



Řídicí systémy

CNC8X9 - DUAL

Výroba a servis

MEFI, s.r.o.
Peroutkova 37
150 00 PRAHA 5

tel: 251 045 113
fax: 251 045 112
e-mail: mefi@mefi.cz
[http: //www.mefi.cz](http://www.mefi.cz)



Návod k programování

Obsah

1. ÚVOD DO PROGRAMOVÁNÍ	1-1
1.1. Základní pojmy	1-1
1.2. Kód vstupních informací	1-2
2. STAVBA PARTPROGRAMU.....	2-1
2.1. Slovo partprogramu	2-1
2.1.1. Stavba slova	2-1
2.1.2. Psaní obsahu čísla	2-1
2.1.3. Typy slov	2-2
2.2. Blok partprogramu	2-2
2.2.1. Další pokyny k sestavení programového bloku	2-6
2.3. Stavba partprogramů, podprogramů, makrocyclů a pevných cyklů.....	2-6
2.3.1. Začátek partprogramu	2-7
2.3.2. Konec partprogramu	2-7
2.3.3. Podprogram (PPRG)	2-7
2.3.4. Příklad partprogramu s podprogramy	2-8
2.3.5. Makrocyclus (MRC)	2-9
2.3.6. Skoky v partprogramu, makrocyclu	2-10
3. PROGRAMOVÁNÍ SOUŘADNIC	3-1
3.1. Souřadnicový systém - označení souřadnic	3-1
3.2. Vlastnosti souvislého řízení souřadnic	3-1
3.3. Zadání dráhy	3-2
3.4. Nulové body stroje a programu	3-3
3.4.1. Definice nulových bodů stroje (NBS).....	3-3
3.4.2. Referenční bod stroje	3-4
3.4.3. Nulový bod programu	3-4
3.4.4. Přesunutí nulových bodů (skupina G5).....	3-4
3.4.5. Plnění tabulky posunutí počátků	3-5
3.4.6. Plnění tabulky posunutí z partprogramu funkcí G92 a G93	3-5
3.4.7. Plnění tabulky posunutí edicí souboru TAB0.POS	3-7
3.4.8. Plnění tabulky posunutí interaktivním zadáním	3-7
3.4.9. Volání posunů nulových bodů	3-7
4. DRUHY POHYBU	4-1
4.1. Stavění souřadnic - funkce G00	4-1
4.2. Lineární interpolace - funkce G01	4-1
4.3. Kruhová interpolace - funkce G02, G03	4-2
5. ŘEZÁNÍ ZÁVITŮ NOŽEM	5-1
5.1. Řezání závitu bez výběhu	5-1
5.2. Programování závitu s výběhem	5-2
5.2.1. Úhel výjezdu ze závitu	5-2
5.3. Závitování s vjezdem a výjezdem (výběhem).....	5-3
5.4. Závitování s velkým stoupáním	5-5
6. ZADÁNÍ POSUVU	6-1
6.1. Konstantní řezná rychlost (KŘR) G96 a G97	6-2
7. KOREKCE NÁSTROJE.....	7-1
7.1. Soubor TAB0.KOR a tabulka korekcí v paměti	7-1
7.2. Poloměrové korekce s ekvidistantou.....	7-3
7.2.1. Řazení poloměrové korekce a průběh korekce	7-5
7.2.2. Vyřazení poloměrové korekce	7-7
7.2.3. Limitní úhel pro vkládání oblouků	7-8
7.3. Poloměrové korekce u soustruhů	7-9
7.4. Indikace u soustruhu při použití poloměrové korekce	7-11
7.5. Řízení rychlosti při poloměrových korekcích	7-13

