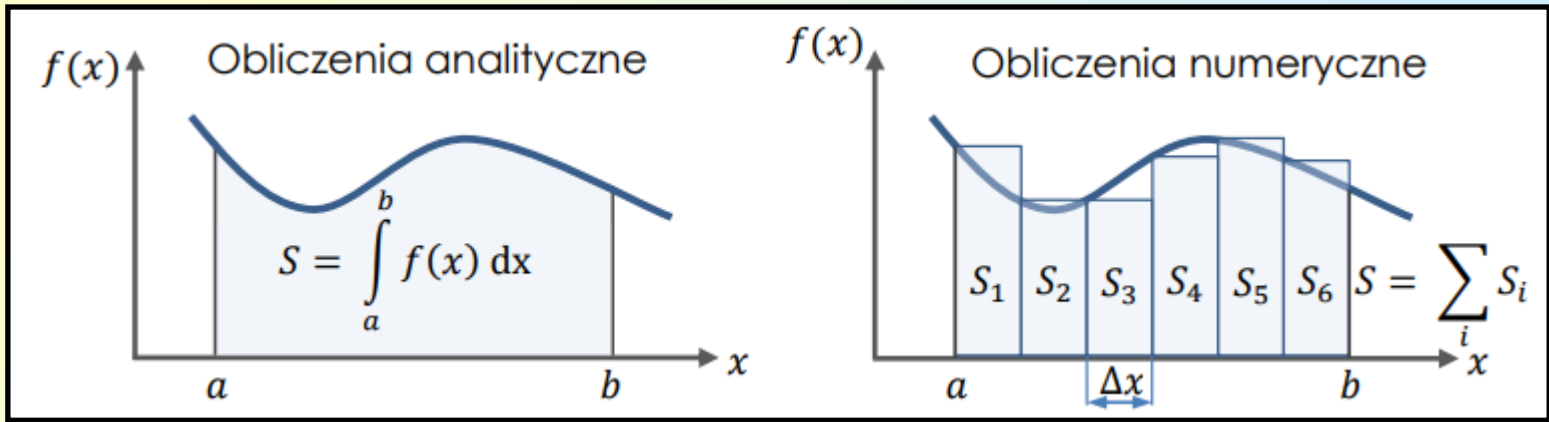


# **Modelowanie i symulacja komputerowa**

**Środowisko SciLab  
Pakiet Xcos**

**Tworzenie graficznych modeli  
procesów dynamicznych**

# Obliczanie wartości całek oznaczonych



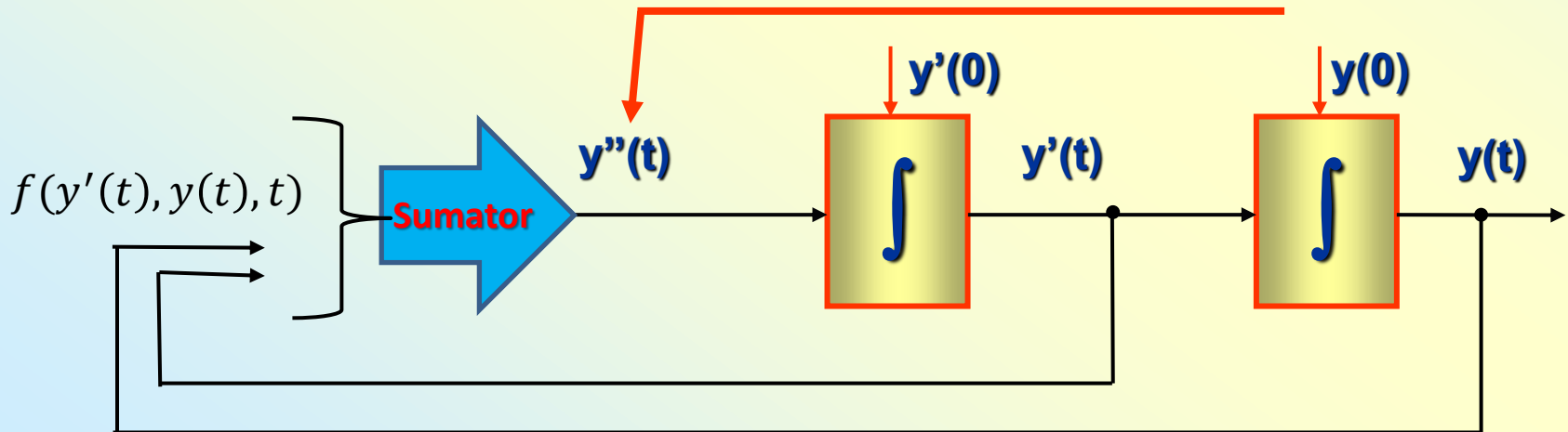
$$F(x) = \int f(x) dx$$

$$\int_a^b f(x) dx = F(b) - F(a)$$

$$\int_a^b f(x) dx = \sum_i S_i$$

## Rozwiązywanie numeryczne równań różniczkowych (języki symulacyjne)

$$y''(t) = f(y'(t), y(t), t)$$



# Środowisko *Scilab* – pakiet *Xcos*

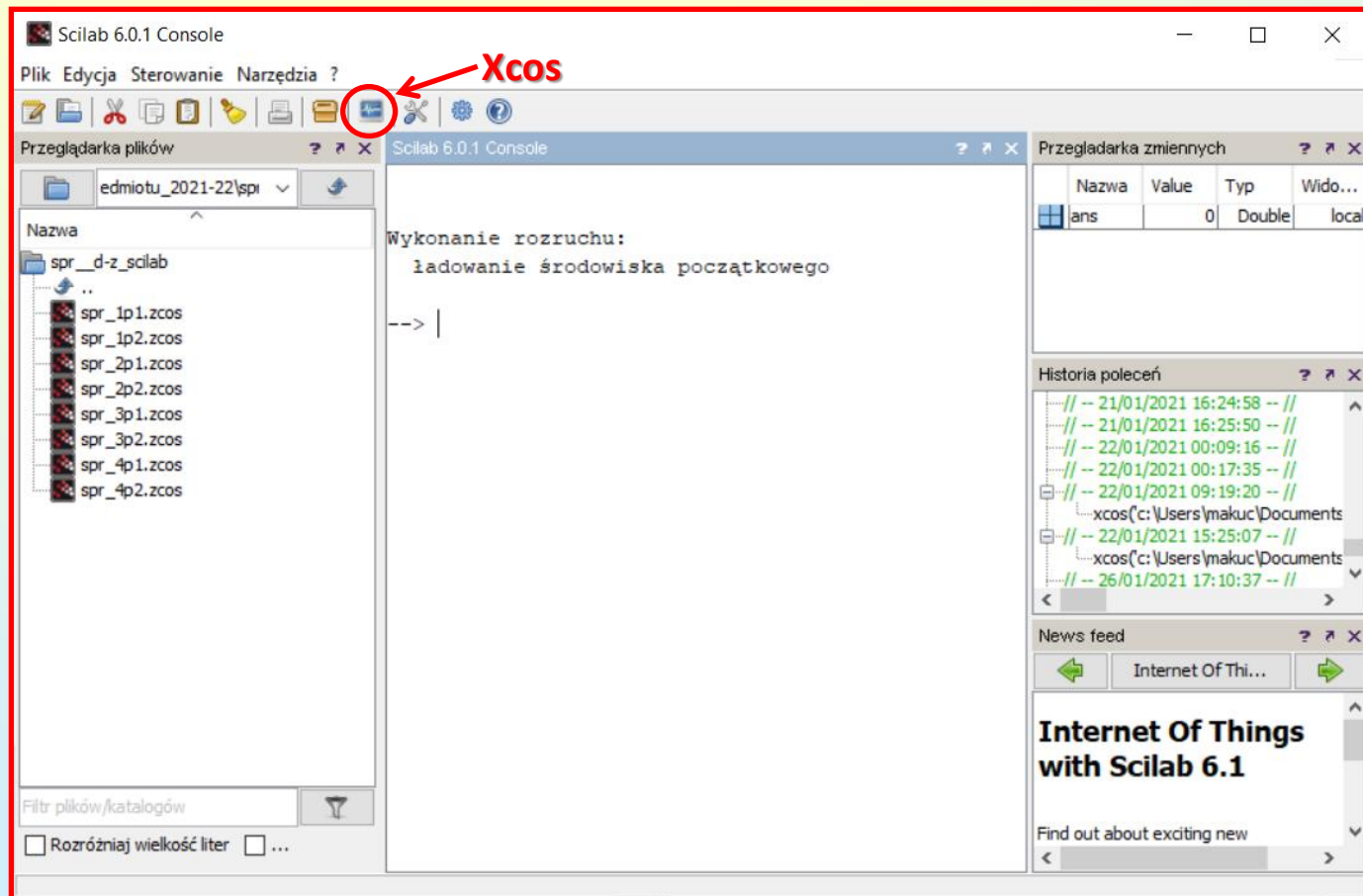
Scilab jest darmowym środowiskiem do obliczeń naukowych rozprowadzonym na licencji open source.

Na stronie projektu [www.scilab.org](http://www.scilab.org) można znaleźć wersje instalacyjne pakietu, dokumentację oraz wiele opracowań pomocnych podczas nauki jego użytkowania.

**Scilab** jest pewnego rodzaju darmowym zamiennikiem komercyjnego pakietu **MatLab**.

Dokumentacja na stronie projektu **Scilab**: [www.scilab.org/resources/documentation](http://www.scilab.org/resources/documentation).

Po uruchomieniu środowiska pojawia się konsola postaci:



W środkowym oknie można wpisywać bezpośrednio instrukcje w języku Scilab (podobnie jak w języku MatLab). Wyniki otrzymujemy natychmiast ponieważ jest to interpreter.

```
scilab
1.zcos
2.zcos
1.zcos
2.zcos
1.zcos
2.zcos
1.zcos
2.zcos

--> A=sin(%pi/2)
A =

    1.

--> A=[1 2 3]; B=[1; 2; 3]
B =

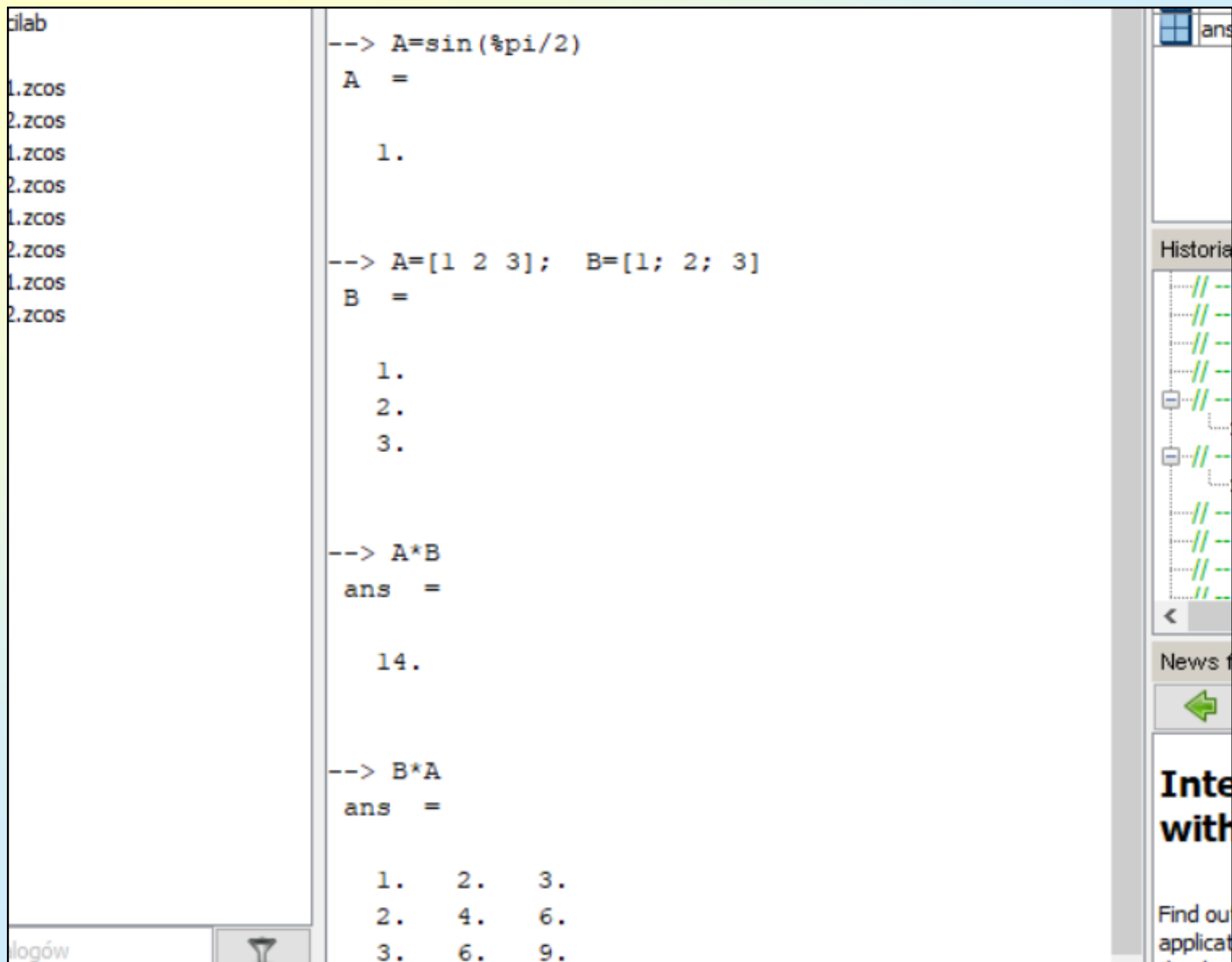
    1.
    2.
    3.

--> A*B
ans =

    14.

--> B*A
ans =

    1.    2.    3.
    2.    4.    6.
    3.    6.    9.
```

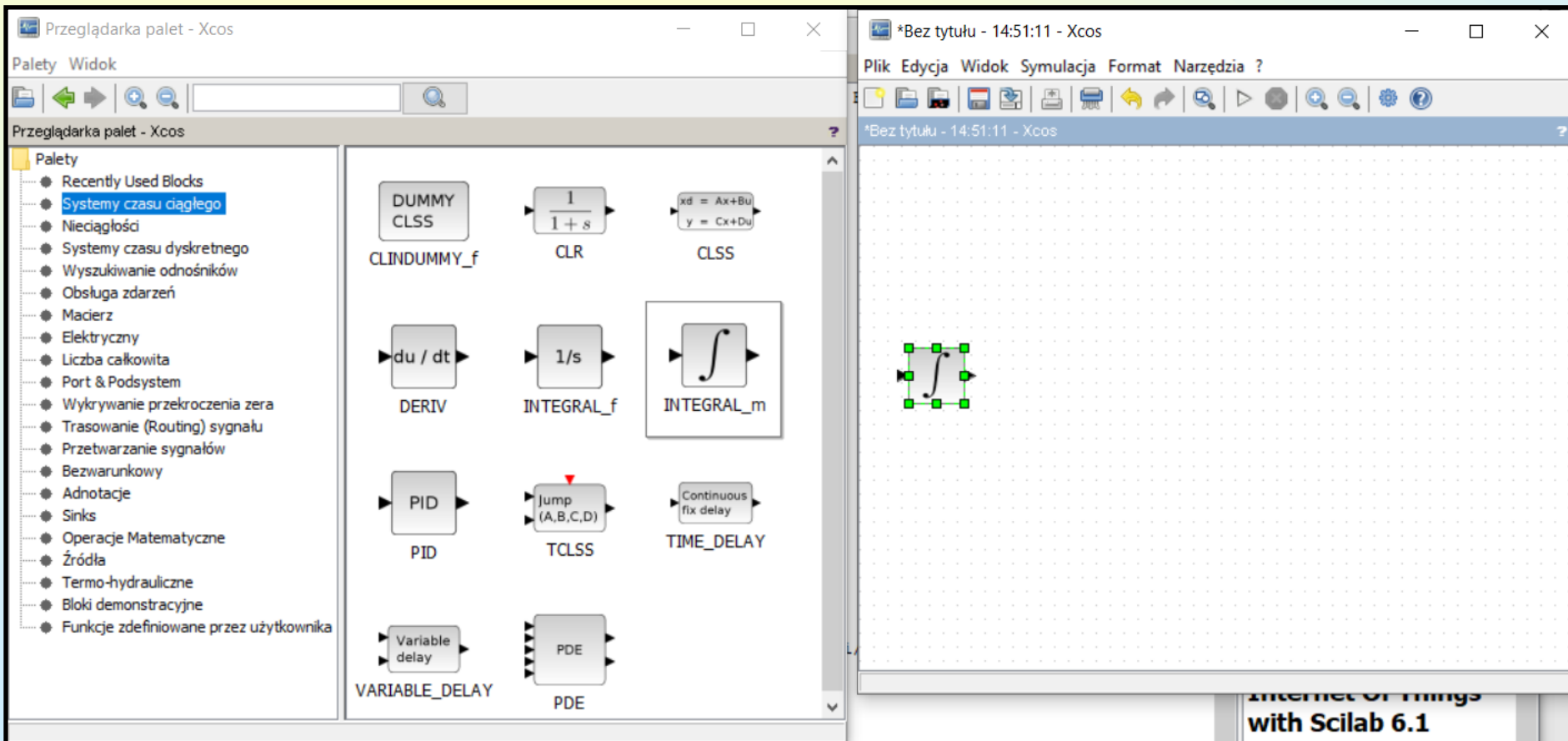


## Przykład podstawowych poleceń

Polecenie					Znaczenie polecenia				
<code>x+y</code>	<code>x-y</code>	<code>x*y</code>	<code>x/y</code>	<code>x^y</code>	<code>x+y</code>	<code>x-y</code>	<code>x*y</code>	<code>x/y</code>	<code>x^y</code>
<code>sqrt(5)</code>		<code>log(5)</code>	<code>log10(5)</code>		$\sqrt{5}$		<code>ln(5)</code>	<code>log(5)</code>	
<code>sin(x)</code>		<code>sind(x)</code>			<code>sin(x[rad])</code>		<code>sin(x[°])</code>		
<code>%pi</code>		<code>%e</code>			$\pi = 3.141592653$		$e = 2.71828182$		
<code>X=156.98*%pi/sind(45)</code>					$X=156.98*\pi/\sin(45^\circ)$				
<code>A=[1 2 3] def. wektora (wiersz)</code>					$A=[1 \ 2 \ 3]$				
<code>B=[1; 2; 3] def. wektora (kolumna)</code>					$B=[1; 2; 3]$				
<code>A*B</code>					<pre>--&gt; A*B ans =      14. (wynik to skalar)</pre>				
<code>B*A</code>					<pre>--&gt; B*A ans =       1.     2.     3.      2.     4.     6.      3.     6.     9. (wynik to macierz)</pre>				

# Xcos


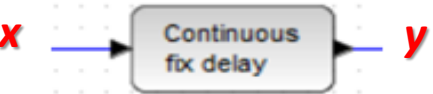
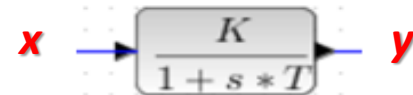
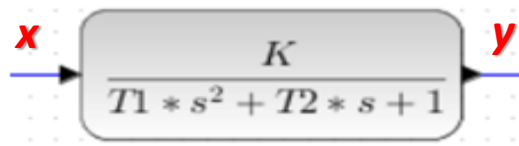

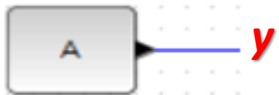

Moduł Xcos uruchamia się wpisując w konsoli polecenie "xcos" lub klikając dziewiątą ikonę na pasku skrótów. Podobnie jak dla MatLab model symulacyjny ma postać schematu blokowego, który budujemy wykorzystując gotowe bloki, są one dostępne w przeglądarce palet. Bloki z Przeglądarki "przeciągamy" (używając lewego klawisza myszki) na obszar okna projektu, gdzie później możemy je łączyć między sobą tworząc schemat blokowy czyli model symulacyjny układu.

