



#### Our offerings at a glance:

#### WinMOD Products

- System Software
- System Configurations
- Add-ons
- Assistances
- OCA Libraries

#### System Training

• Customized training for the WinMOD System Software, as well as the WinMOD Configurations needed for the coupling to your automation system.

#### Start Ups for the System Implementation

- Project Counseling
- Project Monitoring
- WinMOD Libraries
- Work Flow Optimization/Assistance

#### WinMOD Libraries

- Company-specific Libraries
- Technology-specific Libraries

... and more than 10 years worth of experience in Virtual Commissioning.

#### Your way to us



Mewes & Partner GmbH Neuendorfstr. 15 16761 Hennigsdorf

Tel.: + 49(0)3302 2097-0 Fax: + 49(0)3302 2097-111

www.winmod.de winmod@mewes-partner.de **Center for Automation Real-Time Simulation**  **Center for Automation** 

Simulation

**Real-Time** 

# WinMOD®

Fomula # Reststrecke s rest := ABS

# Breasberei s breas :=

# in Breasber in\_s\_breas :=

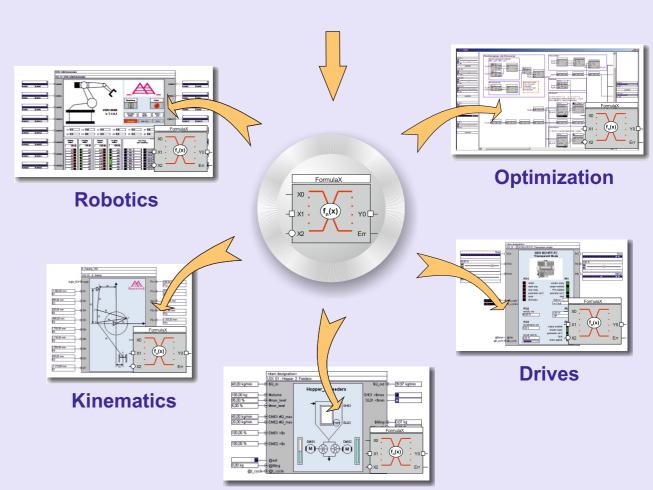
{ Der Konstan von +/- der

Der Konstan der aktuell 

OK

## **FormulaX**

 $s(t) = \frac{1}{2}a_0(t-t_1)^2 + v_1(t-t_1) + s_1 \ k(T) = k_0 \exp\left(-\frac{E_A}{RT}\right)_{M_{eff}} =$  $F = m \cdot a \qquad y = \begin{cases} y_p, x_a - x_s > e, \\ y_n, x_a - x_s < e, \\ a = m \end{cases}$  $\frac{s}{t} \quad f(x) = \begin{cases} -x, x < 0\\ x + 1, x \ge 0 \end{cases} \quad \rho_K =$  $= \frac{F_{G1} \cdot \rho_2 - F_{G2} \cdot \rho_1}{F_{G1} - F_{G2}} p = \frac{\Delta W}{\Lambda_*} b = \min(b_1, b_2, ..., b_n)$  $= c_A^0 \frac{k_1}{k_1 + k_2} \left( \frac{k_2}{k_1} - e^{-(k_1 + k_2)t} \right) \qquad Q = \lambda A \frac{T_1 - T_2}{\delta} t \qquad \beta = \arccos(\frac{a^2 + c^2 - b^2}{2ac}) \quad c_A = c_A^0 \frac{k_1}{k_1} + \frac{c_A^2 - b^2}{2ac} = \frac{1}{2} C_A + \frac{1}{2}$  $((A \leftrightarrow B) \land \neg C \quad p = \rho \cdot g \cdot h + p_0 \quad R = \frac{U}{I} \quad C = A \leftrightarrow B$  $b_n$ )  $P_{A,max} = m(a + g(\mu' \cdot \cos \alpha + \sin \alpha))v$   $x_{1,2} = \frac{p}{2} \pm \frac{p^2}{2} - q \ b = \min(a + g(\mu' \cdot \cos \alpha + \sin \alpha))v$  $= v_0(t - t_1) + s_1$   $c = \sqrt{d^2 + (a + b)^2 - 2d(a + b)\cos\beta}$   $C = A \wedge \neg B \ s(t) = v_0(t - t_1) + s_1$  $s(t) = \frac{1}{2}a_0(t - t_1)^2 + v_1(t - t_1) + s_1 k(T) = k$  $F = m \cdot a \quad y = \begin{cases} y_p, x_a - x_s > e, \\ y_n, x_a - x_s < e, \\ a = \max(a) \end{cases}$  $= \frac{s}{t} \quad f(x) = \begin{cases} -x, x < 0 \\ x + 1, x \ge 0 \end{cases} \quad \rho_K =$  $\frac{F_{1} + P_{2}}{F_{G1} - F_{G2}} \frac{F_{G2}}{P} = \frac{\Delta W}{h} b = \min(b_{1}, b_{2}, \dots, b_{n}) v = \frac{3}{t}$  $-e^{-(k_1+k_2)t}$ )  $Q = \lambda A \frac{T_1 - T_2}{\delta} t$   $\beta = \arccos(\frac{a^2 + c^2 - b^2}{2})$   $c_A = c_A^0 \frac{1}{4} \frac$  $P \land \neg Q \rightarrow ((A \leftrightarrow B) \land \neg C \quad p = p \cdot g \cdot h + p_0 \quad R = C = A \leftrightarrow B$  $n(a + g(\mu' \cdot \cos \alpha + \sin \alpha))v \quad x_{1,2} = \frac{p}{2} \pm \left| \frac{p^2}{2} - q \right| b = \min(b_1, b_2)$  $C = A \wedge \neg B \ s(t) = v_0(t - t_1) + s_1$   $c = \sqrt{d^2 + (a + b)^2 - 2d(a + b)\cos \beta}$ 





