

0 Několik konvencí

- Studujeme starší obrázky.
- Body vyznačujeme pomocí `odot(0.4mm,pozice)`.
- Úhly kreslíme se základním poloměrem 8 mm.
- Šrafovujeme s roztečí 1 mm.




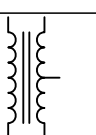


1 Základní balíky




fks.mp

kód	výsledek	použití
<code>draw (0,0)--(u,0) dashed cerch</code>		čerchovaná čára
<code>draw (0,0)--(u,0) dashed cark</code>		čárkovaná čára
<code>draw (0,0)--(u,0) wp5</code>		použije 5x silnější pero
<code>chpen5</code>		změní pero na 5x silnější
<code>uhel(A,B,C,rad)</code>		nakreslí úhel $\angle ABC$ s obloučkem poloměru 8 mm. !TODO! kreslit kruhový nikoli bezier.
<code>uhelR(A,B,rad)</code>		nakreslí pravý úhel $\angle ABC$ s obloučkem poloměru 8 mm. Typicky se dělá menší než 8 mm, zde 5 mm.
<code>axis((0,0),.2u,u,.1u,.5u)</code>		osový kříž se středem v (0,0), dále rozsahy na osách
<code>srafov((0,0)--(u,0)--(0,.5u)--cycle,dir45,1mm)</code>		šrafování, směr měníme dle potřeby, rozteč držíme na 1 mm
<code>drawearrow</code>		šipka s prázdnou hlavou
<code>drawblearrow</code>		dvojitá šipka s prázdnou hlavou
<code>drawvarrow</code>		šipka s "V" hlavou
<code>drawdblvarrow</code>		dvojitá šipka s "V" hlavou
<code>pruzina((0,0),(u,0),.2u,4)</code>		pružinka se šířkou 0.2u a čtyřmi závity
<code>odot(.4mm,(0,0))</code>		označení bodu, používá se 0.4mm
<code>kulicka(5mm,(0,0))</code>		kulička
<code>kotaDPK.ulft(z0,z1,0.5u,btex \$1\$ etex,0.5,0mm)</code>		kóta mezi body z0 a z1 odsazená o 0.5u popisek <code>btex \$1\$ etex</code> umístěný vlevo nahoru, posunutý na 50% k z1, posunutý od dvojšípky o 0mm.
<code>kotaDP.dir(poc,kon,lab,poz)</code>		<code>kotaDPK.dir(poc,kon,0.5u,lab,poz,0mm)</code>
<code>kotaD.dir(poc,kon,lab)</code>		<code>kotaDP.dir(poc,kon,lab,0.5)</code>
<code>kota(poc,kon,lab)</code>		<code>kotaD.dir(poc,kon,lab)</code> , kde <code>dir</code> se nastavuje podle směru dvojšípky automaticky.
!TODO! Zbylé kombinace		
<code>dist(a,b)</code>		
<code>provaz(size,path)</code>		Nakreslí lanko šířky <code>size</code> po trajektorii <code>path</code>
<code>provazVR(size,path)</code>		To samé, ale kreslí provaz, tj. má vroubky. Jednotlivé provazy jdou na sebe navazovat, při křížení cesty se může chovat nestandardně, mezní křivost je 1mm pro šířku 1 (platí přímá úměra).
<i>Další příkazy pocházejí z fks-label.mp</i>		
<code>olabel.poz(label,point)</code>		standardní label, ale vykreslí též bod pomocí <code>odot</code> , který označuje
<code>flabel.poz(label,point)</code>		standardní label, ale vybělí kresbu pod sebou
<code>oflabel.poz(label,point)</code>		standardní label, superpozice výše uvedených
<i>Další příkazy pocházejí z mth-function.mp</i>		
<code>sqr</code>		druhá mocnina
<code>log</code>		desítkový logaritmus
<code>ln</code>		přirozený logaritmus

exp		exponenciála
inv		převrácená hodnota
pow		umocnění; (základ, exponent)
sind		viz plain.mp
cosd		viz plain.mp
tand		tangens ve stupních
cotd		kotangens ve stupních
sin		sinus
cos		kosinus
tan		tangens
cot		kotangens
sinh		hyperbolický sinus
cosh		hyperbolic kosinus
tanh		hyperbolický tangens
coth		hyperbolický kotangens
arcsind		inversní sinus ve stupních
arccosd		inversní kosinus ve stupních
arctand		inversní tangens ve stupních
arccotd		inversní kotangens ve stupních
arcsin		inversní sinus
arccos		inversní kosinus
arctan		inversní tangens
arccot		inversní kotangens
argsinh		inversní sinus hyperbolický
argcosh		inversní kosinus hyperbolický
argtanh		!TODO! inversní tangens hyperbolický
argcoth		!TODO! inversní kotangens hyperbolický