7.6. Test spojitosti pro poloměrové korekce	7-14
7.7. Zpřesňování kruhové interpolace	7-15
7.8. Volba korekčních a interpolačních rovin	7-16
7.9. Kontrola a optimalizace ekvidistanty	7-17
7.10. Délkové korekce	7-19
7.10.1. Korekce na délku nástroje	7-19
7.10.2. Délkové korekce – způsob ovládání A	7-19
7.10.3. Délkové korekce – způsob ovládání B	7-21
7.10.4. Délkové korekce – způsob ovládání C	7-21
8. PROGRAMOVÁNÍ TECHNOLOGICKÝCH FUNKCÍ	8-1
8.1. Časová prodleva	8-1
8.2. Otáčky vřetene	8-1
8.2.1. Stop vřetene v orientovaném bodě a řízení v polohové vazbě	8-2
8.3. Chlazení nástroje	8-2
8.4. Přerušení a konec partprogramu	8-2
8.5. Upnutí a uvolnění obrobku	8-3
8.6. Výměna nástroje a obrobku	8-3
8.7. Pomocné M-funkce skupiny M14	8-3
8.8. Pomocné M-funkce skupiny M10, M11, M12, M13	8-3
8.9. Číslo nástroje - funkce T	8-3
8.10. Pomocné funkce H, P	8-3
9. PARAMETRICKÉ PROGRAMOVÁNÍ - FUNKCE R	9-1
9.1. Deklarace parametrů	9-1
9.2. Vyvolávání parametrů	9-2
9.3. Desetinná tečka u deklarace parametrů	9-3
10. ŠROUBOVICE	10-1
11. ARITMETIKA PARAMETRŮ	11-1
11.1. Společné zásady	11-1
11.2. Řízení dvojité přesnosti	11-1
11.3. Struktura řídicích parametrů	11-2
11.4. Příklad využití aritmetiky parametrů	11-6
12. PROGRAMOVÁNÍ A DEFINICE PEVNÝCH CYKLŮ	12-1
12.1. Tvorba pevných cyklů	12-1
12.2. Příklad frézkových vrtacích pevných cyklů	12-2
12.3. Možnosti úprav pevných cyklů	12-7
12.4. Pevné cykly pro soustruhy	12-8
12.5. Příklad soustružnických pevných cyklů	12-9
12.5.1. Podélné hrubování - G81	12-9
12.5.2. Příklad použití pevného cyklu G81 pro podélné hrubování	12-11
12.5.3. Příčné hrubování - G82	12-12
12.5.4. Vrtání hlubokých děr - G83	12-14
12.5.5. Řezání závitu na válcové ploše - G84	12-16
12.5.6. Řezání závitu na válcové ploše s výjezdem - G85	12-21
12.5.7. Srážení hrany pod úhlem - G86	12-25
12.5.8. Řezání závitu na kulové ploše - G87	12-27
13. PŘEPOČET SOUŘADNIC	13-1
13.1. Společné zásady	13-1
13.2. Zrcadlení	13-2
13.3. Změna měřítka	13-4
13.4. Nezávislý posun	13-6
14. TRANSFORMACE SOUŘADNÉHO SYSTÉMU	14-1
14.1. Společné zásady	14-1
14.2. Principy rovinné transformace souřadnic	14-2
14.3. Principy prostorové transformace souřadnic	14-4
14.4. Řízení transformací z NC programu	14-10
14.4.1. Nastavení parametrů transformace z NC programu	14-10

14.4.2. Aktivace a deaktivace transformace z NC programu	14-11
14.5. Řízení transformací pomocí dialogového okna.....	14-12
14.6. Řízení transformace z PLC programu.....	14-14
14.7. Diagnostika transformace souřadnic	14-15

PŘÍLOHY

PŘÍLOHA A - PLYNULÁ NÁVAZNOST BLOKŮ

Příloha A1 - napojování bloků bez plynulé návaznosti.....	A-1
Příloha A2 - těsné napojování bloků.....	A-2
Příloha A3 - ruční řízení plynulé návaznosti.....	A-2
Příloha A4 - automatické rozpoznávání plynulé návaznosti	A-4
Příloha A5 - grafická diagnostika plynulé návaznosti	A-5
Příloha A6 - obálková rychlost s lineárním průběhem rychlosti.....	A-7
Příloha A6.1 - dynamické lineární řízení rychlosti s analýzou příštích bloků	A-7
Příloha A6.2 - přehled parametrů pro aktivaci a činnost obálkové rychlosti	A-8
Příloha A6.3 - kritérium času pro nárůst rychlosti, parametr R395.	A-10
Příloha A6.4 - kritérium přesnosti, parametr R384.....	A-12
Příloha A6.4 - dynamické kritérium, parametr R385.....	A-13
Příloha A7 - obálková rychlost s parabolickým průběhem rychlosti	A-16
Příloha A7.1 - parabolický průběh řízení rychlosti.....	A-16
Příloha A7.2 - dynamické parabolické řízení rychlosti s analýzou příštích bloků	A-17
Příloha A7.3 - přehled parametrů pro aktivaci a činnost obálkové rychlosti	A-18
Příloha A7.4 - kritérium času pro nárůst rychlosti, parametr R395	A-23

PŘÍLOHA B - PRIORITYNÍ BLOK.....