#### **Process**

$\boxed{\frac{1}{2}\sum_{m}^{m}M_{z}^{2}\cdot\Delta t_{z}} s(t) = \frac{1}{2}a_{0}(t-t_{1})^{2}$	$ \begin{array}{l} + v_1(t-t_1) + s_1 \ k(T) = k_0 \exp\left(-\frac{E_A}{RT}\right)_{M_{eff}} = \sqrt{\frac{1}{2}\sum_{x=1}^m M_x^2 \cdot \Delta t_x} \ s(t) = \frac{1}{2}a_0(t-t_1)^2 + v_1(t-t_1)^2 + v_2(t-t_1)^2 + v_2(t-t_2)^2 + v_2(t-t$
F_02 - (FormulaX)	0, else $v = \frac{s}{r} f(x) = \begin{cases} -x, x < 0 \\ 0, e \end{cases} = \frac{F_{G1} \cdot \rho_2 - F_{G2} \cdot \rho_1}{0} $ AW
puts View Content Licenses Position Lay	$(b_1, b_2, \dots, b_n)$ $t \to \infty$ $(x + 1, x \ge 0 + k$ $F_{G1} - F_{G2}$ $P = \frac{\Delta w}{\Delta t}$ $b = \min(b_1, b_2, \dots, b_n)$
e S(s_ist - s_soll );	$\frac{a^2 + c^2 - b^2}{2ac})  c_A = c_A^0 \frac{k_1}{k_1 + k_2} \left(\frac{k_2}{k_1} - e^{-(k_1 + k_2)t}\right) \qquad Q = \lambda A \frac{T_1 - T_2}{\delta} t \qquad \beta = \arccos(\frac{a^2 + c^2}{2ac}) \frac{a^2 + c^2}{\delta} t = \frac{1}{\delta} \frac{a^2 + c^2}{\delta} t = $
ch ABS(1/2 * SQR(v_act))/ ABS(v_act * t_cycle) - x_breas;	$ \begin{array}{c} \leftrightarrow B \\ p \\ \pm \end{array} \begin{bmatrix} E = P \land \neg Q \rightarrow ((A \leftrightarrow B) \land \neg C  p = \rho \cdot g \cdot h + p_0  R = \frac{U}{I}  C = A \leftrightarrow B \\ p \\ \pm \end{array} \begin{bmatrix} p^2 \\ 2 \pm \end{array} \begin{bmatrix} p^2 \\ -q  b = \min(b_1, b_2, \dots, b_n)  P_{A,max} = m(a + g(\mu' \cdot \cos \alpha + \sin \alpha))\nu  x_{1,2} = \frac{p}{2} \pm \end{bmatrix} \begin{bmatrix} p \\ p \\ \pm \end{array} \begin{bmatrix} p \\ 2 \end{bmatrix} \begin{bmatrix} p^2 \\ p \\ p \end{bmatrix} \begin{bmatrix} p^2 \\ p \\ p \\ p \end{bmatrix} \begin{bmatrix} p \\ p$
reich * (s_brems > s_rest);	$C = A \wedge \neg B  s(t) = v_0(t - t_1) + s_1  c = \sqrt{d^2 + (a + b)^2 - 2d(a + b)\cos\beta} \qquad C = A$
ntfahrbereich ungibt den Bre r Strecke, die in einem Zykl	$ \begin{array}{l} \sum\limits_{a=1}^{m} (t-t_1) + s_1 \ k(T) = k_0 \exp\left(-\frac{E_A}{RT}\right)_{M_{eff}} = \int \\ \sum\limits_{a=1}^{m} M_2^2 \cdot \Delta t_2 \ s(t) = \frac{1}{2} a_0 (t-t_1)^2 + v_1 (t-t_1)^2 + v_2 (t-t_1)^2 + v$
ntfahrbereich muss also dyna: len Geschwindigkeit berechne	$ \begin{array}{c} x_a - x_s > e, \\ x_e - x_e < e, \\ a = \max(a_1, a_2, \dots a_n) \end{array} \\ \begin{array}{c} M_{eff} = \\ \sum_{z=1}^{2} \sum_{z=1}^{2} M_z^2 \cdot \Delta t_z \\ F = m \cdot a \\ y = y_e, x_e - x_e \end{array} $
,	0, else $y = \frac{s}{f(x)} - \int -x, x < 0$ $y = \frac{F_{G1}, \rho_2 - F_{G2}, \rho_1}{0, e}$
L.: 1, C.: 1 Edit Formula	$(b_1, b_2, \dots, b_n)$ $v = t$ $f(x) = (x + 1, x \ge 0)$ $P_K = F_{G1} - F_{G2}$ $P = \frac{\Delta W}{\Delta t}$ $b = \min(b_1, b_2, \dots, b_n)$
Abbrechen Demeinnen Hife	$\frac{t^2 + c^2 - b^2}{2ac})  c_A = c_A^0 \frac{k_1}{k_1 + k_2} \left( \frac{k_2}{k_1} - e^{-(k_1 + k_2)t} \right) \qquad Q = \lambda A \frac{T_1 - T_2}{\delta} t \qquad \beta = \arccos(\frac{a^2 + c^2}{2ac}) dt = \frac{1}{2} \frac{1}{\delta} dt \qquad \beta = \arccos(\frac{a^2 + c^2}{2ac}) dt = \frac{1}{\delta} \frac{1}{\delta} dt = \frac{1}{\delta} \frac$