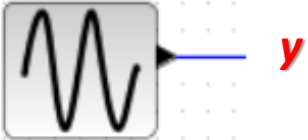
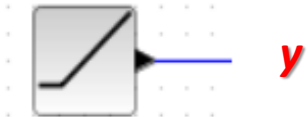



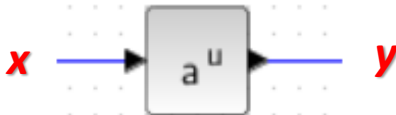


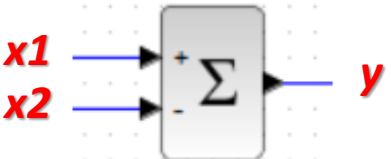
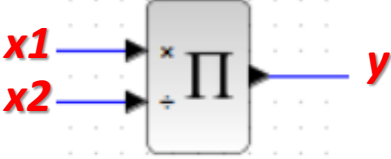
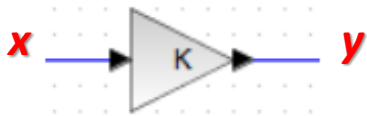



Przeglądarka palet

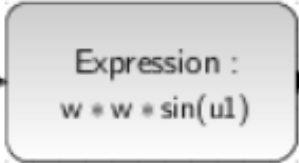
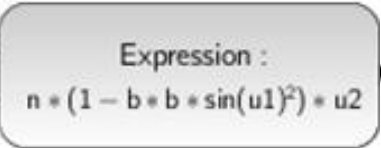

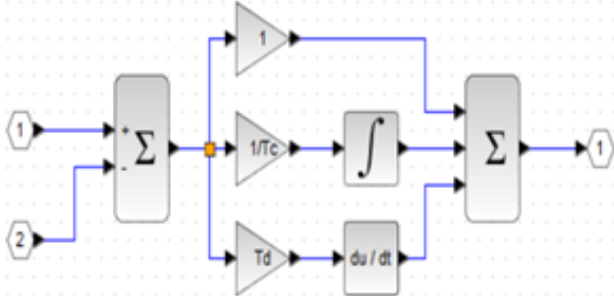


Okno projektu

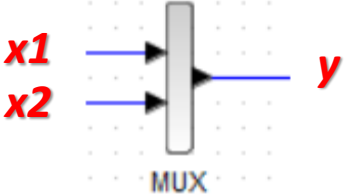
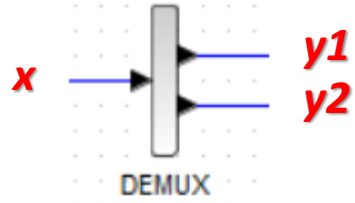
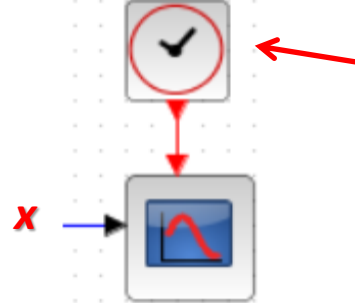
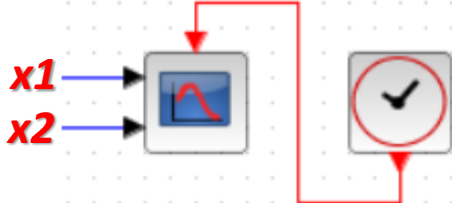
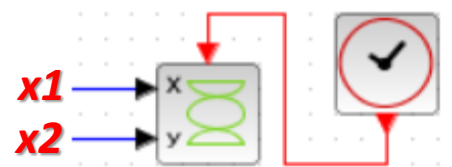
## Ikony wybranych bloków i opis realizowanych operacji

Blok	Operacja	Paleta
	$y = \int_0^t x dt + y(0)$	Systemy czasu ciągłego
	$y = x(t - T_0)$	Systemy czasu ciągłego
	<p>człon inercyjny pierwszego rzędu</p> $T \frac{dy}{dt} + y = K * x$	Systemy czasu ciągłego
	<p>człon inercyjny drugiego rzędu</p> $T1 \frac{d^2y}{dt^2} + T2 \frac{dy}{dt} + y = K * x$	Systemy czasu ciągłego
	$y = \frac{dx}{dt}$	Systemy czasu ciągłego
	$y = A$	Źródła
	<p><b>skok jednostkowy</b></p> $y = \mathbf{1}(t) = \begin{cases} 0 & \text{dla } t < T_0 \\ 1 & \text{dla } t \geq T_0 \end{cases}$	Źródła

	<p>Generator sygnału  <math>y = A * \sin(\omega t + \varphi)</math></p>	<p>Źródła</p>
	<p><math>y = k * t</math></p>	<p>Źródła</p>
	<p><math>y = t</math></p>	<p>Źródła</p>
	<p><math>y = \sin(x[\text{rad}])</math></p>	<p>Operacje matematyczne</p>
	<p><math>y = x^a</math></p>	<p>Operacje matematyczne</p>
	<p><math>y = a^x</math></p>	<p>Operacje matematyczne</p>

	$y = x_1 - x_2$ <p>liczba wejść i ich znaki do ustalenia</p>	<p>Operacje matematyczne</p>
	$y = x_1 / x_2$ <p>liczba wejść i ich znaki do ustalenia</p>	<p>Operacje matematyczne</p>
	$y = k * x$	<p>Operacje matematyczne</p>
	<p>element z nasyceniem</p>	<p>Nieciągłości</p>
	<p>histereza</p>	<p>Nieciągłości</p>
	<p>element ze strefą martwą</p>	<p>Nieciągłości</p>

<p><math>u1</math></p>  <p><math>y = w * w * \sin(u1)</math></p>	<p>number of inputs <input type="text" value="1"/></p> <p>scilab expression <input type="text" value="w*w*sin(u1)"/></p>	<p>Funkcje zdefiniowane przez użytkownika</p>
<p><math>u1</math> <math>u2</math></p>  <p><math>y = n * (1 - b * b * \sin(u1)^2) * u2</math></p>	<p>number of inputs <input type="text" value="2"/></p> <p>scilab expression <input type="text" value="n*(1-b*b*sin(u1)^2)*u2"/></p>	<p>Funkcje zdefiniowane przez użytkownika</p>
<p><math>we1</math> <math>we2</math></p>  <p><i>SUPER_f</i></p>		<p>Funkcje zdefiniowane przez użytkownika</p>
<p><math>x</math></p>  <p><math>y</math></p>	<p>Set PID parameters</p> <p>Proportional <input type="text" value="K"/></p> <p>Integral <input type="text" value="Tc"/></p> <p>Derivation <input type="text" value="Td"/></p>	<p>Systemy czasu ciągłego</p>
<p><math>x</math></p>  <p><math>y</math></p>	<p>Set Interpolation block parameters</p> <p>X coord. <input type="text" value="[-2 : 0.5 : 1]"/></p> <p>Y coord. <input type="text" value="[10, 14, 16, 17, 20, 21, 25]"/></p>	<p>Wyszukiwanie odnośników</p>

	<p>Multiplexer  <u>łączenie</u> sygnałów  w wektor</p>	<p>Trasowanie  (Routing) sygnału</p>
	<p>Demultiplexer  rozdzielanie  wektora sygnałów</p>	<p>Trasowanie  (Routing) sygnału</p>
	<p>blok sterujący procesem  obliczeń. Określa krok  całkowania i czas  inicjalizacji wykresu  <math>x(t)</math>.  Monitor z wykresem <math>x(t)</math></p>	<p>Źródła i Sinks</p>
	<p>Monitor z wykresami  <math>x1(t)</math> i <math>x2(t)</math>  Można ustawić  więcej sygnałów  wejściowych.</p>	<p>Źródła i Sinks</p>
	<p>Monitor graficzny  uzyskujemy wykres <math>y(x)</math>  czyli <math>x2(x1)</math></p>	<p>Źródła i Sinks</p>

## Budowanie symulacyjnego schematu blokowego na podstawie równania różniczkowego zwyczajnego

Niech równanie różniczkowe ma postać:

$$T1 \frac{d^2 y}{dt^2} + T2 \frac{dy}{dt} + y = Kx(t) \quad \text{warunki początkowe: } y'(0) = a, \quad y(0) = b$$

Po prostych przekształceniach otrzymujemy:

$$\frac{d^2 y}{dt^2} = -\frac{T2}{T1} \frac{dy}{dt} - \frac{1}{T1} y + \frac{K}{T1} x(t) \quad \text{z warunki początkowymi}$$

Pamiętamy, że "blok całkowania" ma właściwość całkowania sygnału wejściowego. Oczywiście musimy także zadbać o zadanie warunku początkowego.

Możemy zauważyć, że tworząc łańcuch bloków całkujących dla równania różniczkowego n-tego rzędu otrzymamy ciąg postaci jak niżej, gdzie na wyjściu ostatniego bloku całkującego będzie funkcja  $y(t)$  będąca rozwiązaniem równania różniczkowego.

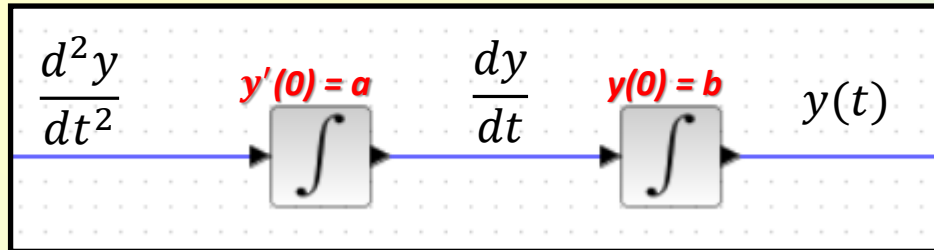


# Budowanie symulacyjnego schematu blokowego na podstawie równania różniczkowego

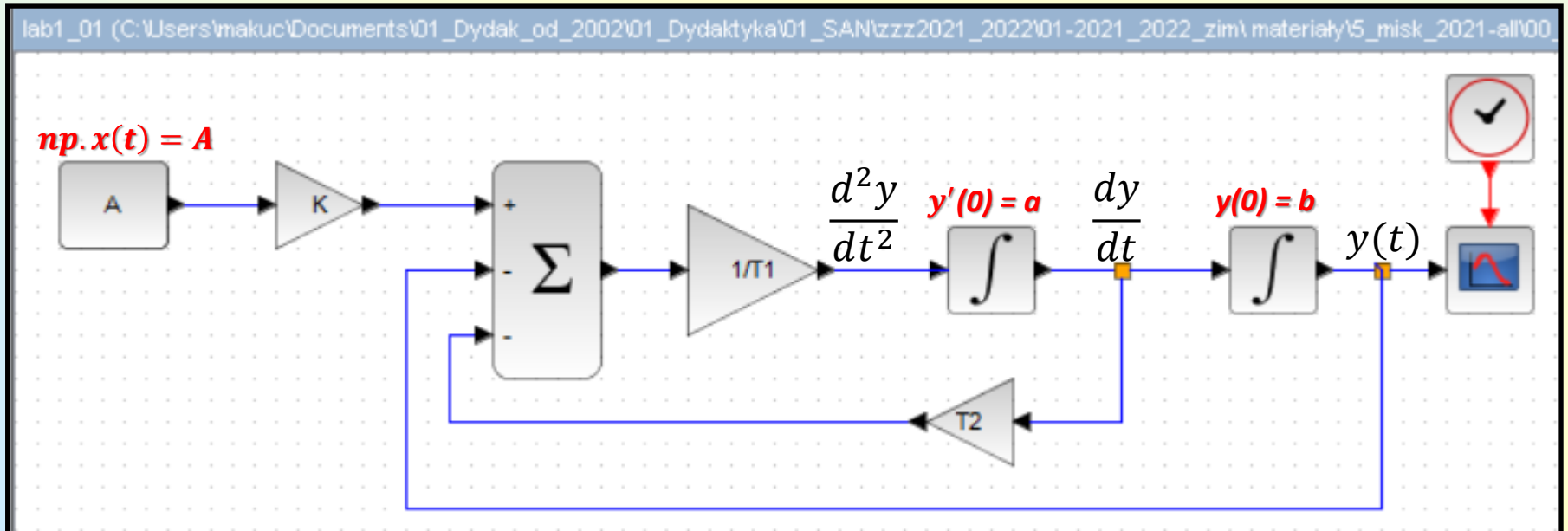
Dla równania, po wyciągnięciu współczynnika  $1/T_1$  przed nawias

$$\frac{d^2y}{dt^2} = \frac{1}{T_1} \left( -T_2 \frac{dy}{dt} - y + Kx(t) \right) \quad \text{warunki początkowe: } y'(0) = a, \quad y(0) = b$$

konstruujemy ciąg bloków całkujących postaci:



Korzystając z dostępnych sygnałów  $dy/dt$  oraz  $y(t)$  oraz innych bloków funkcyjnych budujemy blokowy schemat symulacyjny równania różniczkowego.



# Ustawianie istotnych parametrów – wartości parametrów, czas obliczeń, znaki sumatora

Set sum block parameters

Datatype (1=real double 2=complex 3=int32 ...)

Number of inputs or sign vector (of +1, -1)

Set Constant Block

Constant

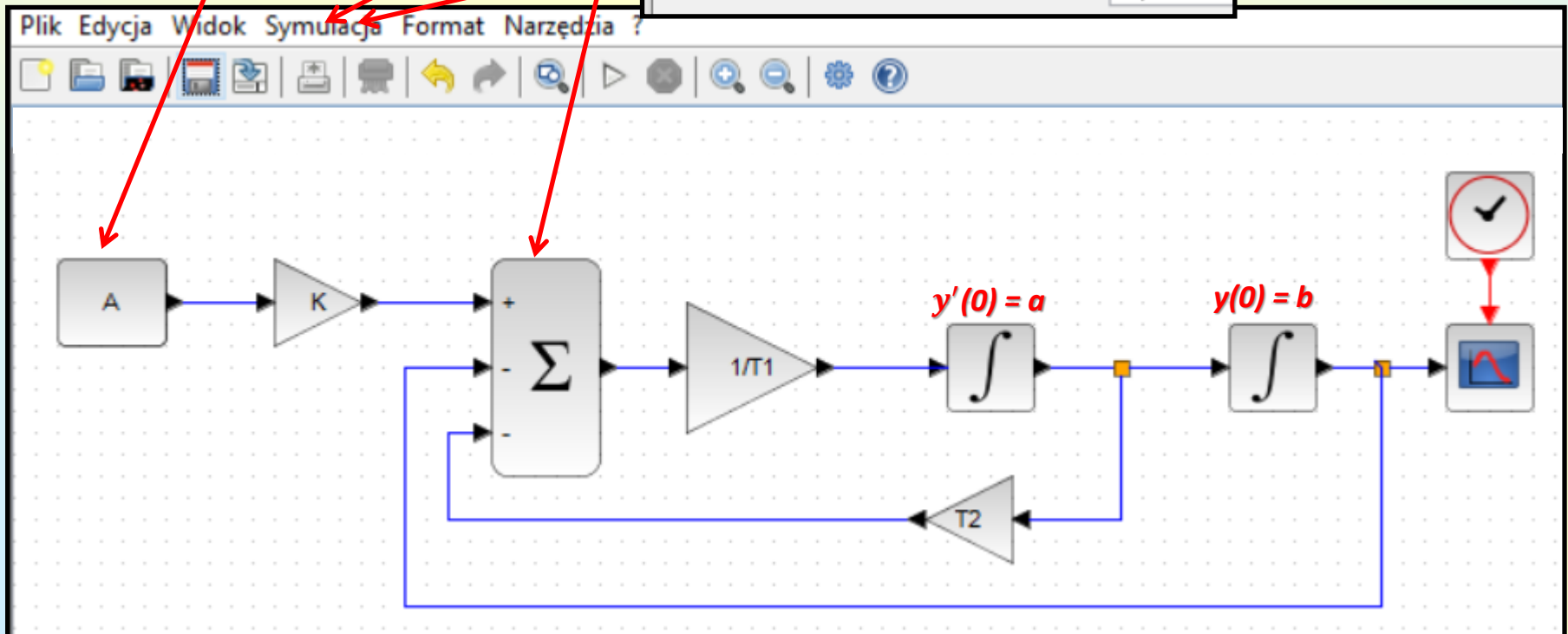
Ustaw kontekst

You may enter here scilab instructions to define definitions using Scilab instructions. These instructions are evaluated once confirmed (after the diagram is loaded).

```
lambda=1; T1=2; T2=3; K=2; a=0; b=0.2
```

Ustaw parametry

Ostateczny czas integracji



# Ustawianie istotnych parametrów – war. początkowe, okres obliczeń, ustaw. monitora

Set Scope parameters

Color (>0) or mark (<0) vector (8 entries)	5 1 3 7 9 11 13 15
Output window number (-1 for automatic)	-1
Output window position	[ ]
Output window sizes	[600;400]
Ymin	-15
Ymax	15
Refresh period	10
Buffer size	20
Accept herited events 0/1	0
Name of Scope (label&Id)	wykres Y(t)

Set Integral block parameters

Initial Condition

Set CLOCK\_c block parameters

Event clock generator

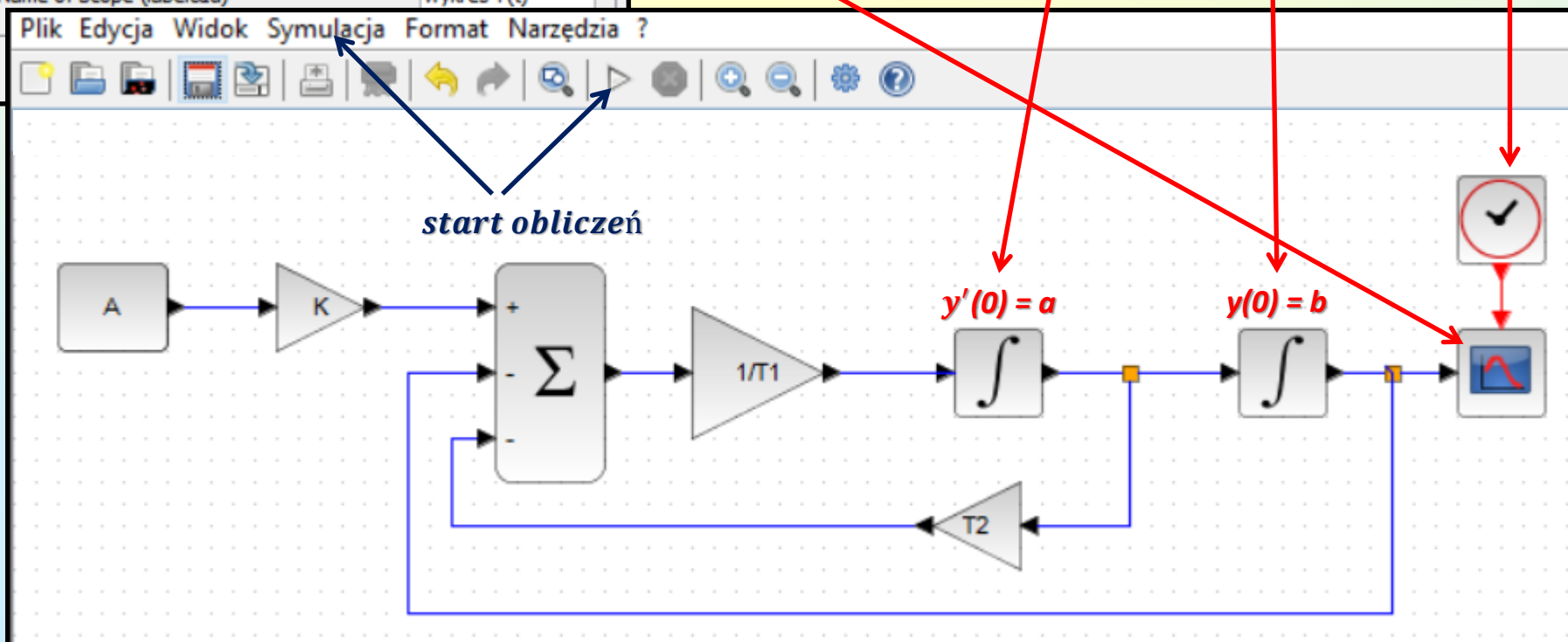
Do not start if 'Initialisation Time' is negative