Příloha B1 - nastavení priorityního bloku.....	B-2
--	-----

1

1. ÚVOD DO PROGRAMOVÁNÍ

1.1 Základní pojmy

U číslicově řízených strojů se k přenosu informace o obráběném polotovaru či součásti používá kódovaného zápisu zvaného **partprogram** (program součásti).

U moderních systémů se sled těchto informací ukládá do zálohované paměti. U řídicích systémů MEFI CNC836/CNC846/CNC856 (dále jen CNC8x6) je kapacita zálohované paměti pro partprogramy řádově stovky kByte pro systém osazený pamětí CMOS (již se nedodává), desítky MByte pro systém osazený flash diskem a stovky MByte až jednotky GByte pro systémy osazené pevným diskem.

Kromě partprogramů jsou v zálohované paměti uloženy také tabulky korekcí, posunutí počátků a parametrů. Partprogramy (a jejich logické součásti t.j. podprogramy, makrocikly, pevné cykly) jsou v paměti uloženy ve formě, kterou může systém během řízení v reálném čase snadno zpracovávat a převádět na okamžité instrukce pro řízený stroj.

Partprogram, vytvořený technologem a uložený na vhodném médiu (disketa, sériový komunikační adaptér, disk nadřazeného počítače apod.), je pak periferním zařízením zaveden do zálohované paměti systému. Partprogram se dá vytvořit buď strojním zpracováním na počítači po zavedení základních geometrických a technologických údajů a parametrů obráběcího stroje (AUTOPROG, KOVOPROG, APT apod.) nebo ručně, výpočtem koncových bodů všech elementárních pohybů nástroje a doplněním příslušných technologických funkcí. Vytvořený partprogram se přenese do paměti systému (viz Návod k obsluze). Partprogram lze zapsat do paměti systému také přímo z ovládacího panelu.

Kódový zápis geometrie a technologie součástky tvořící partprogram je tvořen sledem přípustných znaků (tzv. adres). Tento sled znaků musí jednoznačně popisovat obráběcí postup na konkrétním stroji, musí být jednoznačně identifikovatelný jako celek a ve formě výpisu na tiskárně nebo obrazovce displeje musí být snadno srozumitelný a přehledný.

Při tvoření partprogramu je třeba vycházet z těchto údajů:

- a) geometrie stroje (souřadný systém, orientace os, nulové body)
- b) geometrie polotovaru (možnost kolize obrobek - nástroj během obrábění, umístění obrobku v souřadné soustavě stroje)
- c) geometrie nástroje (rozměry, tvar, korekce dráhy nástroje na tvar obrobku)
- d) geometrie výsledného obrobku (dána výkresem součástí)
- e) technologické a řezné podmínky (řezné rychlosti, síla třísky atd.)
- f) ostatní podmínky důležité pro činnost obráběcího stroje (chlazení, velikost posuvů, otáček, čísla nástrojů, korekcí atd.)

Výstavba partprogramu musí pak vyhovovat předepsané syntaxi zápisu, aby byla zaručena jednoznačnost vyjádření.

Partprogram je sestaven z řady jednotlivých elementárních operací, tzv. **bloků**. Blok se skládá z dílčích údajů,

tzv. **slov**. Každé slovo pak obsahuje (až na některé výjimky) **adresu**, udávající o jaký druh informace jde a **číselný údaj**, udávající rozměrovou hodnotu nebo kódové přiřazení k nějaké funkci či operaci. Jeden blok může být rozepsán na více řádků, další blok je určen pouze adresou N!

Příklad **bloku** partprogramu:

N20 G1 X10.355 Z625.50 F300 S150 T12 M0

Slova v tomto bloku jsou : N20, G1, X10.355, Z625.50, F300, S150, T12, M0

Adresami jsou .: N, G, X, Z, F, S, T, M

Číselné údaje jsou: 20, 1, 10.355, 625.50, 300, 150, 12, 0

Příklad rozepsání jednoho bloku (N10) do více řádků:

N10 R1=100.