fkscirc.mp

kód	výsledek	použití
fks_center:=true		přesun referenčního bodu součástky z konce nožičky na optický střed. (Hotové pouze u níže dokumentovaných součástek.)
svorky		
junction(z0)		puntík na spojení vodičů.
<i>Další příkazy pocházejí z makecirc-fks.mp</i>		
wire(z1,z2,type)		spojí dva body z1 a z2 vodičem, type $\in \{\text{nsq pro přímé spojení, udsq pro spoj začínající vodorovně, rlsq pro spoj začínající svisle}\}$
wireU(z1,z2,dist,type)		analogické, ale dělá U vodič vysunutý o dist. (Netestováno.)
inductor.La(z0,Up,45,name,val)		cívka, na pozici z0, otočená o 45°, Up, Down je orientace kopečků. Dále definuje proměnné L.AA.l a L.AA.r se souřadnicemi vývodů.
capacitor.Ca(z0,type,ang,name,val)		kondenzátor, dtto. volby: normal, electrolytic, variable a variant. Piny C.Ca.r, C.Ca.l.
motor.Ma(z0,ang,name,val)		motor, dtto. Piny M.Ma.D, M.Ma.B
generator.Ga(...)		generátor, dtto. Piny G.Ga.B a G.Ga.D.
transformer.Ta(z0,type,ang)		trafo, type $\in \{\text{normal normální, mid s dvěma sekundáry, Fe takové veliké, auto špatně definové piny}\}$. Piny tr.Ta.pi, tr.Ta.ps, tr.Ta.si, tr.Ta.ss a tr.Ta.m.
source.Sa(z0,type,angle,name,val)		zdroj, dtto. Volby AC, DC, I a V. Piny: G.Ga.p a G.Ga.n.
resistor		Nepoužívat — americký vzor, nahrazen impedance.
diode.Da(z0,type,ang,pin,name,val)		dioda, typ normal, zener a LED. Pin pinA nebo pinK otáčí diodu o 180°. Piny: D.Da.K a D.Da.A.
transistor		!TODO!

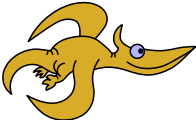
meains		!TODO!
ground		!TODO!
junction.ac(z0,"")(top)		puntík na spojení vodičů v bodě z0. !TODO! ostatní volby.
impedance.Ra(z2,45,name,val)		odpor s popisem a velikostí v bodě z2 otočený o 45°. Současně se vytvoří proměnné Ra.l a Ra.p se souřadnicemi konců vývodů. Nuté pro napojování vodičů. (verb bug)
lamp		!TODO!
switch.Sa(z0,N0,angle,name,val)		spínač, N0, NC pro otevřený/uzavřený; Piny: st.Sa.l, st.Sa.r.
battery.Ba(z0,0,name,val)		baterie. Piny B.Ba.p, B.Ba.n.
current		!TODO!
imesh		!TODO!
rheostat		!TODO!

fks-3d.mp

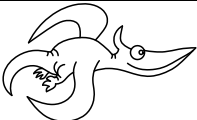
kód	výsledek	použití
VRP_tD		nastavení volného rovnoběžného promítání, volá se na začátku.
AXON_tD		nastavení axonometrického promítání, volá se na začátku.
PzP_tD		nastavení pohledu z prava (zobrazuje se rovina yz , osa x je opačná k ose y), volá se na začátku.
tD0=(2u,3u,1u)		3D proměnná analogická k z .
dD?		pair s projektovanou hodnotou tD?.
proj(x)		vrací pair s projekcí parametru
norm_tD?		norma tD?
norm(x)		vrací velikost parametru
!part_tD?		vrací !-ovou složku tD?; ! psán malým písmenem
!part_tD?(x)		vrací !-ovou složku parametru; ! psán malým písmenem
x dotprod_tD y		vrací skalární součin x a y
getTr(tD_o,tD_x,tD_y)		vrací transformaci z roviny definované body tD_o, tD_x, tD_y do projektivní roviny. Vhodné pro kreslení křivek jež jsou ve 3D na nějaké rovině. tD_o je počátek souřadnic na projektované rovině, tD_x směr osy x a tD_y další bod v projektované rovině (pomocí G-S OG se ortogonalizuje k ose x).
shifted_dD(x)		transformace, jenž posouvá projektovaný bod o 3D souřadnici x .
x SpRotate!_tD phi		otočí bod x o úhel ϕ kolem osy ! (psáno velkým písmenem)
rotate_tD(x,nP,nK,phi)		rotace x okolo normaloveho vektoru $nP \rightarrow nK$ o uhel ϕ


2 Loga – grafika

fks-logo-color.mp

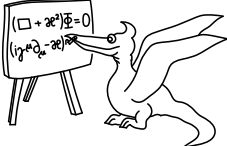
kód	výsledek	použití
mpost fks-logo-color.mp		barevný pták fykosák

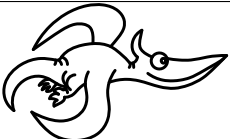
fks-logo.mp

kód	výsledek	použití
mpost fks-logo.mp		fks-logo.1

		fks-logo.2
		fks-logo.3
		fks-logo.4
		fks-logo.5
		fks-logo.6
		fks-logo.7
		fks-logo.8
		fks-logo.9
		fks-logo.10
		fks-logo.11
		fks-logo.12

fks-tricko.mp

kód	výsledek	použití
mpost fks-tricko.mp		fks-tricko.1

		fks-tricko.2
--	--	--------------

3 Nepoužívané balíky

fks-znacky.mp

kód	výsledek	použití
Stará makra pro kreslení el. obvodů.		

cary.mp

kód	výsledek	použití
drawdots(size) path	○ ○	kolečko s otvorem v každém řídicím bodu křivky
drawhlines(size) path	— —	vodorovná čárka, dtto.
drawvlines(size) path		svislá čárka, dtto.
drawctverce(size) path	□ □	čtverce, dtto.
drawtrojuhelniky(size) path	△ △	trojúhelníky, dtto.
drawkosoctverce(size) path	◇ ◇	kosočtverce, dtto.