Okres

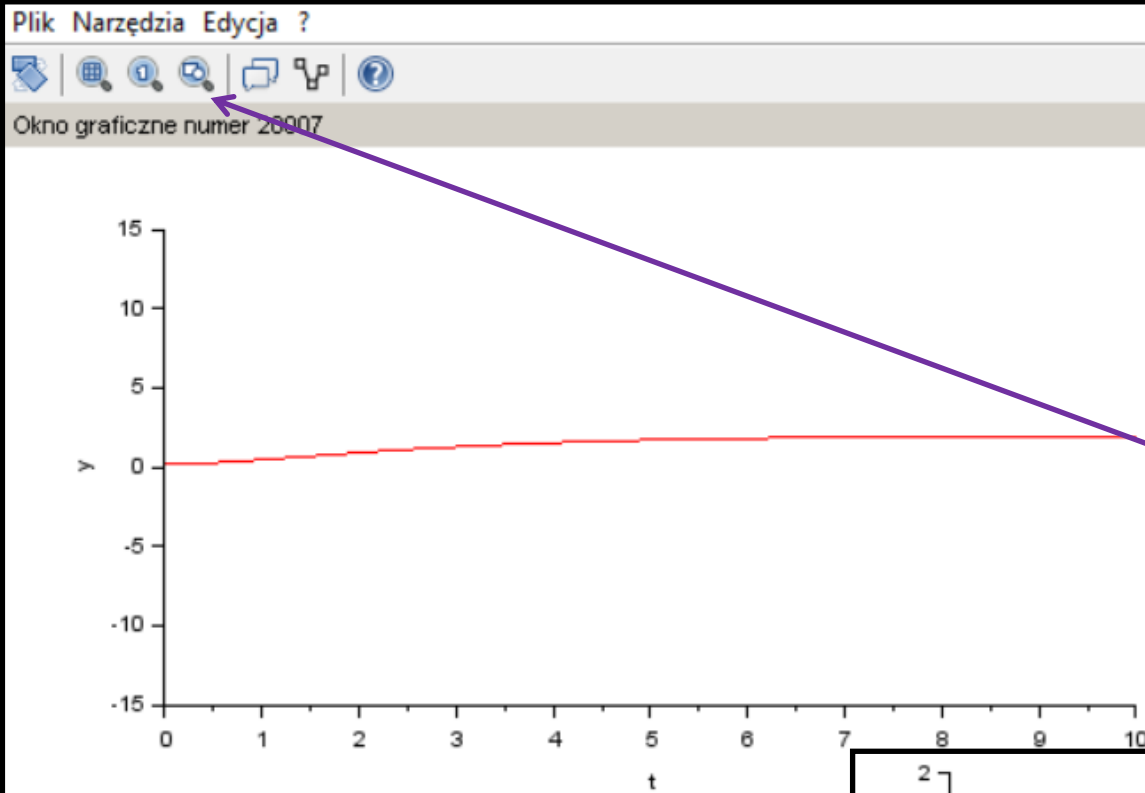
Initialisation Time

Set Integral block parameters

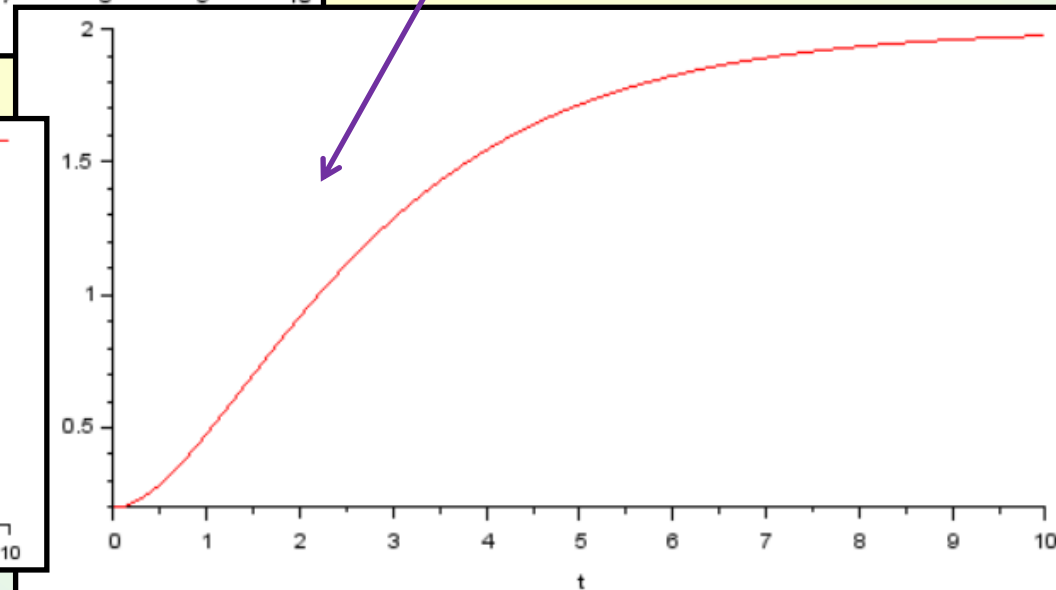
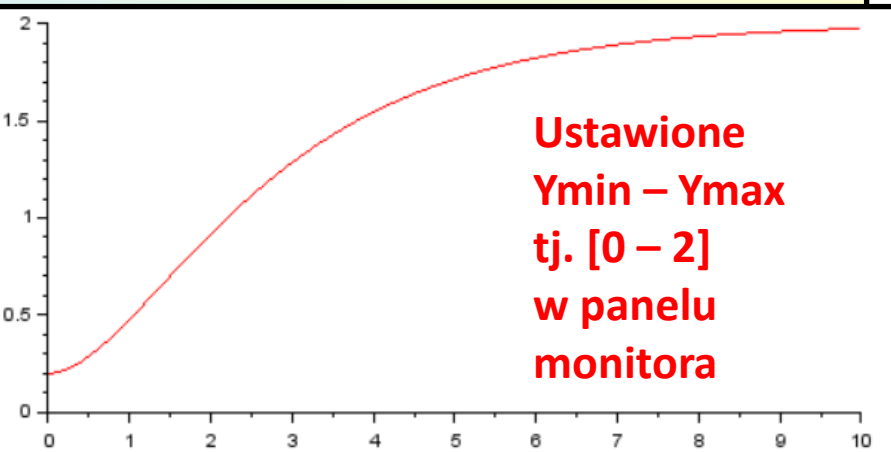
Initial Condition



## Uruchomienie symulacji (obliczeń)



Jeżeli wartości  $Y_{min}$  i  $Y_{max}$  dla okna graficznego będą źle ustawione to po wciśnięciu ikony **LUPKI** (czwarta od lewej na pasku) wykres zostanie dopasowany do rozmiarów okna.



# Silnik elektryczny obcowzbudny prądu stałego (DC)

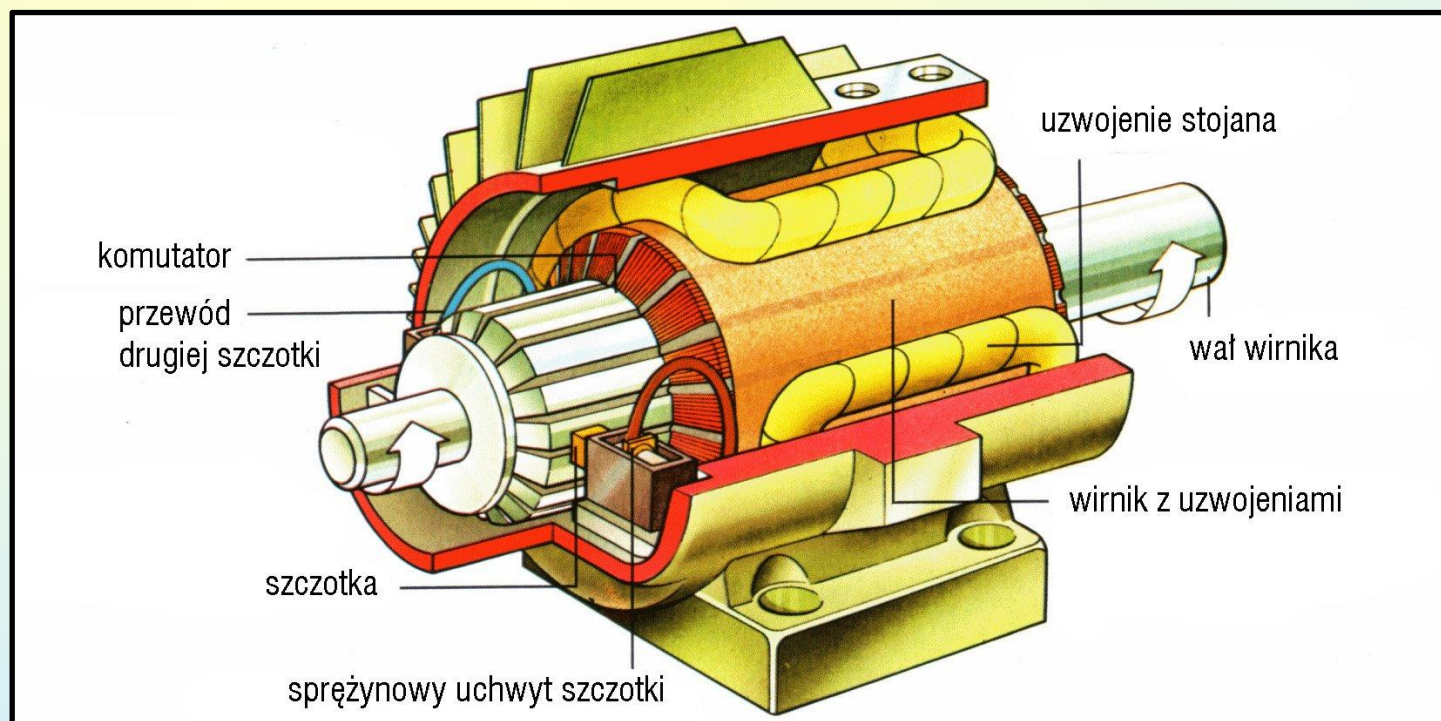
To maszyna, w której uzwojenie wzbudzenia na stojanie zasilane jest z oddzielnego, niezależnego źródła napięcia niż uzwojenie twornika (wirnika). Umożliwia to precyzyjną, niezależną regulację prędkości obrotowej i momentu, zapewniając stabilną pracę, często wykorzystywaną w zaawansowanych napędach.

## Kluczowe cechy i właściwości:

- **Niezależne zasilanie:** Uzwojenie wzbudzenia i twornika są zasilane z różnych źródeł, co odróżnia go od silników samowzbudnych (szeregowych lub bocznikowych).
- **Regulacja prędkości:** Prędkość obrotową reguluje się zazwyczaj poprzez zmianę napięcia twornika lub regulację prądu wzbudzenia.
- **Charakterystyka mechaniczna:** Silnik obcowzbudny posiada zazwyczaj sztywną charakterystykę mechaniczną, co oznacza, że prędkość obrotowa jest mało podatna na zmiany obciążenia.
- **Moment obrotowy:** Moment obrotowy jest wprost proporcjonalny do prądu twornika.
- **Zastosowanie:** Stosowany tam, gdzie wymagana jest szeroka i precyzyjna regulacja prędkości obrotowej, np. w systemach automatyki, obrabiarkach czy dźwigach.

# Silnik elektryczny obcowzbudny prądu stałego (DC)

## Rysunek poglądowy



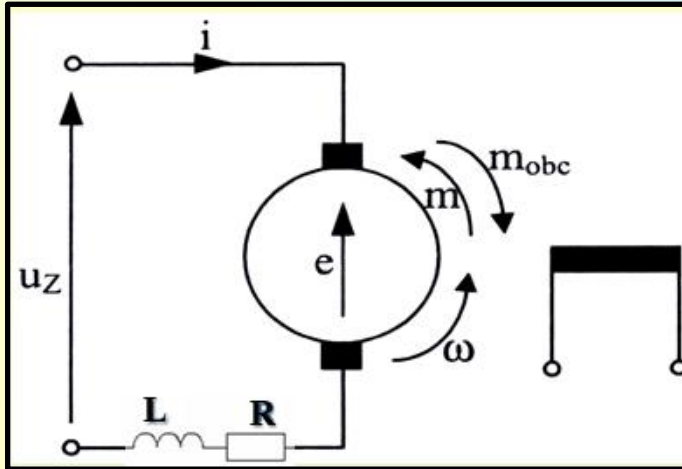
# Silnik obcowzbudny prądu stałego

Sposób działania obcowzbudnego silnika prądu stałego opisuje prosty układ liniowych równań różniczkowych i algebraicznych postaci:

$$\begin{cases} u_z = R * i(t) + L * \frac{di(t)}{dt} + e(t) \\ e(t) = k_e * \omega(t) \\ m(t) = m_{obc} + m_{dyn}(t) \\ m(t) = k_m * i(t) \\ m_{dyn}(t) = J * \frac{d\omega(t)}{dt} \end{cases}$$



$$\begin{cases} \frac{di}{dt} = \frac{1}{L} (u_z - k_e * \omega - R * i) \\ \frac{d\omega}{dt} = \frac{1}{J} (k_m * i - m_{obc}) \end{cases}$$



przykładowe wartości danych znamionowych:

$$J = 2.7 \text{ [kg*m}^2\text{]}$$

$$R = 0.465 \text{ [}\Omega\text{]}$$

$$L = 0.015345 \text{ [H]}$$

$$k_e = 2.62 \text{ [V*s]}$$

$$k_m = 2.62 \text{ [N*m/A]}$$

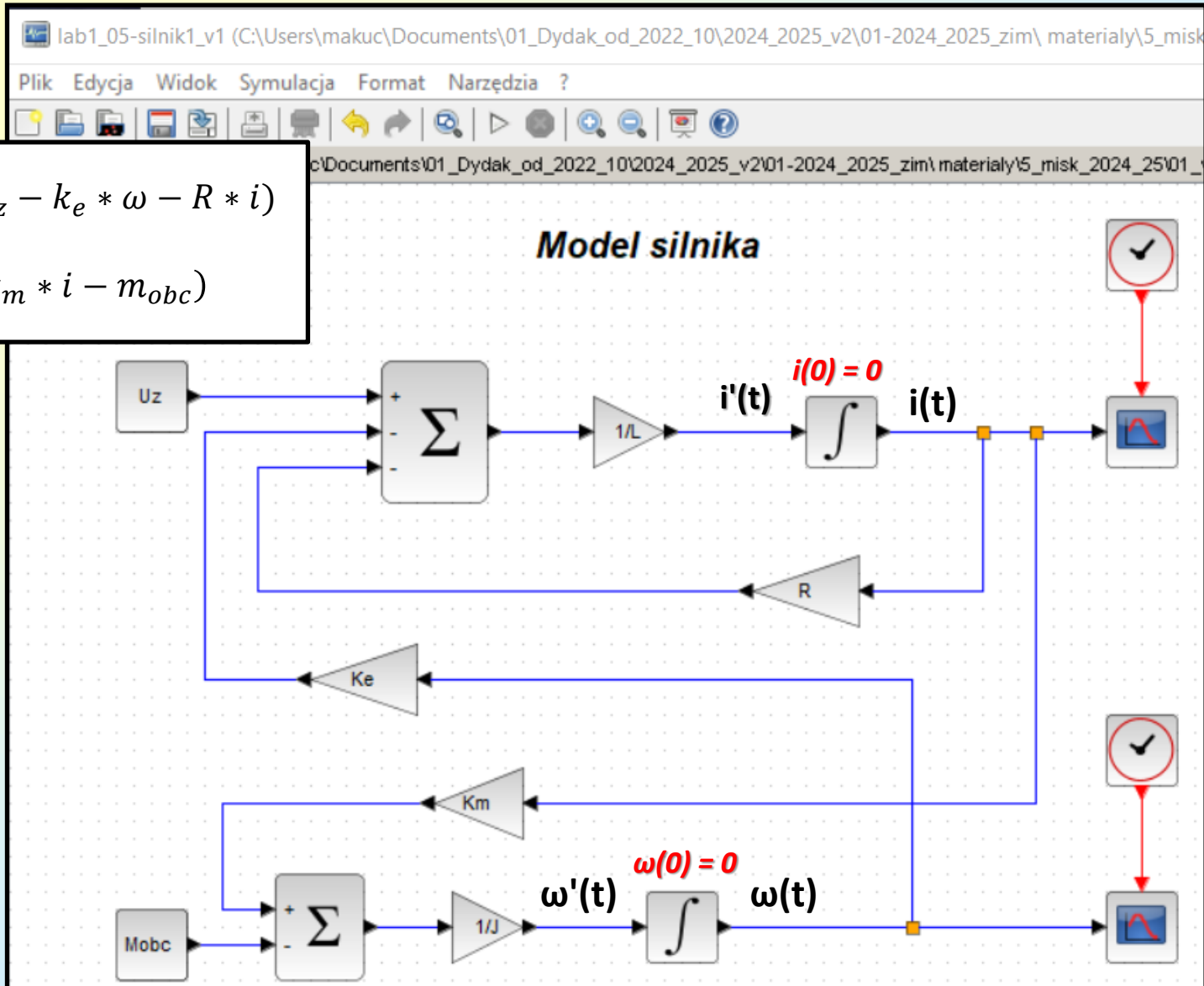
warunki początkowe:  $i(t=0) = i_0$        $\omega(t=0) = \omega_0$

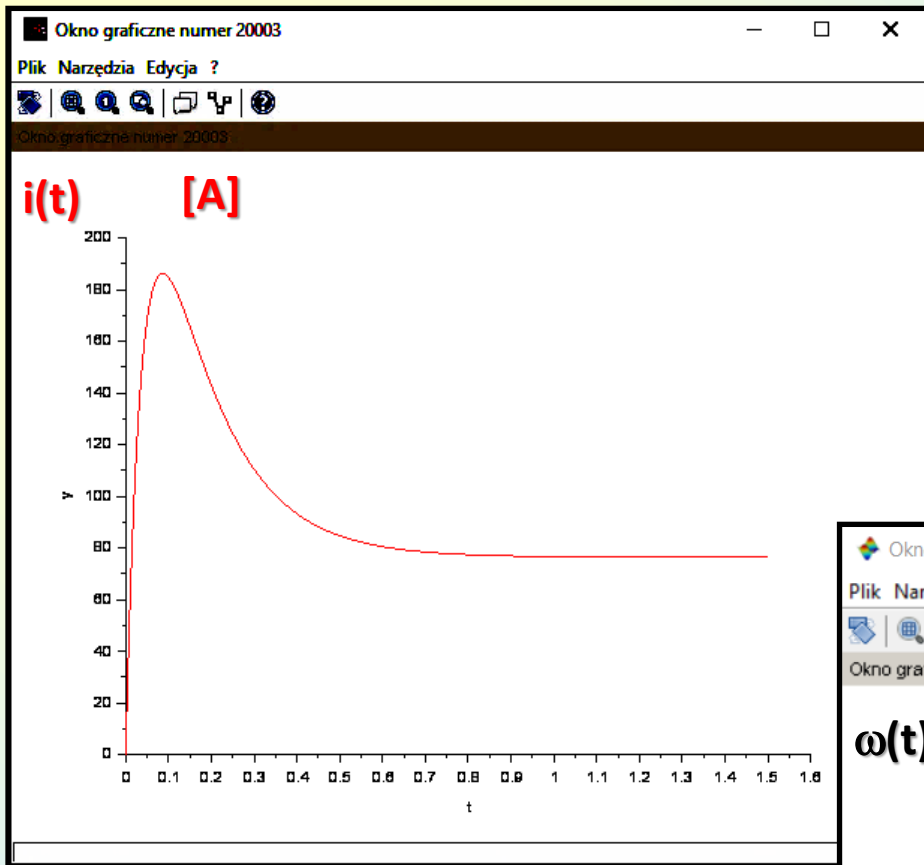
**Symbole literowe w równaniach oznaczają:**