R2=200.

R3=300.

N20 G54 G0 X0 Y0 Z0

1.2 Kód vstupních informací

Systém přijímá vstupní informace partprogramu v textovém tvaru t.j. bez parity. Případná diaktrická znaménka češtiny se mohou vyskytovat pouze v komentáři partprogramu. Pokud má být čeština v komentářích správně zobrazená, musí být v kódu kódu Kamenických. Adresy, t.j. znaky, jež systém zpracovává, jsou uvedeny v tabulce 1.



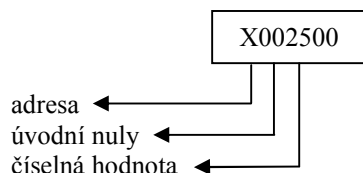
2. STAVBA PARTPROGRAMU

2.1 Slovo partprogramu

2.1.1 Stavba slova

Elementárním stavebním prvkem partprogramu je tzv. slovo (instrukce programu). Každé slovo sestává z písmene adresy a jedno nebo víceciferného čísla obsahu adresy s případným znaménkem a (nepovinnou) desetinnou tečkou.

Příklad:



2.1.2 Psaní obsahu čísla

- U zápisu hodnoty pomocných funkcí mohou být vynechány úvodní nuly před první významovou číslicí.
Např.: F 1.4 S 35 atd.
- U zápisu hodnot souřadnic bez desetinné čárky mohou být rovněž vynechány úvodní nuly před první významovou číslicí a údaj znamená dráhu v mm.
Např.: X 3620 = 3620 mm
- U zápisu hodnot souřadnic s desetinnou čárkou mohou být vynechány úvodní nuly před první platnou číslicí před desetinnou čárkou a za poslední platnou číslicí za desetinnou čárkou.
Např.:
X 36.12 = 36.12 mm
X 0.2 = 0.2 mm
X 384. = 384 mm
X .12 = 0.12 mm
- Znaménko "-" musí být uváděno, ale znaménko "+" nemusí být uváděno.
- Nese-li slovo nulovou informaci, musí být zapsána minimálně jedna nula.
- Mezi adresou a číselnou hodnotou slova může, ale nemusí být jedna, případně více mezer. Naopak **číselná hodnota včetně znaménka nesmí být přerušena mezerou**. V případě parametrického programování je číselná hodnota nahrazena parametrem (viz kapitola o parametrickém programování).

Např.:

X36.12	správně
X 36.12	správně (mezera za adresou)
X36. 12	chybně (mezera uprostřed čísla)
X- 36.12	chybně (mezera za znaménkem)

2.1.3 Typy slov

Podle jakosti informace, kterými jsou jednotlivá slova partprogramu nositeli, rozdělujeme slova a jejich adresy do dvou základních skupin, a to na rozměrové a bezrozměrové.

- rozměrová slova - vyjadřují kromě hodnoty také rozměr. Mohou být kladná i záporná, s desetinnou čárkou nebo bez. Patří sem adresy: A,B,C,X,Y,Z,U,V,W,I,J,K,F,R a adresa Q pro časovou prodlevu.
- bezrozměrová slova - vyjadřují pouze hodnotu. Patří sem adresy: D,G,M,S,T,P,H a adresa Q pro počet opakování.

Některá slova partprogramu se v systému pamatují trvale, tzn. že jejich platnost sahá do všech následujících bloků partprogramu, dokud nejsou přepsána jinou hodnotou téže adresy. Druhou skupinu slov tvoří slova, která mají platnost pouze v bloku, ve kterém byla programována.

2.2 Blok partprogramu

Jednotlivá slova se v partprogramu sestavují do programových bloků. Programový blok tvoří základní informační jednotku nesoucí údaj o geometrii a technologii obrábění. Formát bloku má proměnnou délku.

Maximální formát bloku má tvar:

Pozn.:

V závorce je uveden max. počet cifer dané adresy. Číslice před adresou G a M udává počet skupin.

