnie, zbudowany układ powinien wyglądać jak na rys. 8.

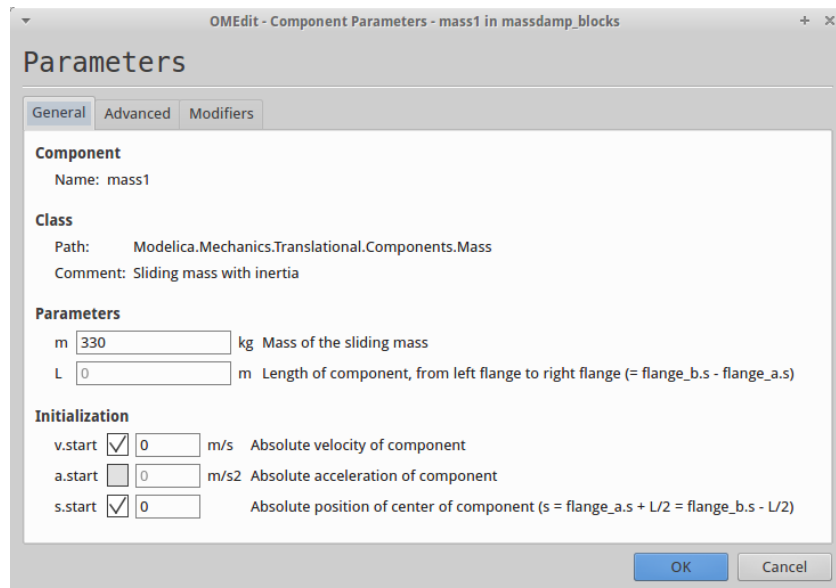


Rysunek 8. Gotowy model

Każdemu z elementów należy nadać odpowiednie parametry – masę, sprężystość, etc. Należy kliknąć dwukrotnie na danym elemencie i podać odpowiednie wartości, przykładowo, na rys. 9 pokazano, jak nadać odpowiednią amplitudę i częstotliwość sile wymuszającej. Dodatkowo, należy nadać masie odpowiednie warunki początkowe (rys. 10).



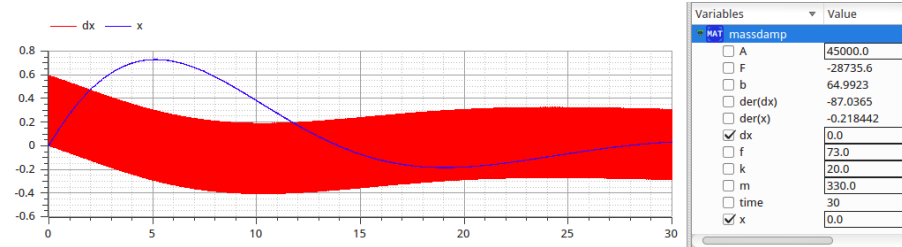
Rysunek 9. Parametry wymuszenia sinusoidalnego



Rysunek 10. Parametry i warunki początkowe masy x

3.3. Wyniki symulacji

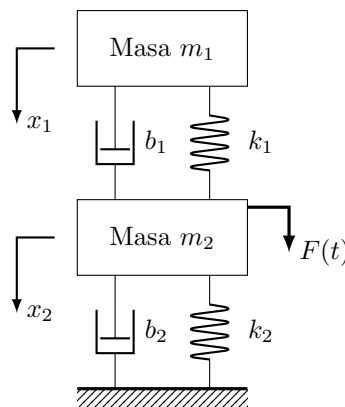
Dwa modele, jeden zbudowany za pomocą równań i drugi, blokowy, powinny dać te same wyniki, przedstawione na rys. 11. Powinny one również zgadzać się z wynikami uzyskanymi w programie MATLAB.



Rysunek 11. Wyniki symulacji masy na sprężynie i tłumiku

3.4. Ćwiczenie

Zbudować model układu przedstawionego na rys. 12 warunków początkowych $x_1 = x_2 = 0m, \dot{x}_1 = \dot{x}_2 = 0 \frac{m}{s}$ i dla parametrów przedstawionych w tabeli 2.



Rysunek 12. Układ mechaniczny o dwóch stopniach swobody

Tabela 2. Parametry symulacji

Parametr	Wartość	Jednostka
masa m_1	200	kg
sprężystość k_1	$(1 \div 20)$	$\frac{N}{m}$
tłumienie b_1	$0.2 \cdot (0.1 \div 1) \sqrt{k_1 \cdot m_1}$	$\frac{N \cdot s}{m}$
masa m_2	130	kg
sprężystość k_2	20	$\frac{N}{m}$
tłumienie b_2	$0.2 \cdot 0.4 \sqrt{k_2 \cdot m_2}$	$\frac{N \cdot s}{m}$
częstotliwość f	10	Hz
amplituda A	100	N