

# SIMULINK 5

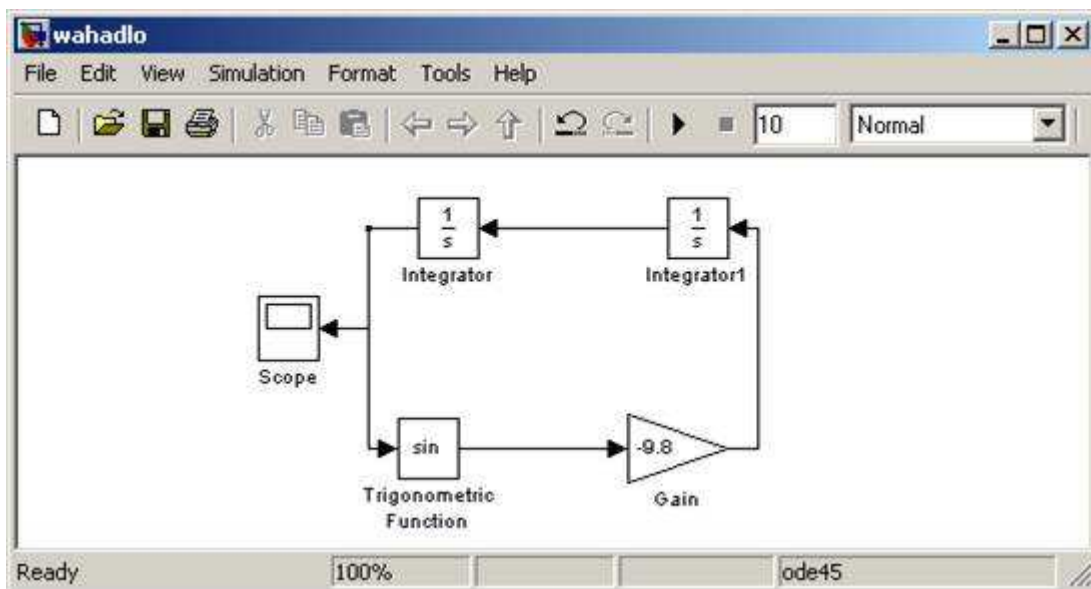
---

## Zawartość

Wahadło – przypomnienie .....	1
Używanie bloku <i>MATLAB Function</i> .....	1
Efekt końcowy .....	3

## Wahadło – przypomnienie

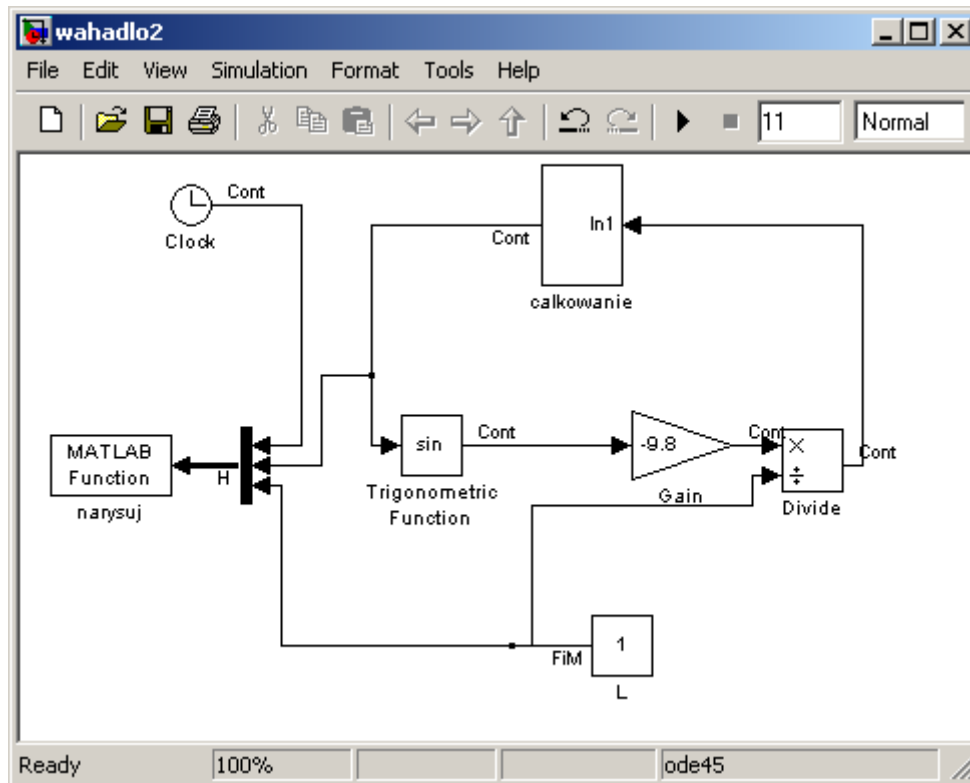
W pierwszej części kursu jednym z pierwszych modeli było [wahadło matematyczne](#), gotowy schemat wyglądał tak:



Narzędzie, którego użyliśmy do wizualizacji wyniku, to blok *Scope*. Oczywiście taki sposób prezentacji danych jest wystarczająco jasny, ale często modelując układ mechaniczny chcielibyśmy zobaczyć końcowy efekt w postaci animacji. W tym celu stworzymy własny blok, który na podstawie przekazywanych wyników symulacji online będzie rysował ruszające się wahadło.

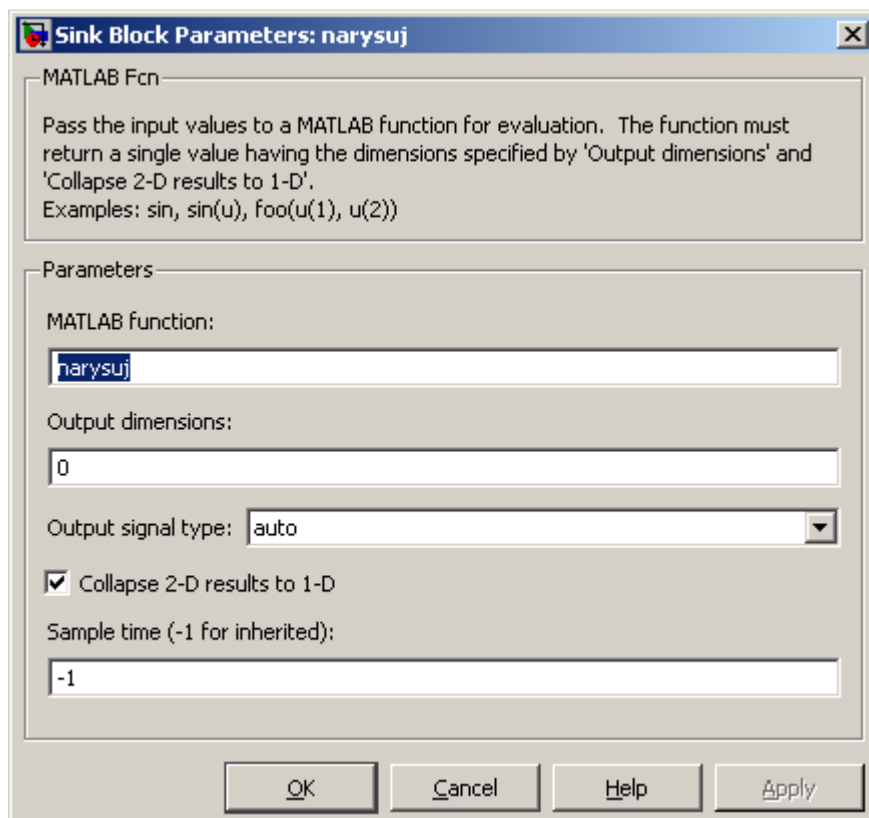
## Używanie bloku *MATLAB Function*

Do rozwiązania naszego problemu posłużą nam blok *MATLAB Function*. Oprócz tego rozbudujemy trochę nasz model poprzez dodanie kilku dodatkowych elementów. Jednym z nich będzie długość nici  $L$  podana *explicite*. Do tej pory wpisywaliśmy tę wartość na stałe do bloku *Gain*, teraz będzie nam również potrzebna jako parametr wejściowy do naszej funkcji. Do modelu dodamy również aktualny czas symulacji, aby móc go wyświetlać podczas trwania animacji. Oprócz tytułowego bloku *MATLAB Function*, potrzebny jest nam jeszcze blok *Divide*, żeby podzielić nasze równanie w odpowiednim miejscu przez  $L$  oraz blok *Mux*, ponieważ nasza funkcja będzie miała jeden argument wejściowy w postaci trzejelementowego wektora. Nasz chce mat powinien wyglądać jak poniżej:



Podsystem *calkowanie* składa się z podwójnego całkowania.

Domyślnie blok *MATLAB Function* ma jedno wejście i jedno wyjście. Nasza funkcja tylko rysuje i nie zwraca żadnych argumentów, dlatego też w preferencjach powinniśmy ustawić 0 jako wymiar argumentu wyjściowego:



A miejscu nazwy funkcji wpisujemy nazwę m-pliku, który będzie realizował to zadanie. W naszym przypadku jest to stworzona funkcja *narysuj*. Obowiązuje tu oczywiście taka sama zasada jak w matlabie dotycząca widoczności funkcji, tzn. plik z funkcją powinien znajdować się w tym samym katalogu co model lub też w jednej z lokalizacji dodanych wcześniej do pliku *pathdef.m*.