- $u_z, m_{obc}$  – napięcie zasilania i moment obciążenia silnika
- $i, \omega$  – prąd i prędkość kątowa silnika
- $R, L, J, k_e, k_m$  – stałe silnika
- $e$  – siła elektromotoryczna indukowana w tworniku
- $m, m_{dyn}$  – moment obrotowy i moment dynamiczny silnika
- $i_0 = 0$      $\omega_0 = 0$     warunki początkowe

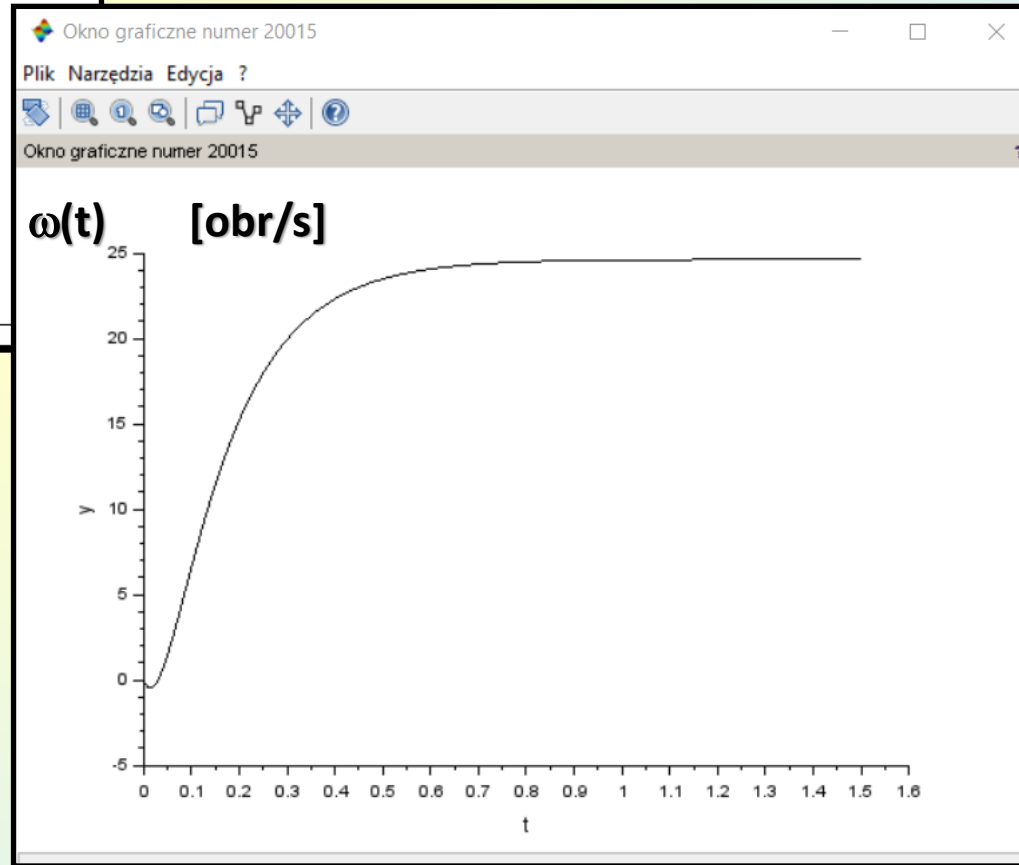
Równania te prezentują **model matematyczny** rozważanego silnika, wynikający wprost z zasady jego działania.

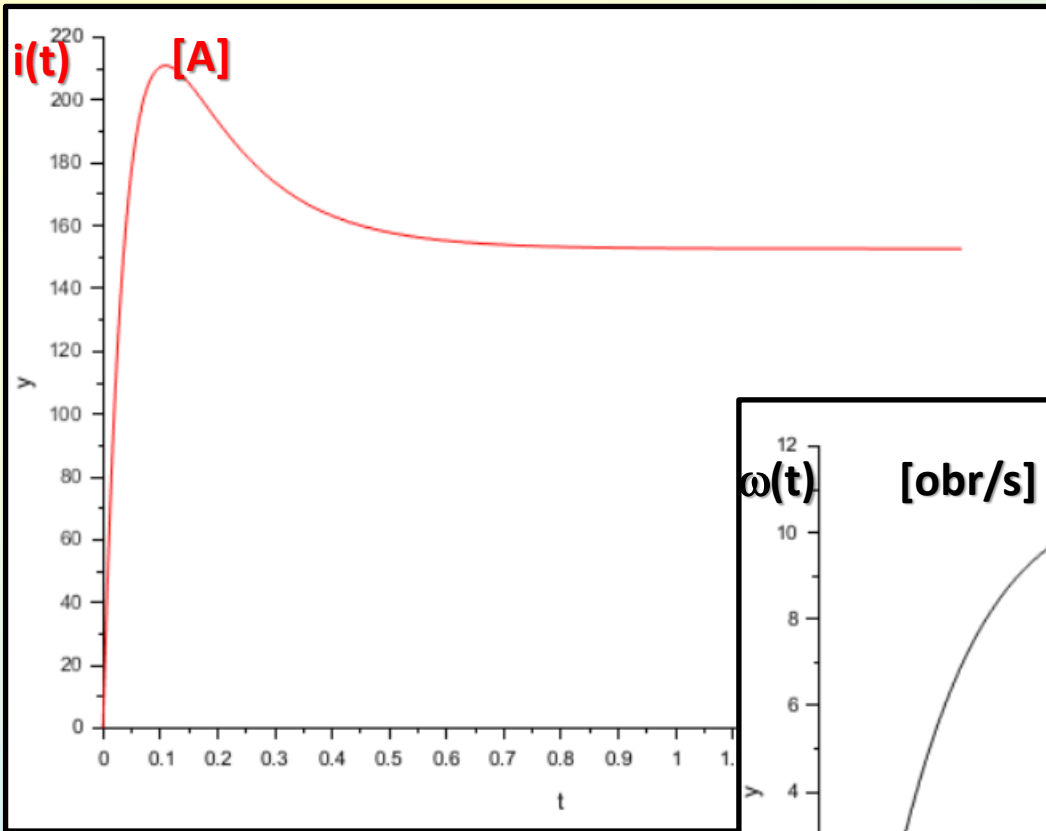
Wartości zmiennych danych do obliczeń:  $u_z = 100 [V]$   $m_{obc} = 200 [Nm]$



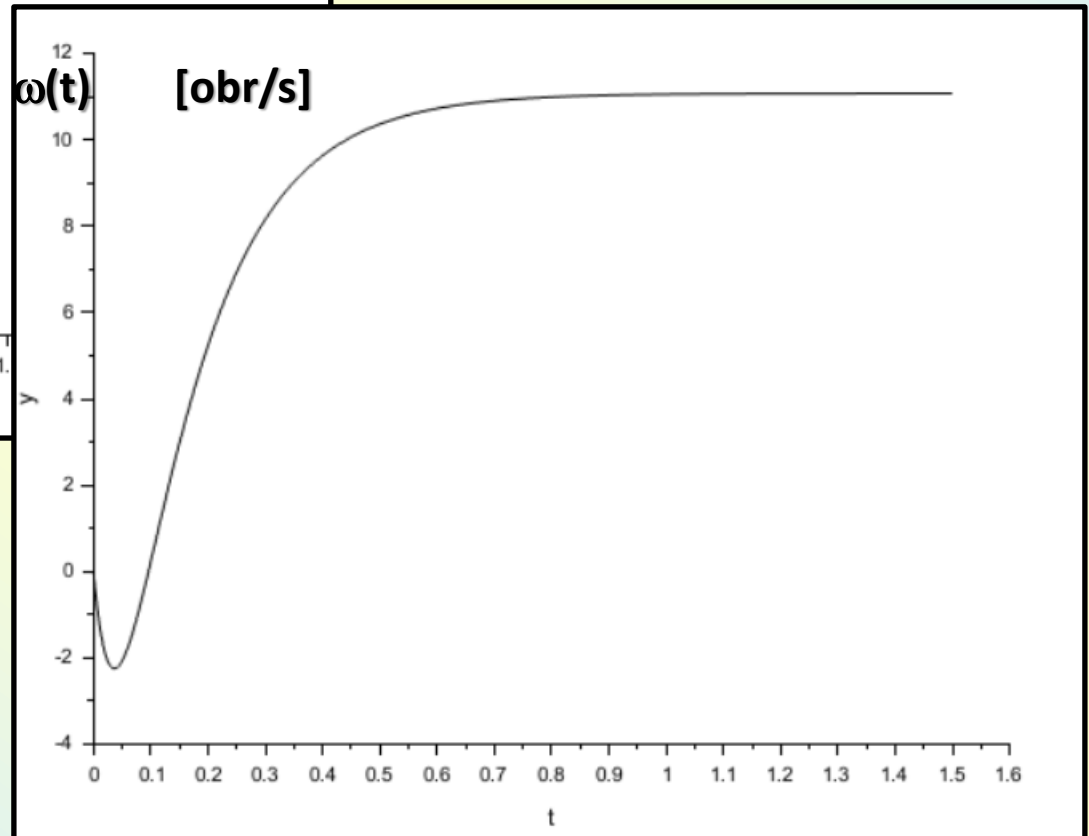


$$u_z = 100 [V] \quad m_{obc} = 200 [Nm]$$

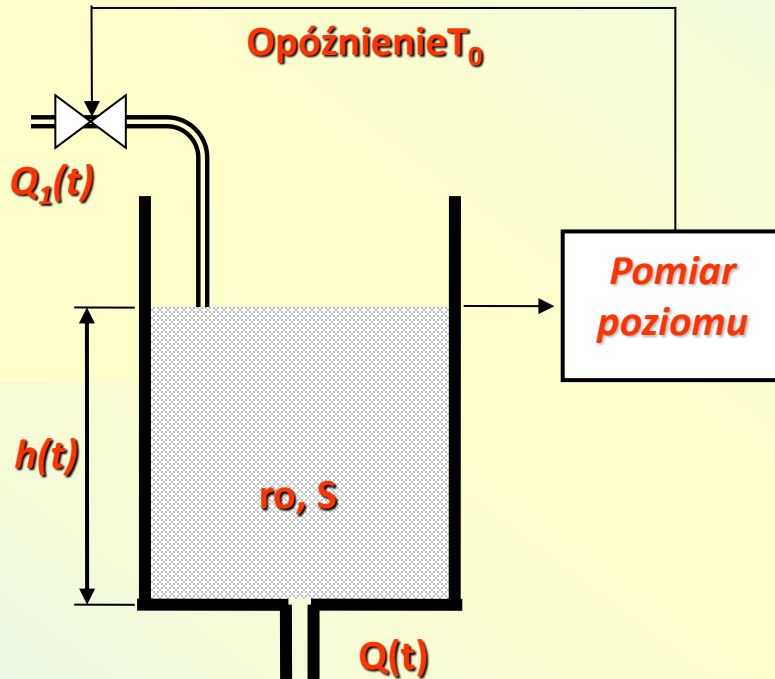




$$u_z = 100 [V] \quad m_{obc} = 400 [Nm]$$



# Regulacja poziomu ciecchy w zbiorniku; pomiar wysokości stupa ciecchy dokonywany jest z opóźnieniem $T_0$



Rozważmy problem związany z regulowanym dopływem ciecchy do zbiornika i jej swobodnym wypływem.

Problem polega na takim regulowaniu dopływu aby przy zadanym wypływie osiągnąć i utrzymać stały poziom ciecchy w zbiorniku.

Do zbiornika o kształcie walca o podstawie  $S$  [m<sup>2</sup>]:

- wpływa ciecch o gęstości  $\rho$  [kg/ m<sup>3</sup>] z szybkością  $Q_1(t)$  [kg/s],
- szybkość ta jest kontrolowana przy pomocy zaworu, który zmniejsza dopływ proporcjonalnie do zmierzonego poziomu  $h(t)$  ciecchy w zbiorniku z pewnym opóźnieniem  $T_0$  [s]
- równocześnie ciecch wypływa rurą wylotową u dołu zbiornika z szybkością  $Q(t)$  [kg/ s] także proporcjonalną do poziomu ciecchy  $h(t)$  [m].

Należy wyznaczyć wysokość poziomu ciecchy  $h(t)$  [m] od chwili  $t=0$ , jeżeli  $h(0)=h_0$  [m], tzn. w chwili początkowej zbiornik był napełniony do wysokości  $h_0$ .

Szybkość wypływu ze zbiornika:

$$Q(t) = k * h(t)$$

Szybkość dopływu do zbiornika przez zawór:  $Q_1(t) = Q_{1m} - k_1 * h(t - T_0)$

gdzie:

$k, k_1$  - odpowiednie współczynniki proporcjonalności,

$T_0$  - czas opóźnienia reakcji zaworu,

$Q_{1m}$  - szybkość dopływu w chwili początkowej (zawór w pełni otwarty) odpowiadająca pełnej wydajności.

Zauważmy, że w równaniu drugim jeżeli  $h(t-T_0)$  rośnie to dopływ  $Q_1(t)$  maleje, a jeżeli  $h(t-T_0)$  maleje to dopływ  $Q_1(t)$  rośnie.

**Równanie równowagi mas:**

$$r_0 * S * \Delta h = [Q_1(t) - Q(t)] * \Delta t$$

W dostatecznie małym przedziale czasu  $dt$  wokół chwili czasowej  $t$  różnica mas dopływu i odpływu musi odpowiadać przyrostowi  $r_0 * S * dh$  masy cieczy w zbiorniku tzn. :

$$r_0 * S * dh = [Q_1(t) - Q(t)] * dt$$

$$r_0 * S * \frac{dh}{dt} = Q_{1m} - k_1 * h(t - T_0) - k * h(t)$$

wyznaczając pochodną  $dh/dt$  mamy:

$$\frac{dh}{dt} = \frac{Q_{1m}}{r_0 * S} - \frac{k_1}{r_0 * S} * h(t - T_0) - \frac{k}{r_0 * S} * h(t)$$

ponieważ wartość  $r_0 * S$  jest stała, to równanie możemy zapisać w postaci:

$$\frac{dh}{dt} = Q_{1m}^* - k_1^* * h(t - T_0) - k^* * h(t)$$

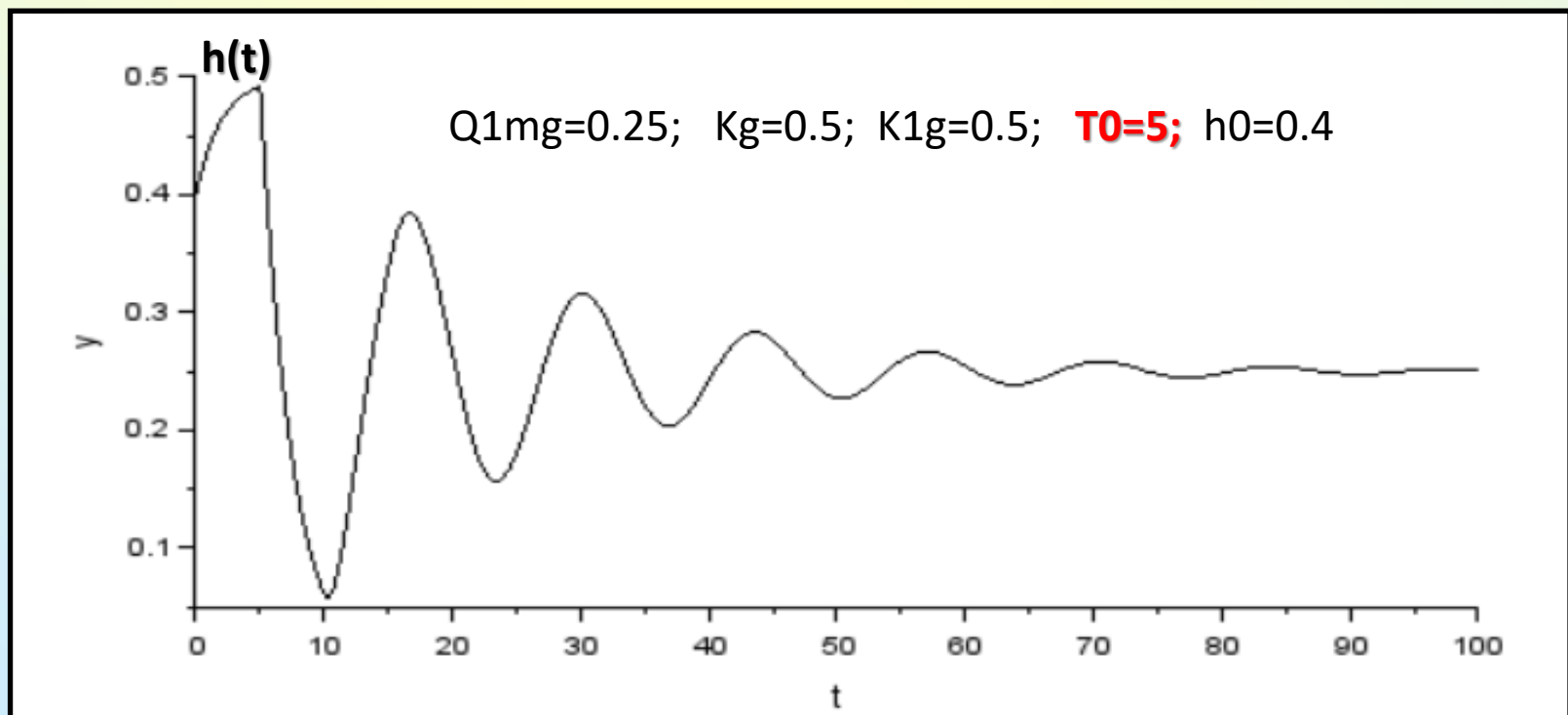
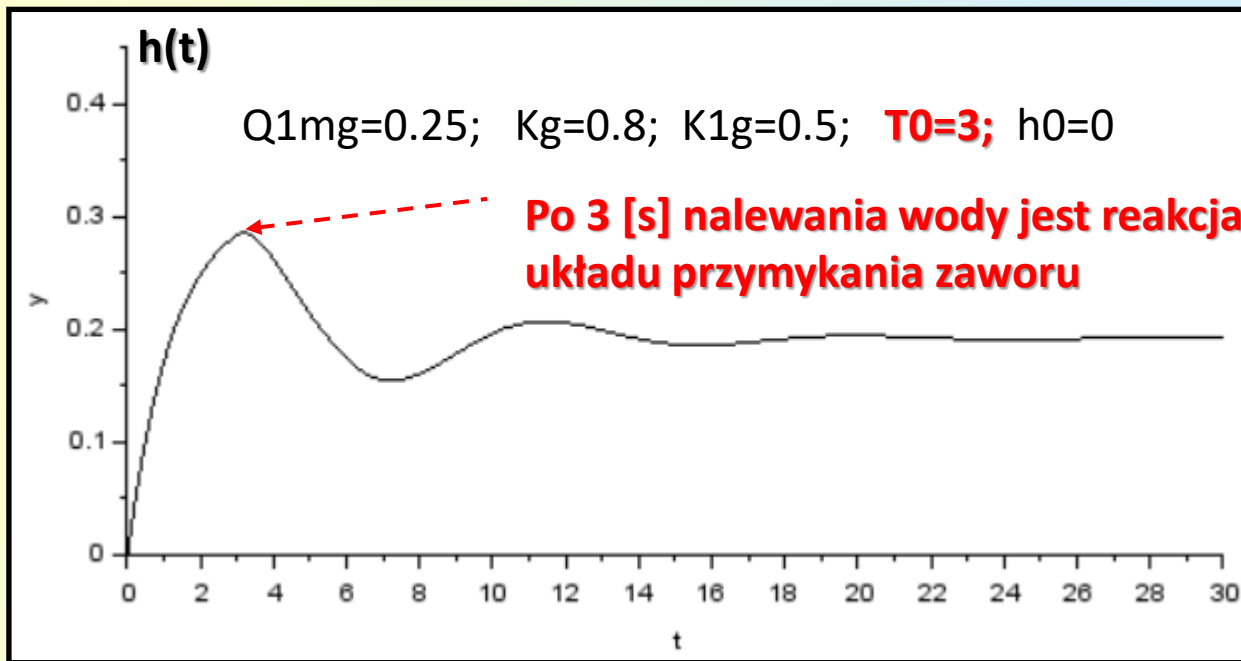
$h(t)$  – to poszukiwane rozwiązanie, a  $h(0)=h_0$  to warunek początkowy mówiący o wstępnym wypełnieniu zbiornika.