N	(8)	11G	(2)	D	(2)
X	(+5.3)	Y	(+5.3)	Z	(+5.3)
4.	(+5.3)	5.	(+5.3)	6.	(+5.3)
I	(+5.3)	J	(+5.3)	K	(+5.3)
R	(+5.3) (+8)	&	(4)	F	(2..3) (5)
S	(4)	L	(4)	Q	(4)
H	(2)	P	(2)	T	(8)
10M	(2)	"KOMENTÁŘ" nebo LF			

Význam adres a jim příslušejících hodnot udává následující tabulka (slovník systému):

Adresa	Skupina	Hodnota	význam
%		1 až 999999	Číslo partprogramu
A		1 až 99999999	Číslo bloku partprogramu, podprogramu nebo makrocyclu
/		---	Vynechání bloku při režimu "/"
G	G0	00 01 02 03 30 33	Najíždění rychloposuvem Lineární interpolace Kruhová interpolace CW Kruhová interpolace CCW Nájezd do reference z programu Řezání závitů
	G1	17+ 18 19 14 15 16	Volba roviny XY (pouze pro korekce ... Volba roviny YZ ... na poloměr nástroje) Volba roviny ZX (G41 a G42) Volba roviny Z4 (viz R340, R341) Volba roviny Y4 Volba roviny 4X
	G2	05 06 07 08 09 10 24 23 98	Kopírování podle předlohy Kopírování z paměti Kopírování do paměti Vlečení os Šroubovice (spirála) Zadání stoupání pro spirálu Těsné napojování bloků Plynulá návaznost bloků Základní stav skupiny 2
	G3	40+ 41 42	Zrušení poloměrových korekcí Poloměrová korekce vlevo Poloměrová korekce vpravo
	G4	26 27 28 29	Aritmetické operace – 1. vzorec Aritmetické operace – 2. vzorce Aritmetické operace – 3. vzorce Aritmetické operace – 4. vzorce
	G5	53 54 55 56 57 58x 59x	Posunutí 0. Nulového bodu Posunutí 1. Nulového bodu Posunutí 2. Nulového bodu Posunutí 3. Nulového bodu Posunutí 4. Nulového bodu Posunutí 5. Nulového bodu Posunutí 6. Nulového bodu
	G6	94+ 95 96 97+	Posuv v mm/min bez KŘR Posuv v mm/ot bez KŘR Posuv v mm/min s KŘR Posuv v mm/ot s KŘR
	G7	70+ 71x 72x 73x 79	Konec makrocyclu nebo podprogramu Volání podprogramu-v bloku musí být současně programována adresa L Volání makrocyclu-v bloku musí být současně programována adresa L Skok na programový blok, jehož číslo udává hodnota u adresy L Vstup do makrocyclu nebo podprogramu Funkce musí být programována společně s adresou L

	G8	80+ 81 až 89	Zrušení pevného cyklu Pevné cykly podle normy ISO
	G9	90+ 91	Zadání absolutních hodnot souřadnic Zadání přírůstkových hodnot souřadnic
	G10	04x 92x 93x 74 75	Časová prodleva - velikost je zadána funkcí Q. Dosazení hodnot do tabulek počátků pro danou G-funkci ze skupiny G5. Dosazení hodnot do tabulek korekcí. Dosazení hodnot posunutí počátků pro danou G-funkci ze skupiny G5, při které dojde ke ztotožnění nulového bodu programu pro G-funkci ze skupiny G5 s bodem souřadnicového systému, který je vzdálen od skutečné polohy nástroje o hodnoty programovaných souřadnic v daném bloku. Průměrové programování Poloměrové programování
	G11	34 35 36 37 38 39 31	Zrušení všech přepočtů souřadnic. Zrcadlení, převádí směr ve zvolených osách. Změna měřítka. Nezávislý posun. První rovinná transformace souřadnic Druhá rovinná transformace souřadnic Prostorová transformace souřadnic
&		0000 až 2222	Řídící funkce určující přiřazení délkových korekcí do příslušných souřadnic a v příslušném směru.
D		1 až 99	Číslo položky v tabulce korekcí
X		+69999.999 -69999.999	Zadání dráhy v mm
Y		+69999.999 -69999.999	Zadání dráhy v mm
Z		+69999.999 -69999.999	Zadání dráhy v mm
4.		+69999.999 -69999.999	Zadání dráhy v mm
5.		+69999.999 -69999.999	Zadání dráhy v mm
6.		+69999.999 -69999.999	Zadání dráhy v mm
I		+69999.999 -69999.999	Vzdálenost středu kružnice ve směru první interpolované souřadnice
J		+69999.999 -69999.999	Vzdálenost středu kružnice ve směru druhé interpolované souřadnice
R0 až R95		+69999.999 -69999.999	Hodnota parametru přiřaditelného k jednotlivým souřadnicím nebo technologickým funkcím
Q		1 až 9999	Počet opakování podprogramu pro G71, makrocyklu pro G72 nebo skoků na programový blok pro G73. Velikost časové prodlevy po 10ms při zadání funkce G04