Funkcja rysująca wygląda następująco:

```
function narysuj(u)
% NARYSUJ - funkcja rysująca ruch wahadła matematycznego
% jej argumentem wejściowym jest trzelementowy wektor u zawierający
% aktualny czas symulacji, wartość kąta odpowiadającą temu czasowi, oraz
% długość nici wahadła
% Autor: Grzegorz Knor

t = u(1);
teta = u(2);
l=u(3);

y=-l*cos(teta);
x=l*sin(teta);

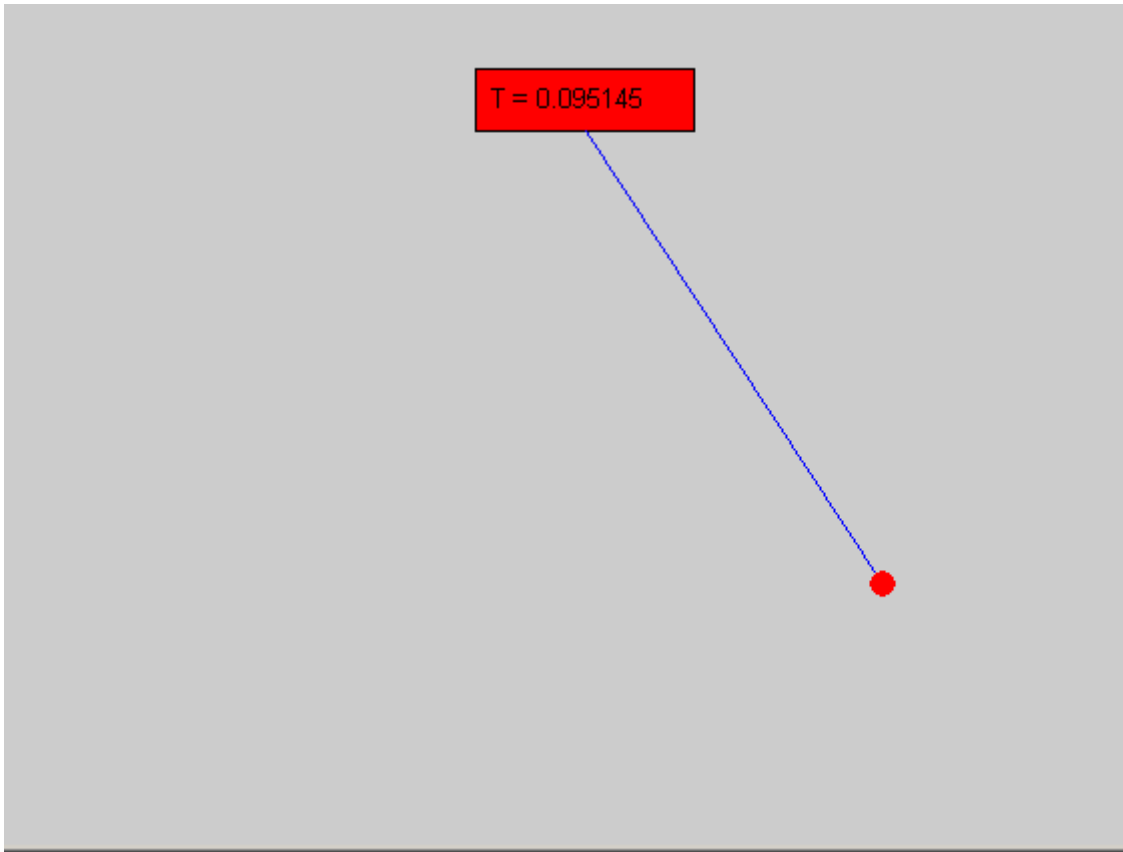
cla
fill([-1/4 -1/4 1/4 1/4],[0.1 0 0 0.1], 'r')
hold on
line([0 x],[0 y])
plot(x,y, 'r.', 'MarkerSize', 30)
xlim([-1 1])
ylim([-1 0.1])
text(-1/4,0.05,[' T = ' num2str(t)])
hold off
axis off

drawnow
```

Animuje ona ruch wahadła i wyświetla aktualny czas symulacji.

## Efekt końcowy

Efekt końcowym jest wykres przedstawiający ruch wahadła tak jak na poniższym rysunku:



---

2009 – Grzegorz Knor