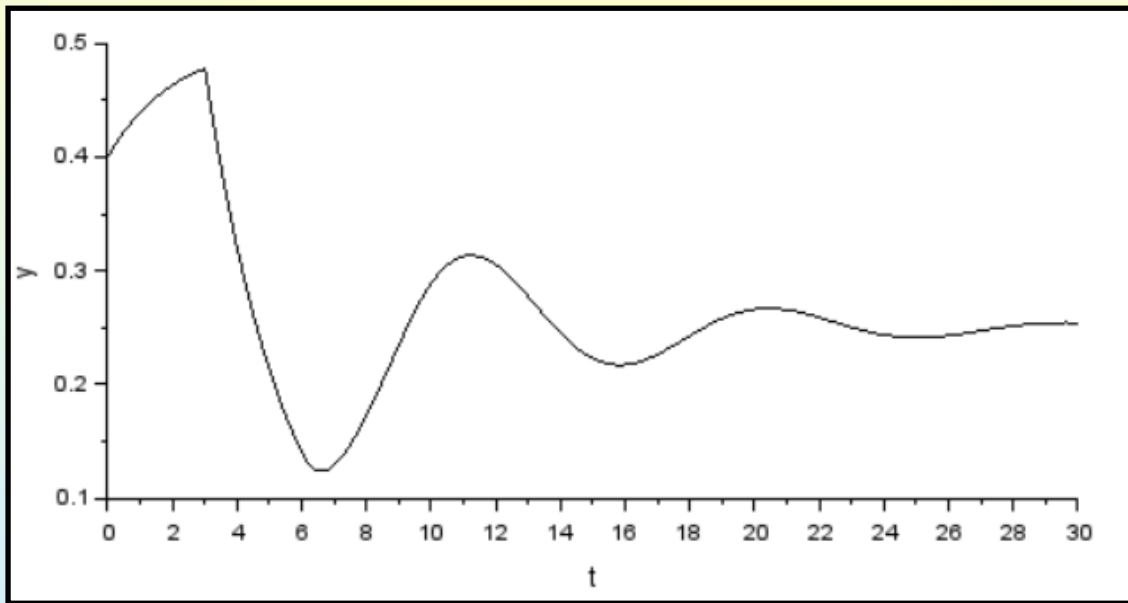
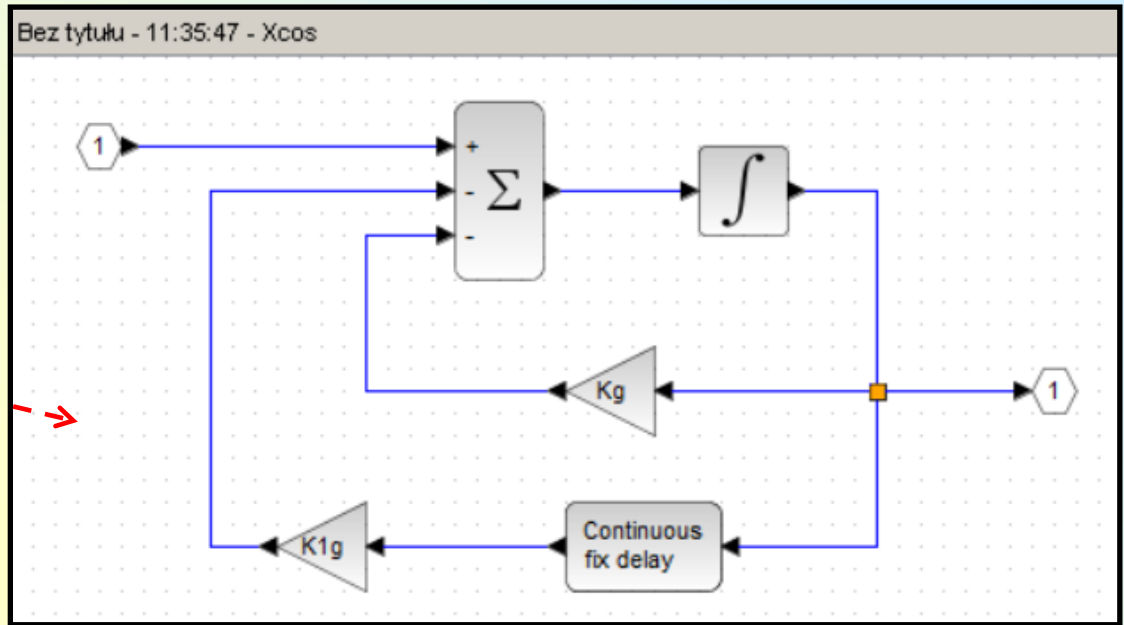
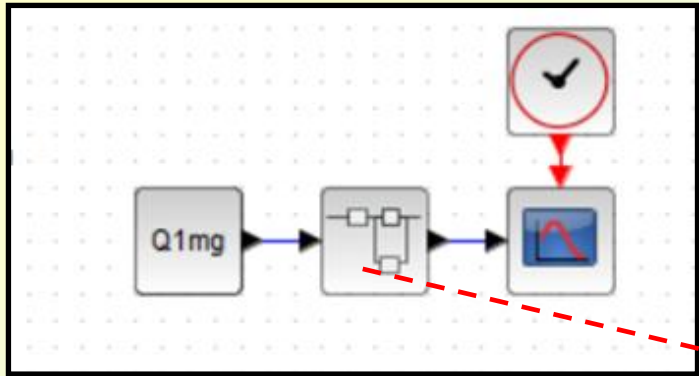
przy czym:

$Q_{1m}^*$  wydajność dopływu wyrażona w [m/s],  
 $k_1^*$ ,  $k^*$  współczynniki, które mają wymiar [1/s].

W modelu symulacyjnym należy użyć bloku opóźnienia: **TIME\_DELAY**

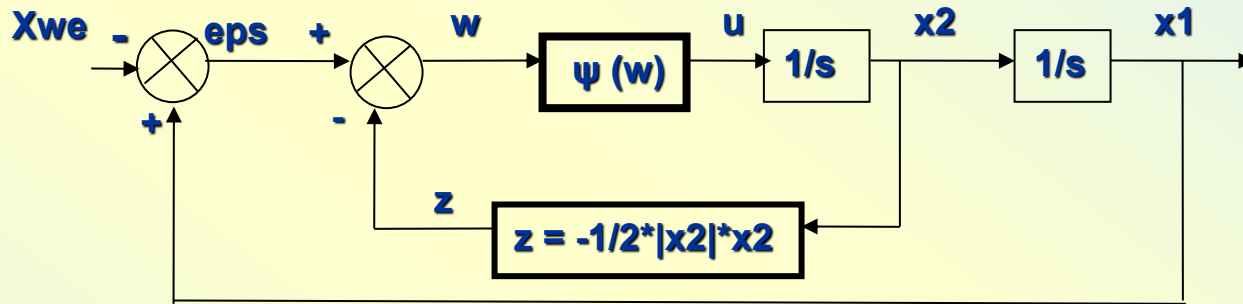






# Nieliniowy układ regulacji ze sprzężeniem zwrotnym wariant 1

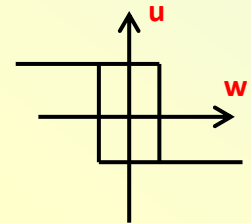
Zadaniem przedstawionego na rysunku układu regulacji jest sprowadzenie w najkrótszym czasie sygnału  $x_1$  do wartości  $s$  oraz sygnału  $x_2$  do zera.



Przedstawić zachowanie się układu regulacji przez podanie przebiegów czasowych sygnałów modelujących zmienne  $x_1$  i  $x_2$  oraz przez wykreślenie trajektorii na płaszczyźnie fazowej. Element nieliniowy o charakterystyce  $u = \psi(w)$  może być:

przełącznikiem  
dwupołożeniowym  
z histerezą  $|w| = 0.025$

$$u = \Psi_1(w) = \begin{cases} -1 & \text{gdy } w > 0.025 \\ \text{histereza} & \text{gdy } |w| \leq 0.025 \\ 1 & \text{gdy } w < -0.025 \end{cases}$$



Obliczenia wykonać dla warunków początkowych:

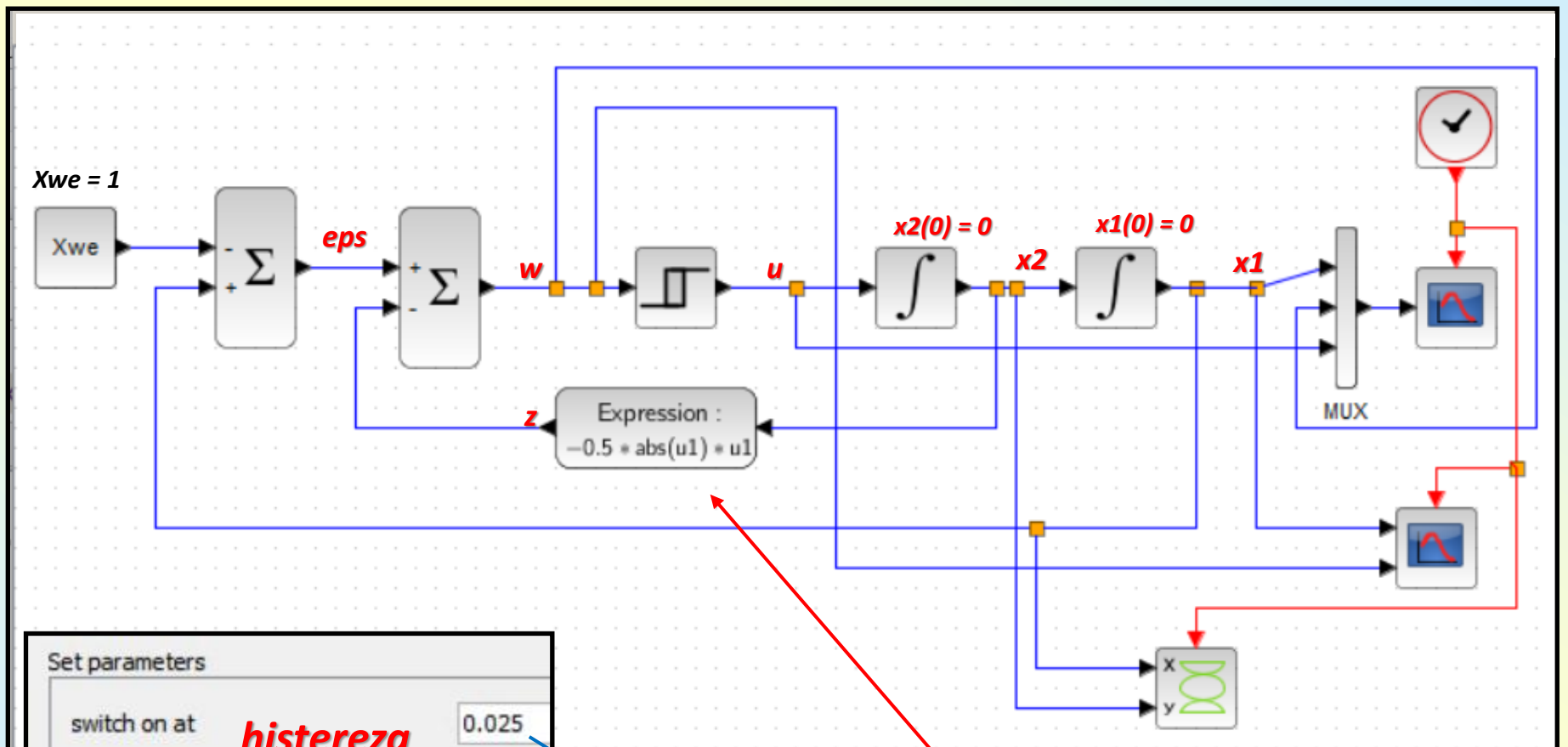
$$Xwe = 1;$$

$$x_1(0) = 0;$$

$$x_2(0) = 0$$

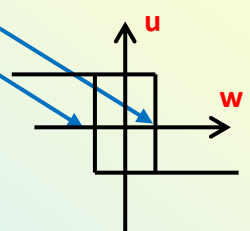
# przełącznik dwupołożeniowy z histerezą

$$u = \Psi_1(w) = \begin{cases} -1 & \text{gdy } w > 0.025 \\ \text{histereza} & \text{gdy } |w| \leq 0.025 \\ 1 & \text{gdy } w < -0.025 \end{cases}$$

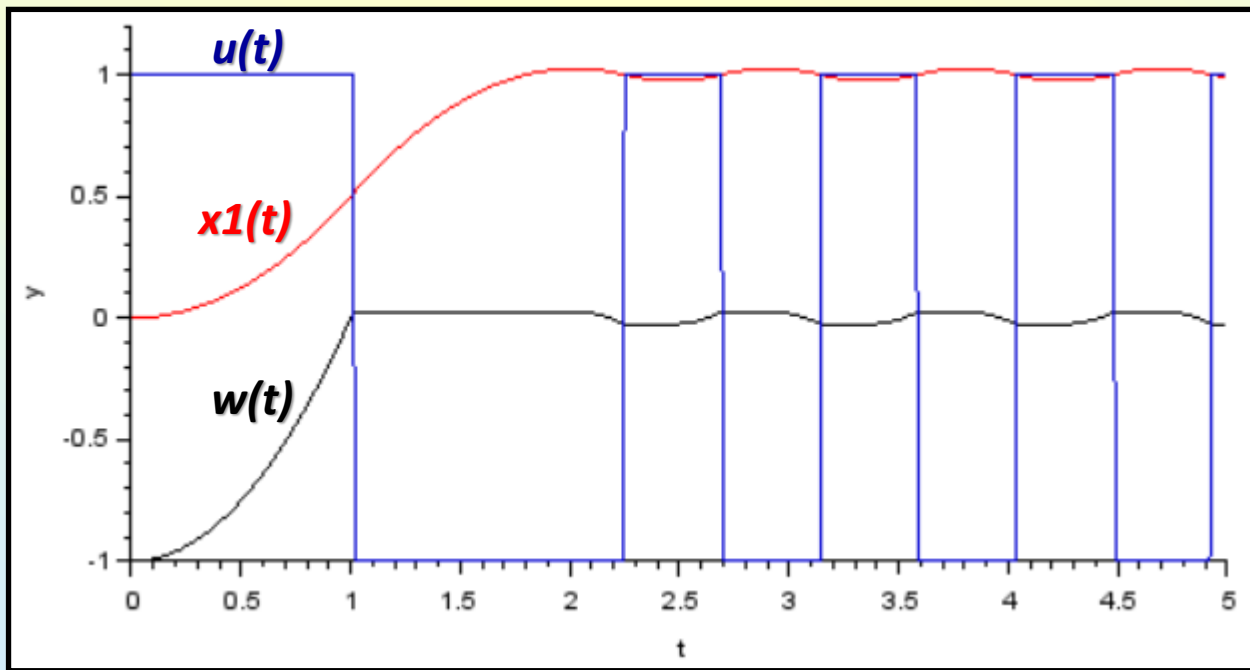
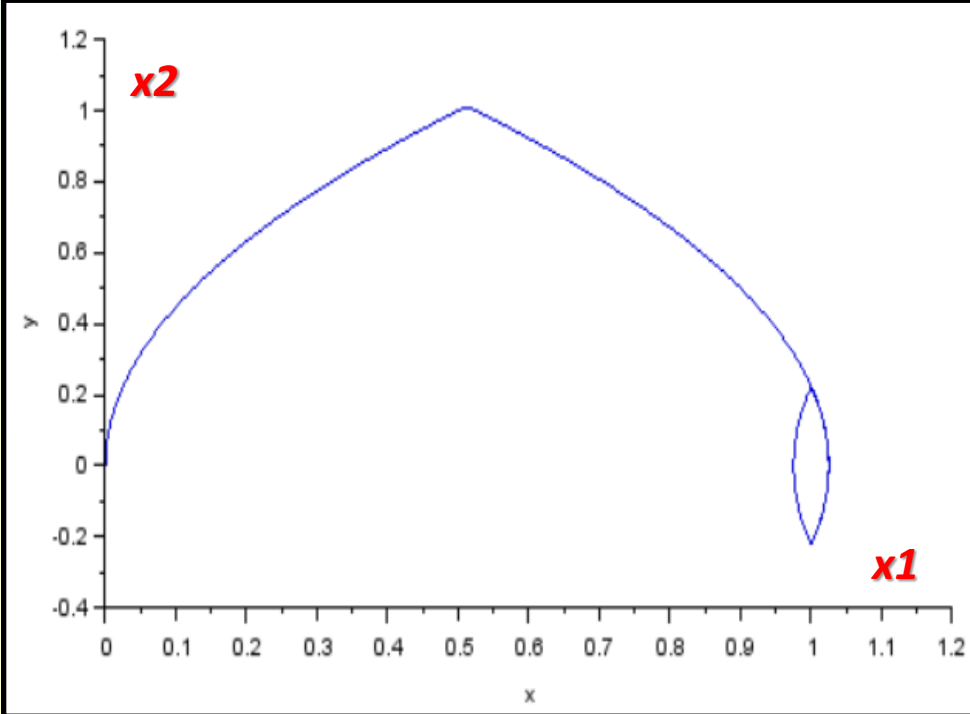
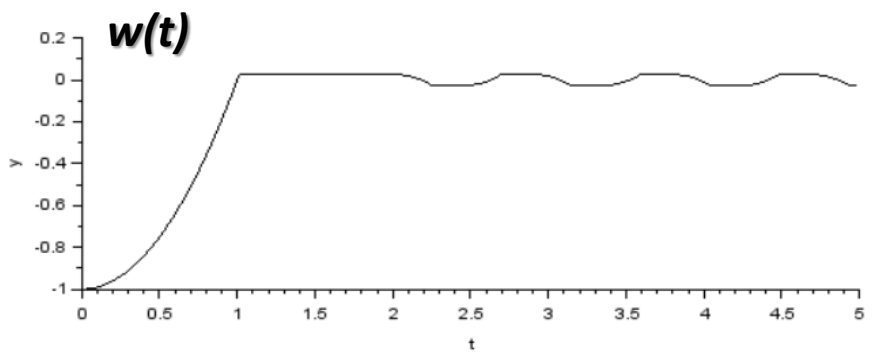
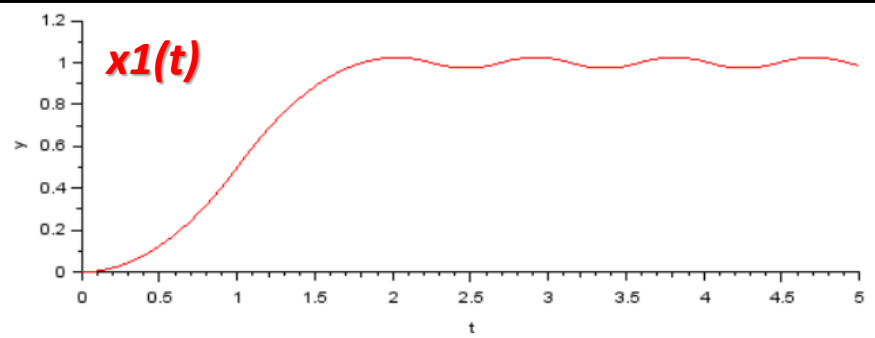


Set parameters

switch on at	<b>histereza</b>	0.025
switch off at		-0.025
output when on		-1
output when off		1
use zero crossing:		yes (1), no (0) 1

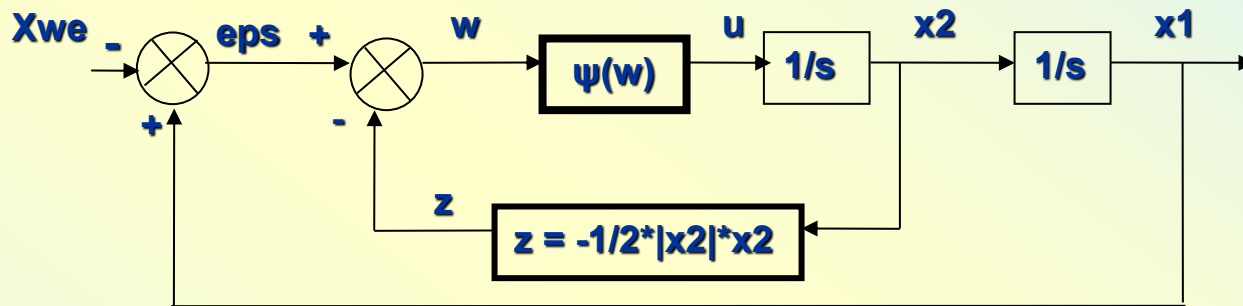


$$z = -1/2 * |x2|^2 * x2$$



# Nieliniowy układ regulacji ze sprzężeniem zwrotnym wariant 2

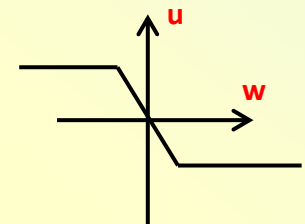
Zadaniem przedstawionego na rysunku układu regulacji jest sprowadzenie w najkrótszym czasie sygnału  $x_1$  do wartości  $s$  oraz sygnału  $x_2$  do zera.