F		1 až 24 000 pro posuv v mm/min. 1 až 99 999 pro posuv v mm/ot.	Posuv v mm/min nebo v mm/ot
L		1 až 9999	Číslo volaného podprogramu nebo makrociklu při funkci G71, G72, G79. Číslo bloku, na nějž se provádí odskok při funkci G73.
H		1 až 99	Pomocná funkce
P		1 až 99	Pomocná funkce
S		1 až 9999	Otáčky vřetena
T		+ až 69999999	Číslo nástroje
M	M1	00 ^x 01 ^x 02 ^x 30 ^x	Programový stop Volitelný stop (STOP v režimu M01) Konec partprogramu se zpětným návratem na začátek partprogramu. Poznámka : nastaví M-funkce ve skupinách M2, M5, M6 a M8 na hodnoty M funkcí označených +. dtto viz M02
	M2	03 04 05 ⁺ 19(+)	Start vřetena "CW" Start vřetena "CCW" Stop vřetena Stop vřetena v orientovaném bodě
	M3	41 42 43 44 40 ⁺	Otáčky vřetena rozsah 1 Otáčky vřetena rozsah 2 Otáčky vřetena rozsah 3 Otáčky vřetena rozsah 4 Rozsah otáček vřetena je vypočten přímo z funkce S
	M4	36 ⁺ 37	Posuv programován přímo Posuv zmenšen 1 : 100
	M5	07 08 09 ⁺ 17	Zapnutí chlazení 2 Zapnutí chlazení 1 Vypnutí chlazení 1 a 2 Zapnutí chlazení 1 a 2
	M6	50 51 53 ⁺ 52	Zapnutí chlazení 3 Zapnutí chlazení 4 Vypnutí chlazení 3 a 4 Zapnutí chlazení 3 a 4
	M7	10 ^x 11 ^x	Upnutí obrobku Uvolnění obrobku
	M8	49 48 ⁺	Překlenutí ručního FEED OVERRIDE Zrušení překlenutí (zařazení) FEED OVERRIDE
	M9	06 ^x 60 ^x	Výměna nástroje Výměna obrobku
	M10	10 až 99 ⁺	Funkce dle strojní konstanty R56
	M11	10 až 99 ⁺	Funkce dle strojní konstanty R57
	M12	10 až 99 ⁺	Funkce dle strojní konstanty R58
	M13	10 až 99 ⁺	Funkce dle strojní konstanty R59
	M14	10 až 99 ⁺	Všechny pomocné funkce, které nebyly v ostatních skupinách uvedeny

*			Konec načítání partprogramu včetně jeho podprogramů
---	--	--	---

Vysvětlení značek použitých v tabulce:

x - Pomocná funkce, platí jen v bloku, kde byla uvedena.

+ - Číslo funkcí, které ruší uvedená čísla v příslušných skupinách a které nastanou dočasně ve skupinách M2, M5 a M6 při funkcích MOO, MO1 a trvale ve všech skupinách při funkcích MO2, M3O a po vynulování systému.

Pozn.: Eventuální další M-funkce, neuvedené v tomto seznamu, musí dodat tvůrce PLC programu pro konkrétní stroj.

2.2.1 Další pokyny k sestavení programového bloku

- Za adresu 4. nebo 5. je možno programovat písemný znak A,B,C,U,V,W (podle dohody s výrobcem stroje).
- Za adresu 6. je možno programovat písemný znak A,B,C, (podle dohody s výrobcem stroje).
- KOMENTÁŘ je řetězec libovolných znaků (kromě uvozovek) uzavřený mezi uvozovkami. Pokud je za komentářem uveden znak konce řádku (CR,LF), nemusí být koncové uvozovky uvedeny.