 $E = P \land \neg Q \to ((A \leftrightarrow B) \land \neg C \quad p = \rho \cdot g \cdot h + p_0 \quad R = \frac{U}{I} \quad C = A \leftrightarrow B$ 

 $\left|\frac{p^2}{2} - q\right| b = \min(b_1, b_2, ..., b_n) P_{A,max} = m(a + g(\mu' \cdot \cos \alpha + \sin \alpha))v x_{1,2} = \frac{p}{2} \pm \left|\frac{p}{2}\right| \frac{p}{2}$ 

 $A \wedge \neg B \quad s(t) = v_0(t - t_1) + s_1 \quad c = \sqrt{d^2 + (a + b)^2 - 2d(a + b)\cos^2}$ 





#### Engineering with WinMOD FormulaX

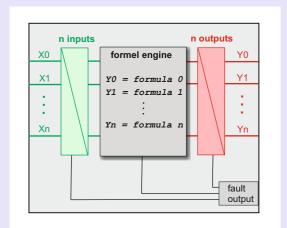


### **Use of WinMOD FormulaX**

#### What is WinMOD FormulaX?

WinMOD FormulaX is an Add-on for the realization of complex mathematical and physical formulas in the WinMOD System Software.

WinMOD FormulaX simplifies the engineering and optimizes the performance by the utilization of:



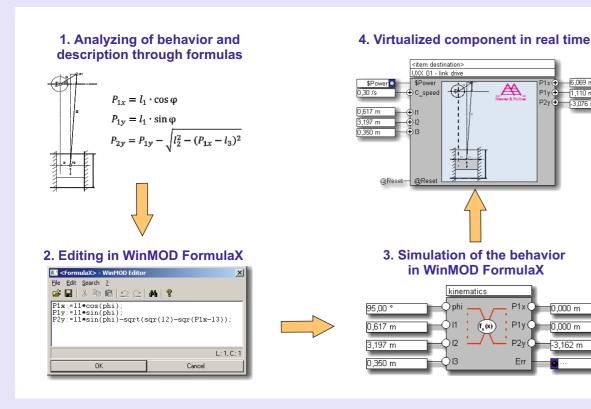
- · complex formulas with analog and binary operations
- compact notation in a WinMOD FormulaX element
- · clearly structured display of complex formula systems

The number of formulas and outputs in the runtime and editing mode is subject to the licensing.

WinMOD FormulaX

#### The Engineering with WinMOD FormulaX

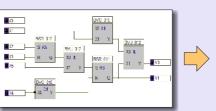
The behavior to be simulated is analyzed and relations of analog and binary state variables and process functions are described by formulas. The formula style corresponds with the general known rules of formula notations. Formulas can be chosen easily from collections of mathematical formulas and transfered to WinMOD.

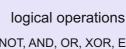


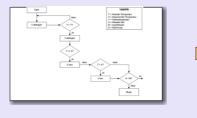
#### What is the Benefit of WinMOD FormulaX?

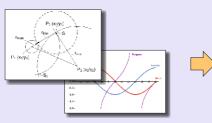
- extension of the WinMOD System Platform for the utilization of symbolic formula language
- higher simulation depth
- creation of customized libraries with WinMOD FormulaX
- integration of WinMOD FormulaX into the well-known WinMOD Components

#### Functions of WinMOD FormulaX









...and WinMOD specific: Identifier conversion, system variables, comments

#### OCA Libraries with WinMOD FormulaX

Basic libraries for WinMOD FormulaX developed by Mewes & Partner are provided free of charge for users.

Outlook on further development of WinMOD FormulaX libraries:

- libraries for physical formulas
- libraries for mechanical modeling
- libraries for motion mathematics
- libraries for pipe systems

#### Licensed Libraries with WinMOD FormulaX

On request, Mewes & Partner can develop costumer-specific libraries, which can be protected by WinMOD content licensing.