Przedstawić zachowanie się układu regulacji przez podanie przebiegów czasowych sygnałów modelujących zmienne  $x_1$  i  $x_2$  oraz przez wykreślenie trajektorii na płaszczyźnie fazowej. Element nieliniowy o charakterystyce  $u = \psi(w)$  może być:

przełącznikiem typu element z nasyceniem

$$u = \Psi_2(w) = \begin{cases} 1 & \text{gdy } w < -0.025 \\ -40 * w & \text{gdy } |w| \leq 0.025 \\ -1 & \text{gdy } w > 0.025 \end{cases}$$



Obliczenia wykonać dla warunków początkowych:

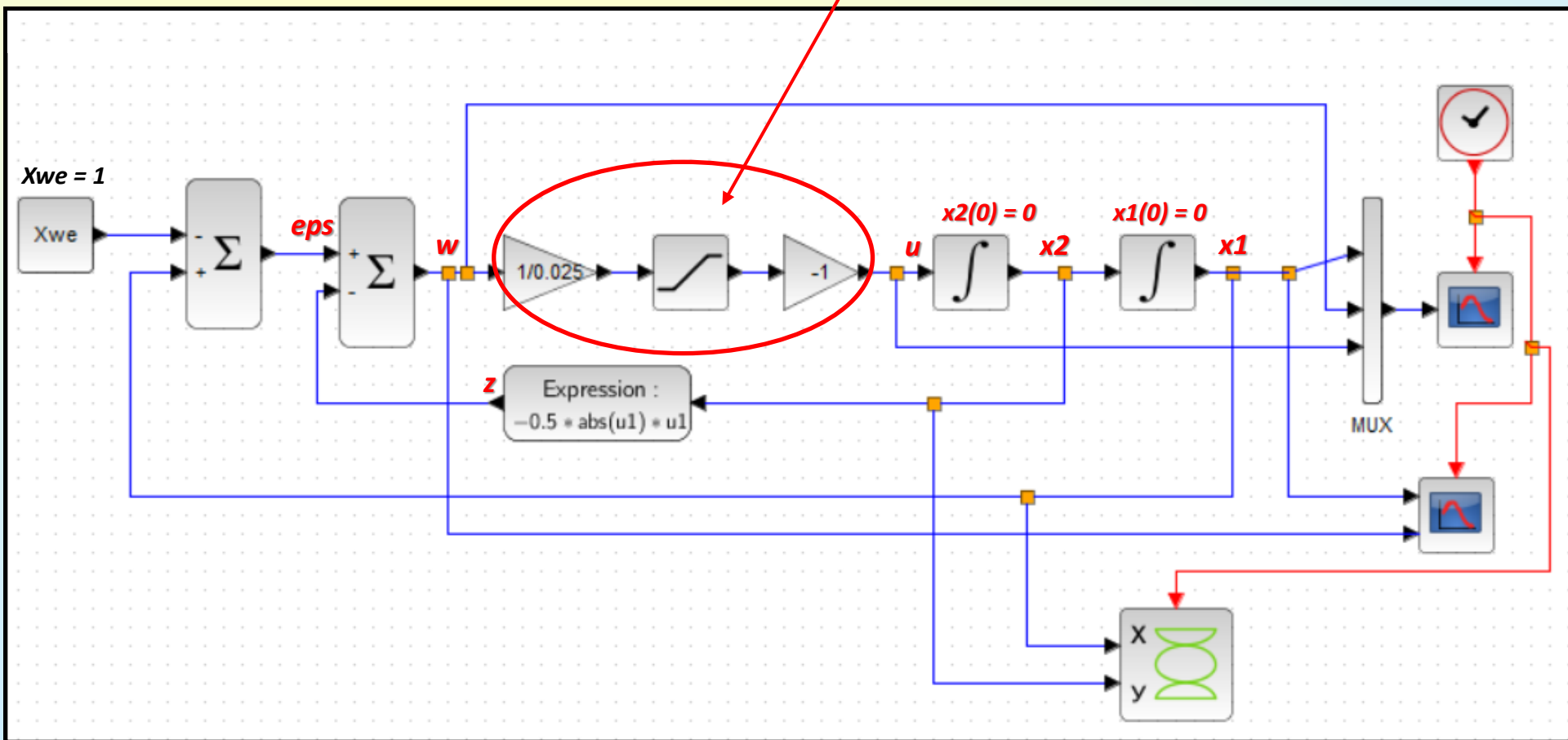
$$Xwe = 1;$$

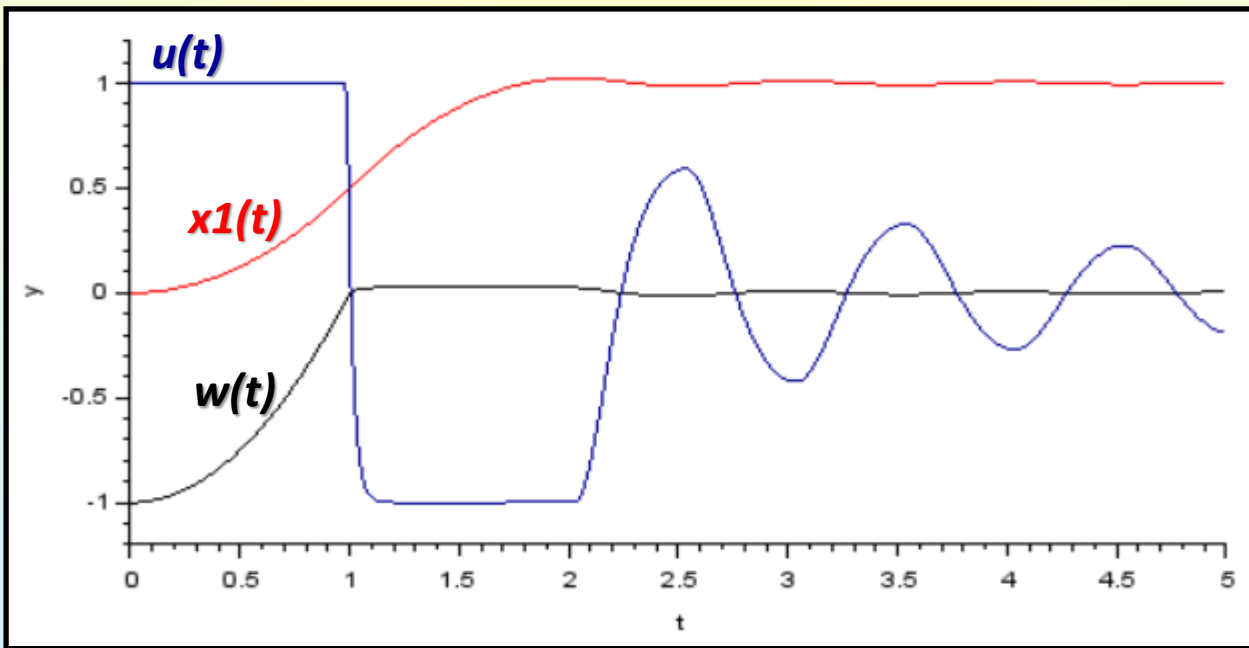
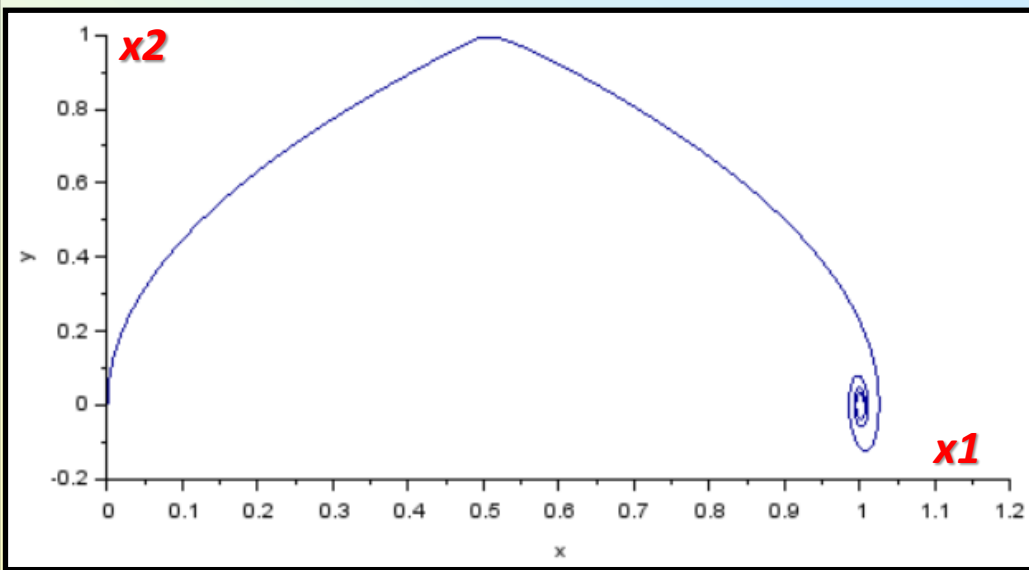
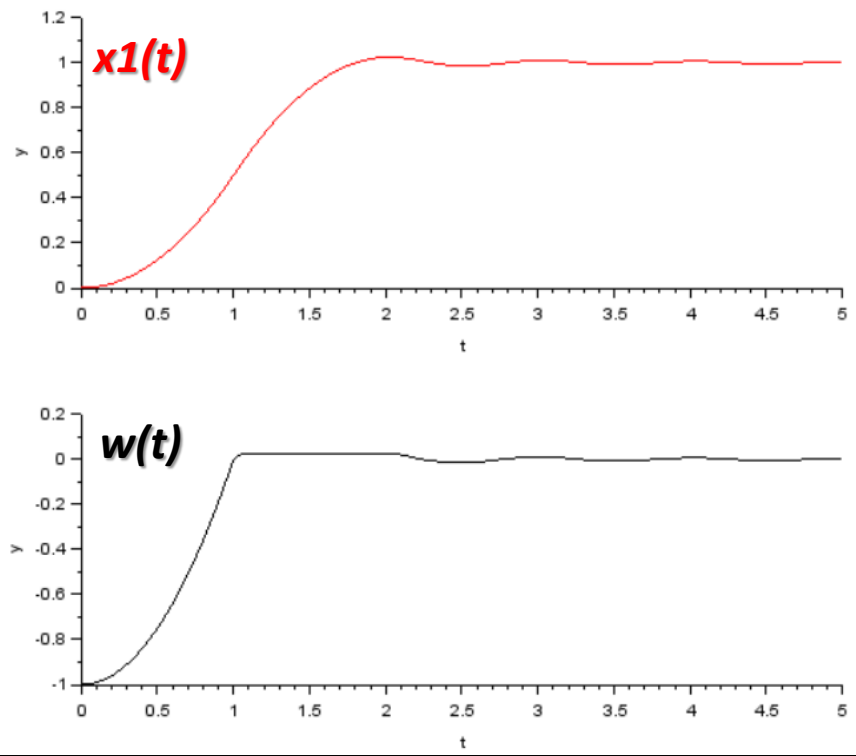
$$x_1(0) = 0;$$

$$x_2(0) = 0$$

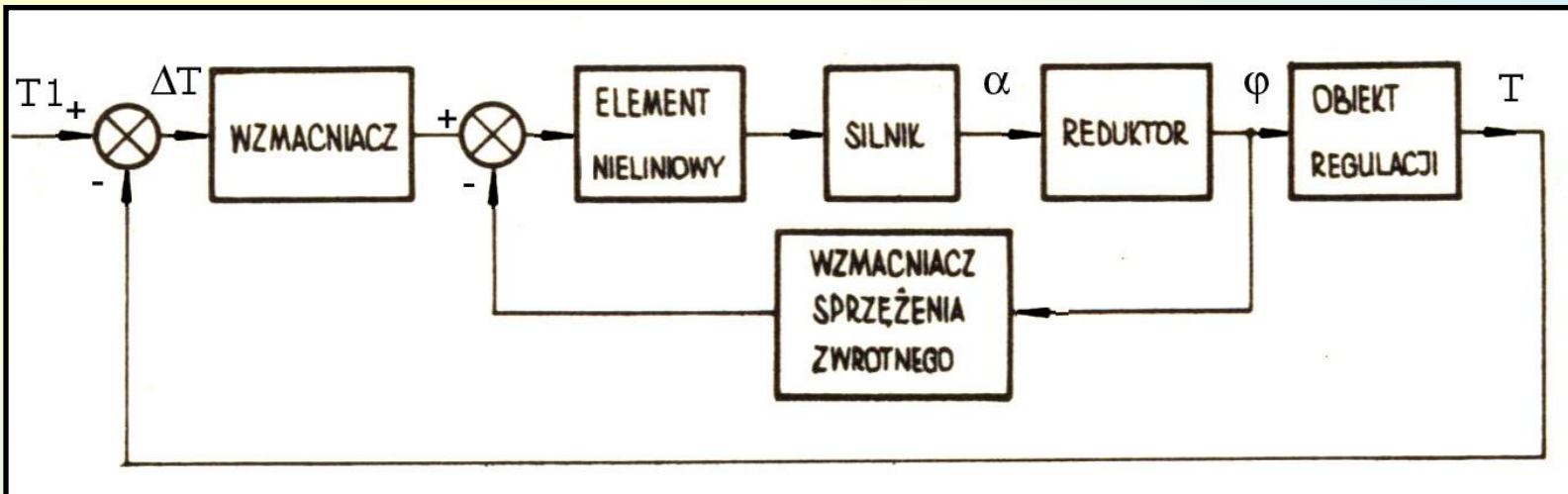
# przełącznik typu element z nasyceniem

$$u = \Psi_2(w) = \begin{cases} 1 & \text{gdy } w < -0.025 \\ -40 * w & \text{gdy } |w| \leq 0.025 \\ -1 & \text{gdy } w > 0.025 \end{cases}$$





# Układ stabilizacji temperatury ze sprzężeniem zwrotnym ujemnym



**schemat blokowy układu stabilizacji temperatury**

**Wprowadzone na rysunku wielkości mają następujące znaczenie:**

**$T_1$  - zadana temperatura obiektu**

**$\varphi$  - kąt obrotu organu regulacyjnego (reduktor)**

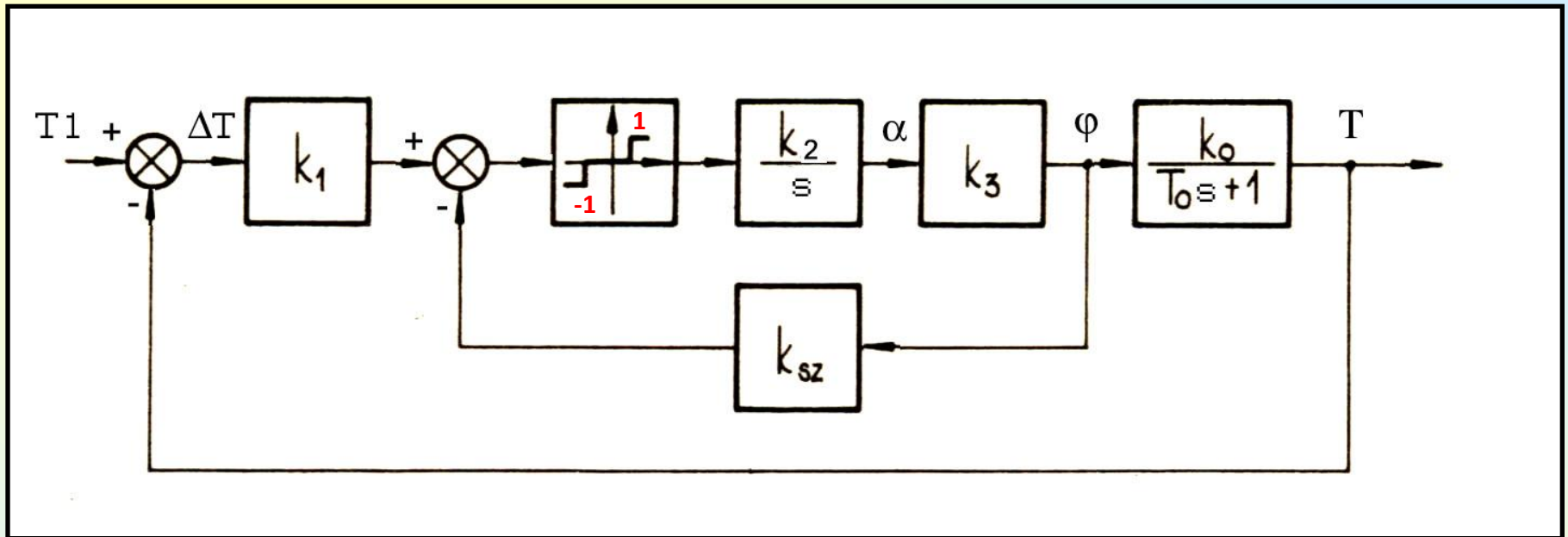
**$\alpha$  - kąt obrotu organu wykonawczego (silnik)**

**$T$  - rzeczywista wartość temperatury obiektu**

**$\Delta T$  - błąd regulacji (uchyb)  $\Delta T = T - T_1$**

**Układ ten służy do utrzymania temperatury  $T$  obiektu regulowanego na poziomie równym temperaturze zadanej  $T_1$  z określonym z góry błędem regulacji.**

## Schemat strukturalny układu stabilizacji temperatury



### Dane liczbowe:

$$T_1 = 1 \text{ [}^\circ\text{C]}$$

$$K_o = 25 \text{ [}^\circ\text{C/rad]}$$

$$T_o = 2.5 \text{ [s]}$$

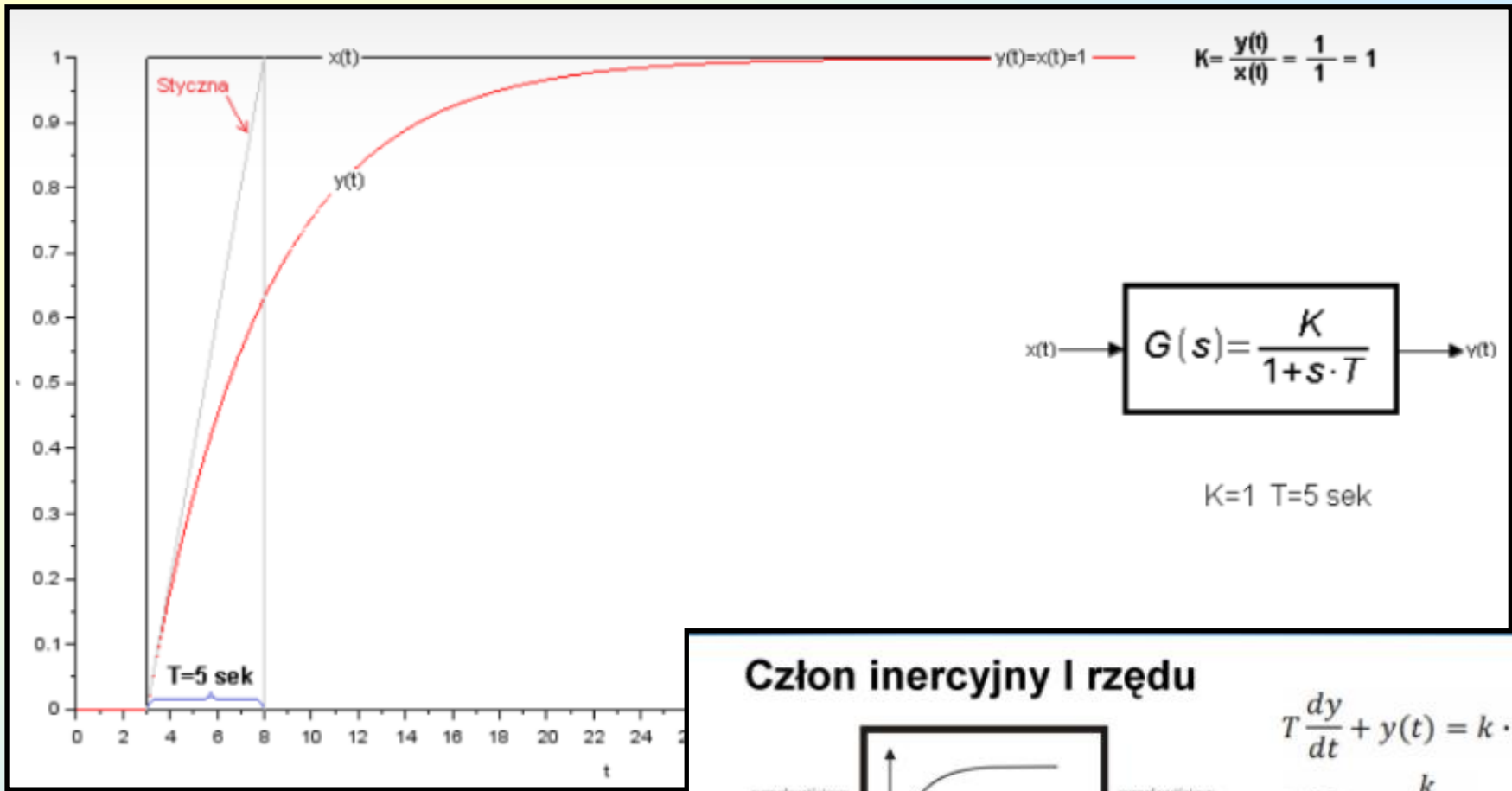
$$K_1 = 25 \text{ [amperozwoi/}^\circ\text{C]}$$

$$K_2 = 2 \text{ [rad/s]}$$

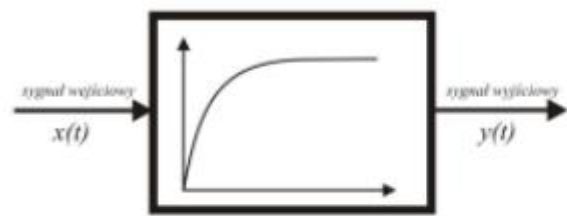
$$K_3 = 0.1 \text{ (reduktor)}$$

$$K_{sz} = 2.5 \text{ [amperozwoi/rad]}$$

$$\text{szerokość strefy martwej [-0.18; 0.18]}$$



### Człon inercyjny I rzędu

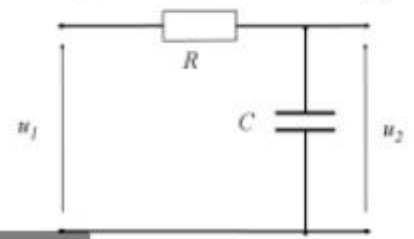


$$T \frac{dy}{dt} + y(t) = k \cdot x(t)$$

$$G(s) = \frac{k}{Ts + 1}$$

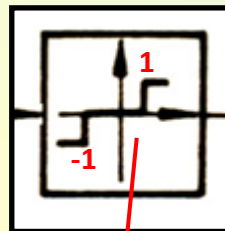
$k$  – współczynnik wzmocnienia  
 $T$  – stała czasowa

Przykład członu inercyjnego 1 rzędu:



$$G(s) = \frac{U_2(s)}{U_1(s)} = \frac{1}{RCs + R}$$

Set Deadband parameters	
End of dead band	0.18
Start of dead band	-0.18
zero crossing (0:no, 1:yes)	1



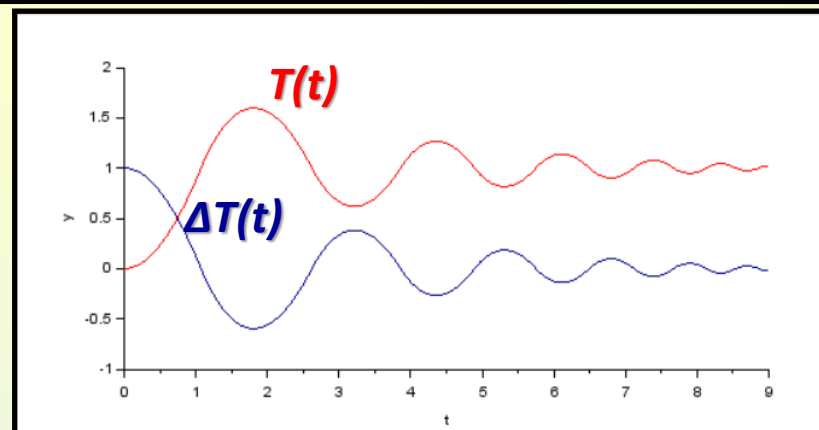
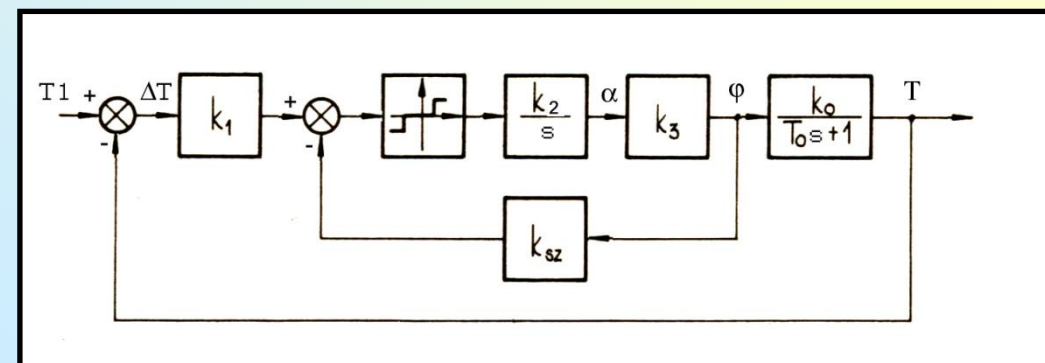
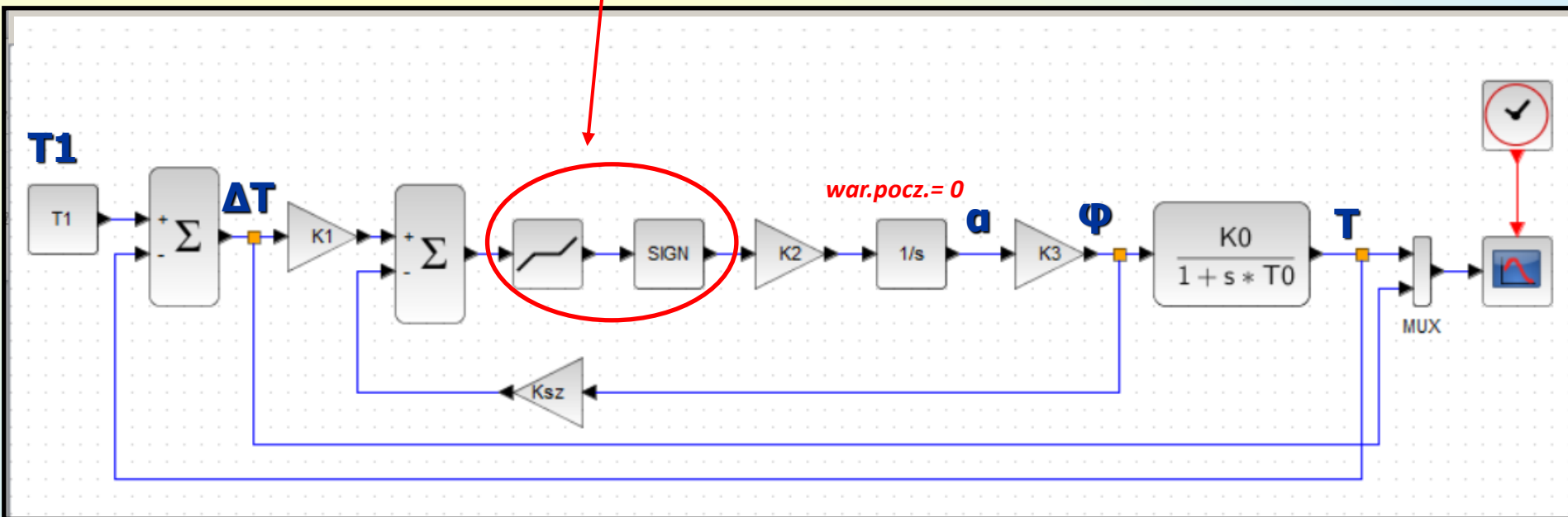
**Dane liczbowe:**

$T1 = 1; K0 = 25; T0 = 2.5$

$K1 = 25; K2 = 2; K3 = 0.1; Ksz = 2.5$

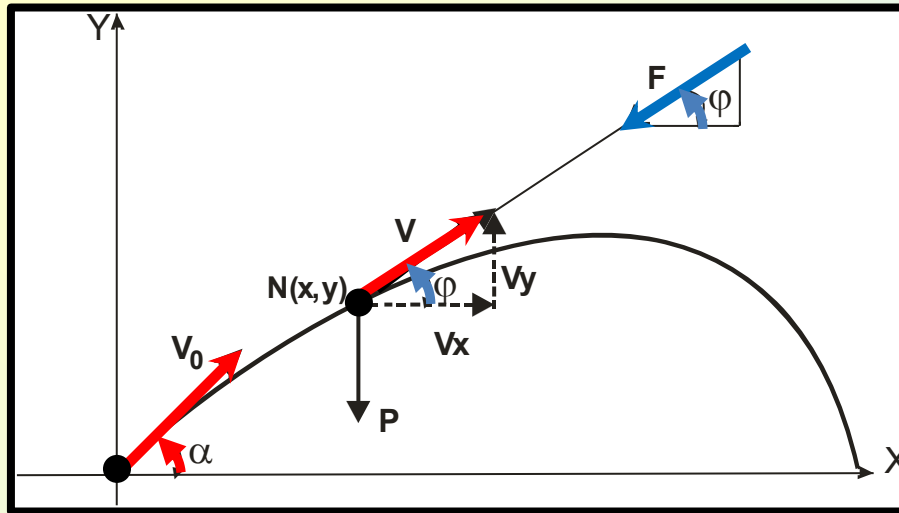
**szerokość strefy martwej [-0.18; 0.18]**

**czas obliczeń 9 [s]**



# Krzywa balistyczna

Kula o masie  $m$ , wystrzelona pod kątem  $\alpha$  do poziomu z prędkością początkową  $v_0$ , porusza się w ośrodku, którego opór jest proporcjonalny do kwadratu prędkości  $v$ . Wyznaczyć tor pocisku.



W dowolnym punkcie  $N(x, y)$  toru na ciało  $m$  działają dwie siły:

$P = m * g$  siła ciężkości oraz

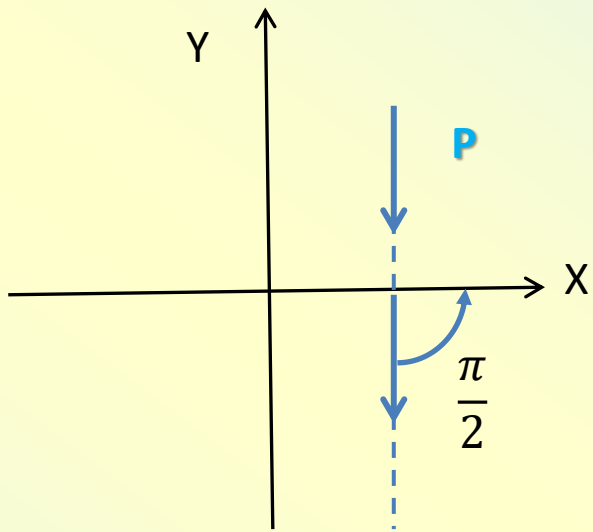
$F = f * v^2$  siła oporu ośrodka

Składowe rzuty wypadkowych tych sił na osie układu mają postać:

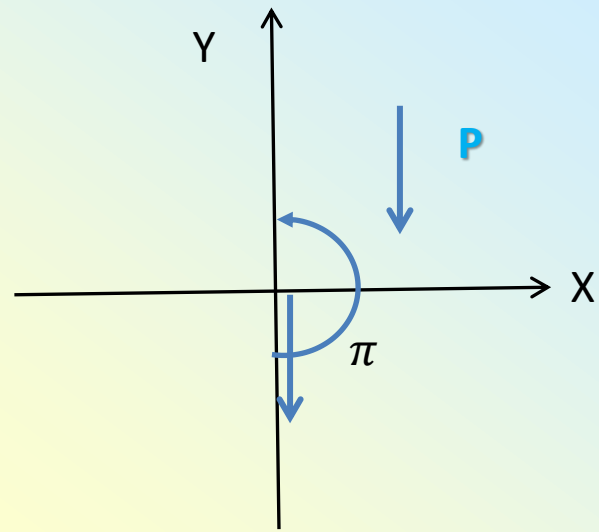
$$X = P * \cos(P, x) + F * \cos(F, x) \quad \cos(P, x) = 0 \quad i \quad \cos(F, x) = -\cos(\varphi)$$

gdzie:

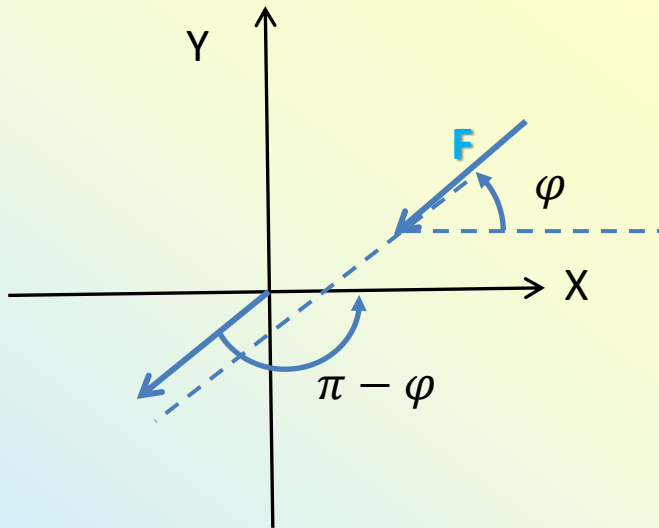
$$Y = P * \cos(P, y) + F * \cos(F, y) \quad \cos(P, y) = -1 \quad i \quad \cos(F, y) = -\cos\left(\frac{\pi}{2} - \varphi\right) = -\sin(\varphi)$$



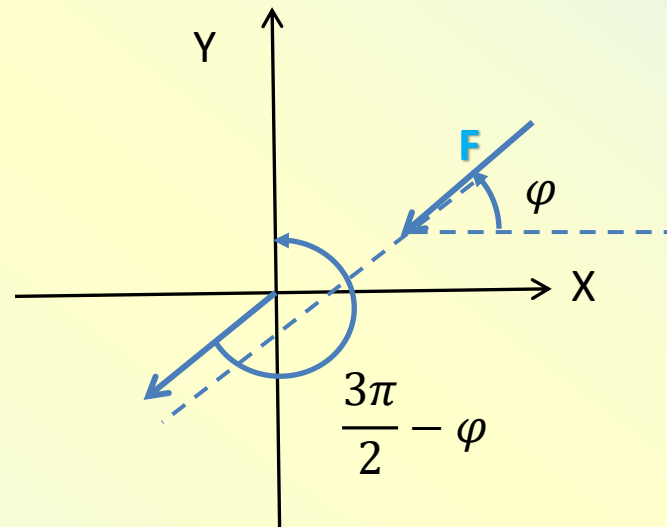
$$\cos(P, x) = \cos\left(\frac{\pi}{2}\right) = 0$$



$$\cos(P, y) = \cos(\pi) = -1$$



$$\cos(F, x) = \cos(\pi - \varphi) = -\cos(\varphi)$$



$$\cos(F, y) = \cos\left(\frac{3\pi}{2} - \varphi\right) = -\sin(\varphi)$$

Na podstawie II prawa dynamiki Newtona można napisać:

$$\begin{cases} m * \frac{d^2x}{dt^2} = X \\ m * \frac{d^2y}{dt^2} = Y \end{cases}$$

Otrzymujemy zatem:

$$\begin{cases} m * \frac{d^2x}{dt^2} = -F * \cos(\varphi) \\ m * \frac{d^2y}{dt^2} = -P - F * \sin(\varphi) \end{cases} \quad \text{gdzie:} \quad P = m * g \quad \text{i} \quad F = f * v^2$$

Ponadto zachodzi:

$$\cos(\varphi) = \frac{v_x}{v} = \frac{dx}{dt} / v \quad \sin(\varphi) = \frac{v_y}{v} = \frac{dy}{dt} / v \quad v = \sqrt{v_x^2 + v_y^2}$$

Model matematyczny procesu ma więc postać:

$$\begin{cases} \frac{d^2x}{dt^2} = -\frac{f}{m} * \frac{1}{v} * \frac{dx}{dt} \\ \frac{d^2y}{dt^2} = -g - \frac{f}{m} * v * \frac{dy}{dt} \\ v = \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2} \end{cases}$$

Warunki początkowe:

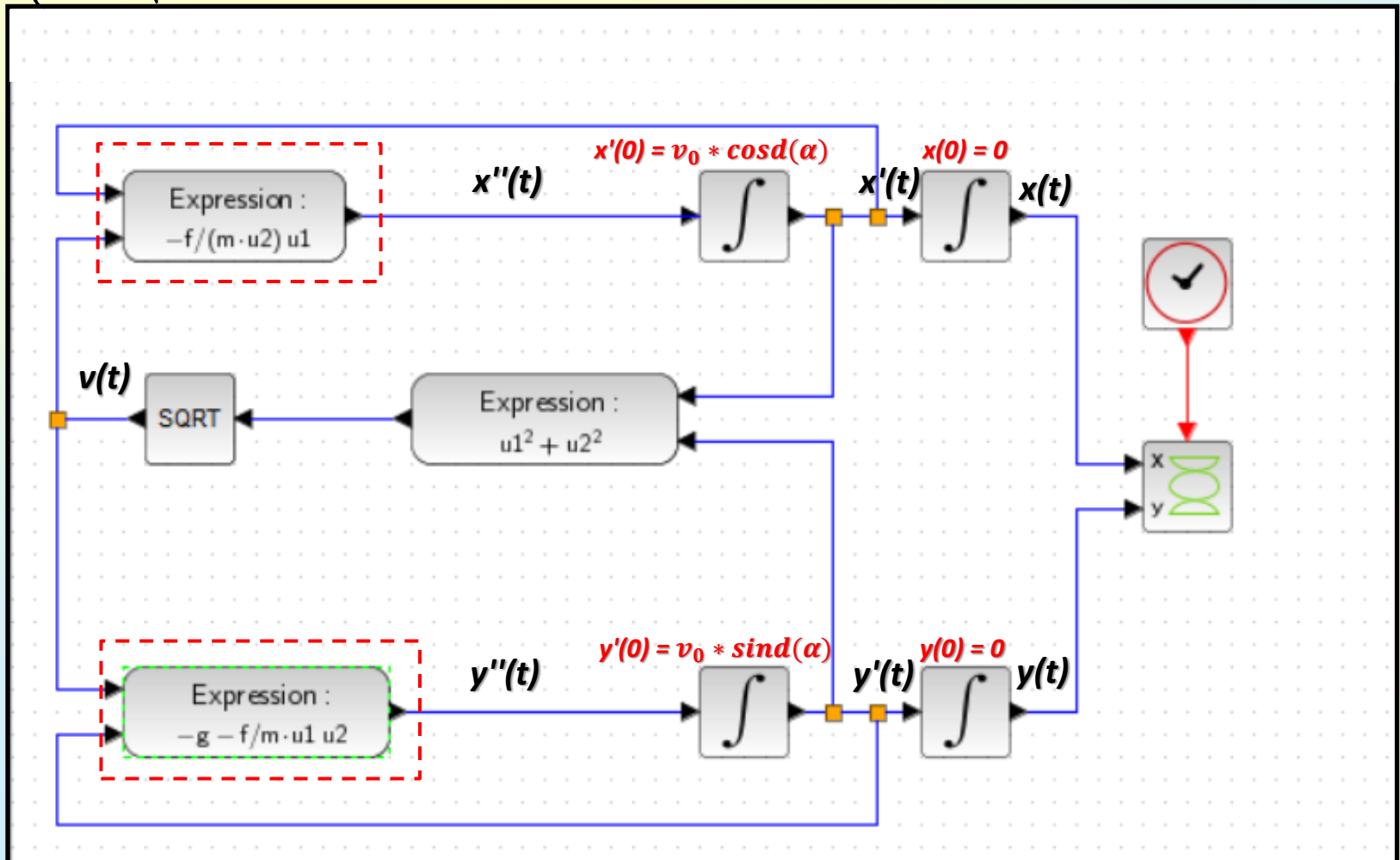
$$x(0) = 0 \quad x'(0) = v_x(0) = v_0 * \cos(\alpha)$$

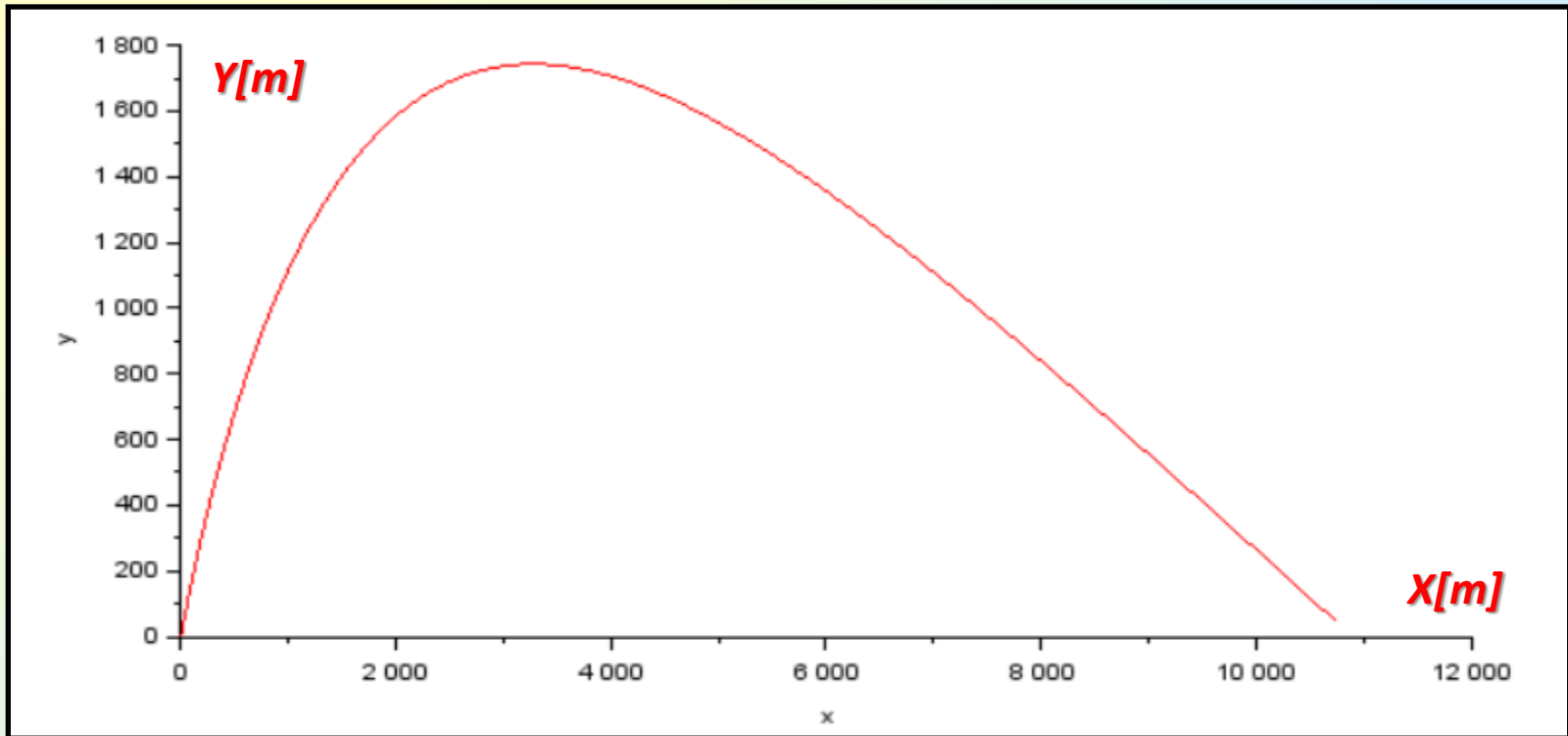
$$y(0) = 0 \quad y'(0) = v_y(0) = v_0 * \sin(\alpha)$$

$$\begin{cases} \frac{d^2x}{dt^2} = -\frac{f}{m} * \frac{1}{v} * \frac{dx}{dt} \\ \frac{d^2y}{dt^2} = -g - \frac{f}{m} * v * \frac{dy}{dt} \\ v = \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2} \end{cases}$$

$$x(0) = 0 \quad x'(0) = v_x(0) = v_0 * \cos(\alpha)$$

$$y(0) = 0 \quad y'(0) = v_y(0) = v_0 * \sin(\alpha)$$



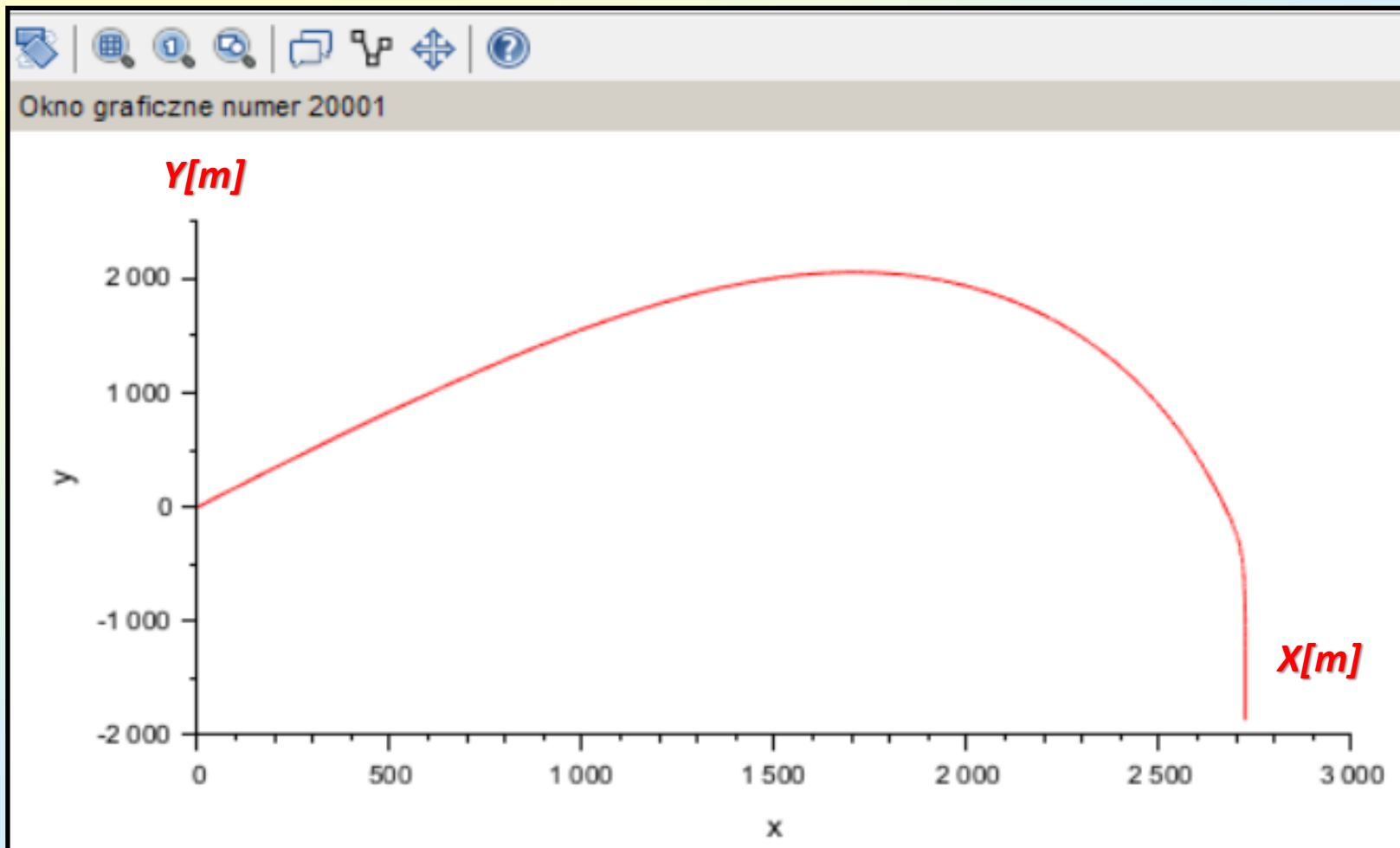


**Tor pocisku dla danych liczbowych:**

$$\alpha = 60^{\circ}; \quad v_0 = 500; \quad m = 1; \quad f = 0.0005; \quad g = 9.81$$

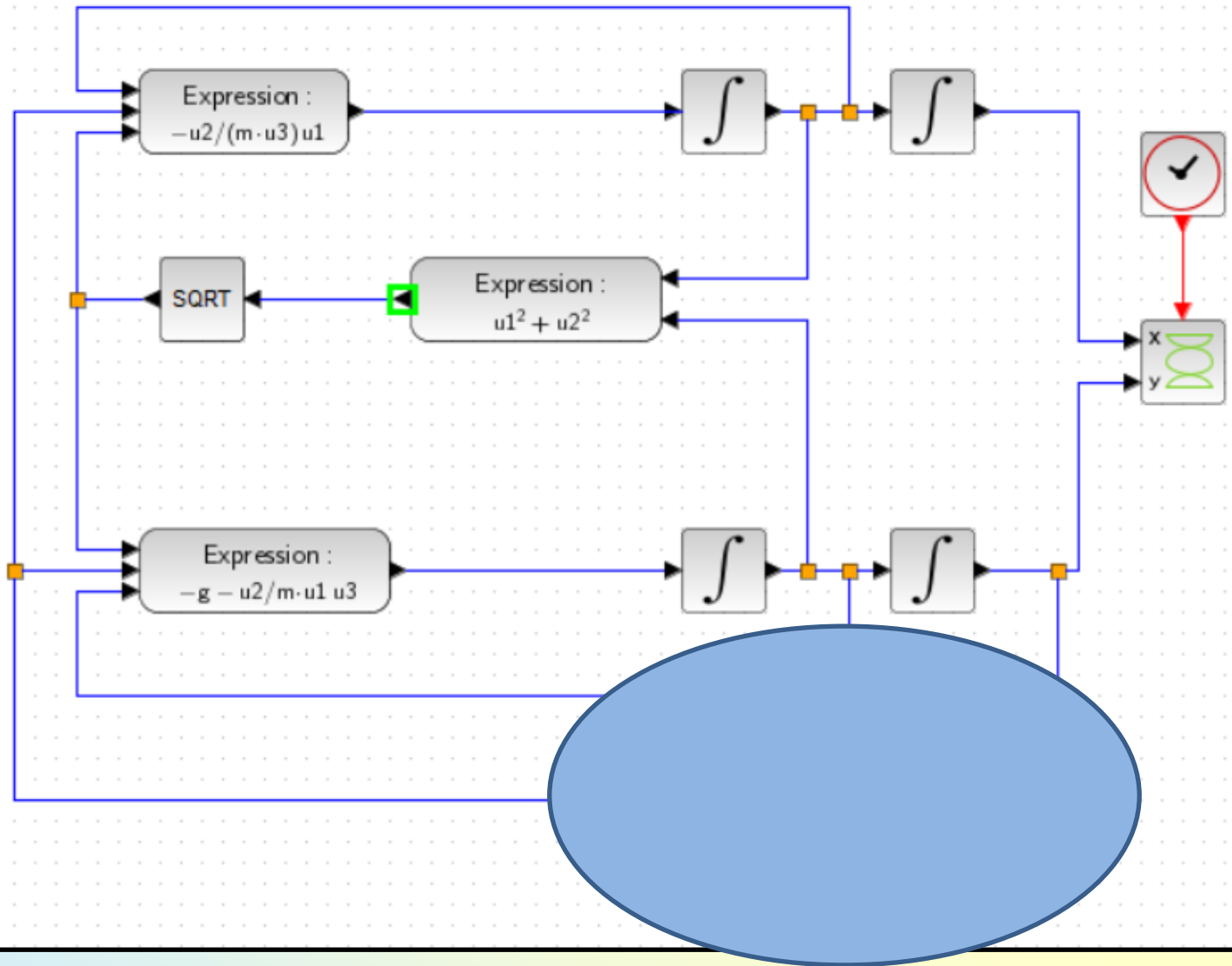
*czas obliczeń 43.00 [s]; a  $Y = 0$  dla 43.45 [s]*

Gdyby założyć, że pocisk po osiągnięciu wysokości  $Y=0$  zanurza się np. w wodzie, dla której współczynnik  $f$  związany z oporem w wodzie rośnie 10 razy, krzywa balistyczna wyglądałaby tak:

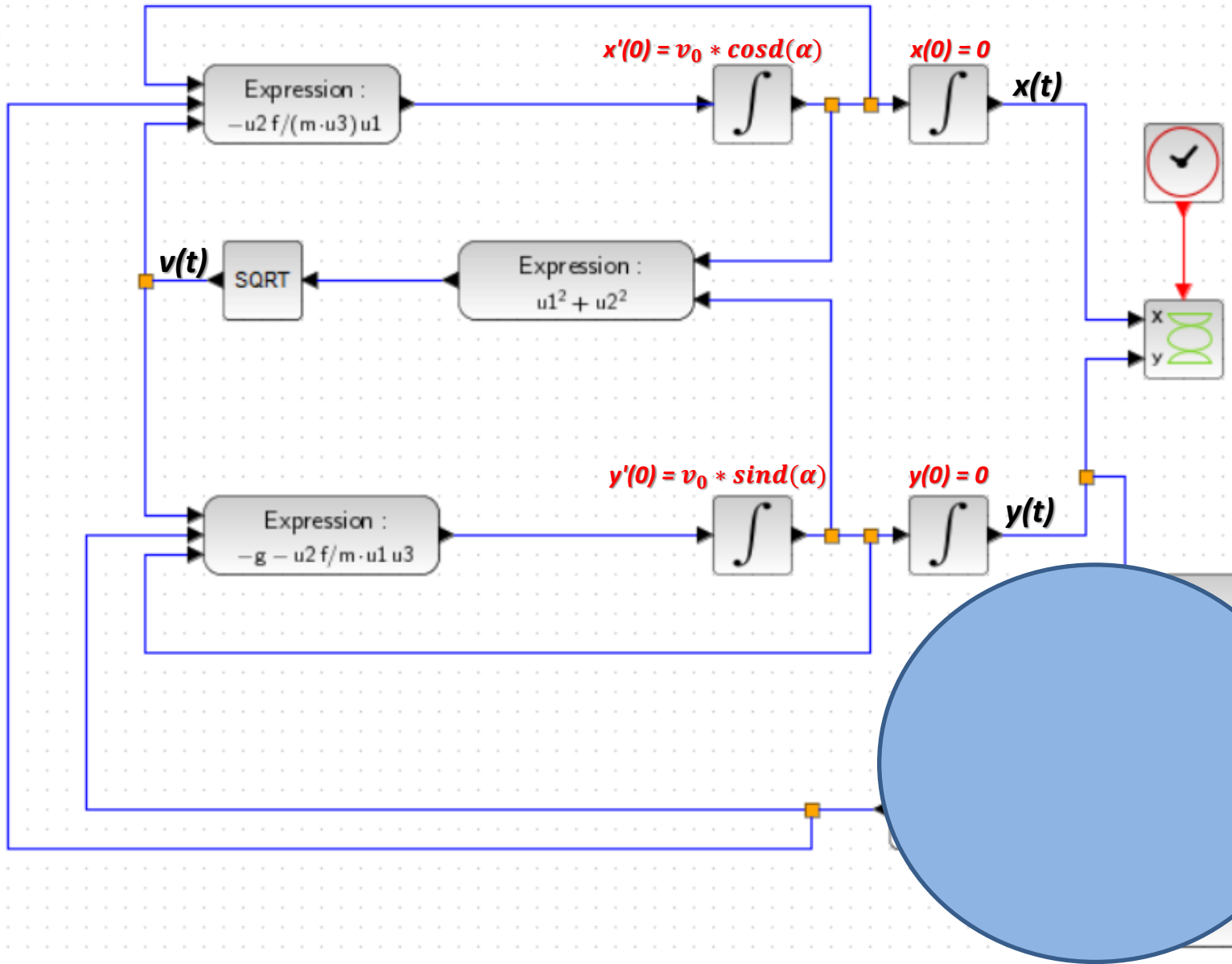


*czas obliczeń 60.00 [s]*

ver.1



ver.2



***Koniec w. 2***