Příklad:

N10 X100 Y100 " TOTO JE KOMENTAR "

N20 X200 Y200 " TOTO JE KOMENTAR

" "

"

N30 X100 "TOTO JE KOMENTAR" Y200 Z300

N40 X100 "TOTO JE KOMENTAR Y200 Z300

Pozn.:

V bloku N40 se do komentáře zahrne i Y200 Z300, neboť nejsou uvedeny druhé uvozovky, takže se za konec komentáře považuje konec řádky!

- Každý programový blok musí začínat adresou N - (číslo bloku). Blok končí před následujícím znakem N (číslo následujícího bloku) nebo koncem souboru (může být uvedena *, která je ovšem povinná pouze při načítání po sériové lince). Ostatní pořadí slov v jednom bloku je libovolné. Doporučuje se však užívat standardního sledu slov (viz maximální formát bloku).
- Systém pracuje s proměnnou délkou bloku. To znamená, že v každém bloku může mít libovolný počet slov. (Do paměti se uloží i prázdný blok označený pouze číslem bloku, což však nemá praktický význam). Každá adresa (slovo) může být zapsána v jednom bloku pouze jednou, s výjimkou skupinových funkcí (M,G), u kterých může být zapsána jedna hodnota z každé skupiny a adresy R, která může být v bloku použita vícekrát (max. 95krát).
- Počet všech znaků programového bloku (včetně oddělovačů a komentáře) není omezen. Na řádce (mezi znaky CR,LF) však smí být maximálně 80 znaků.
- Místo hodnoty adresy je možné u všech adres kromě N zapisovat číslo parametru R - viz kapitola o parametrickém programování.
- Jako oddělovače mezi jednotlivými slovy bloku lze užít libovolný počet mezer. Mezery se mohou navíc užít i uvnitř slova, avšak pouze mezi adresou a číslem (nikdy ne uvnitř číselné hodnoty !).
- Znak "/" (lomítko) je jediná bezhodnotová adresa (číslice za ní následující není vyhodnocována). Při navolení režimu "/" (vynechání bloků) je blok, ve kterém je uvedeno "/", při zpracovávání partprogramu ignorován. Znak lomítka musí být uveden až za číslem bloku, kterého se týká. Pokud by byl uveden před znakem N, platil by pro předešlý blok.
- Je-li v bloku programovaná časová prodleva (funkcí G04) společně s některou funkcí ze skupiny G7, platí funkce Q přednostně pro časovou prodlevu. V tomto případě se počet podprogramů, makrocyklů případně skoků vykoná pouze jedenkrát (stejně jako při Q=1).

2.3 Stavba partprogramů, podprogramů, makrocyklů a pevných cyklů

Partprogram dané součásti se obvykle skládá z části vlastního partprogramu a z části tzv. podprogramů. Podprogramy se vztahují pouze k danému partprogramu, za jehož koncem musí být bezprostředně zapsány. Partprogram nemusí podprogramy obsahovat.

2.3.1 Začátek partprogramu

Partprogram musí začínat číslem partprogramu. Je uveden adresou % a maximálně šestidekádovým kladným nenulovým číslem bez desetinné čárky, které je identifikátorem partprogramu při automatickém chodu. Toto číslo se vypisuje při chodu partprogramu ve formátu listingu, jiný význam nemá. Toto číslo partprogramu nemusí souhlasit s názvem souboru, ve kterém je partprogram uchováván v zálohované paměti (na disku nebo v paměti CMOS) i když je vhodné aby název souboru s partprogramem i číslo partprogramu byly totožné. Pokud za číslem partprogramu není uveden žádný komentář, vypíše se ve formátu listingu kromě čísla partprogramu pro informaci i název souboru v hranatých závorkách. Pokud je za procentem uveden komentář, vypíše se tento komentář.

Příklad:

```
%1  
%123456 "KOMENTAR"  
%200
```

Pokud se během načítání partprogramu ze vstupního zařízení nenalezne znak %, znaky se ignorují, tzn. že před znakem % se může nacházet jakýkoli znak kromě %.

2.3.2 Konec partprogramu

Logický konec partprogramu je určen programováním funkce M02 nebo M30 v posledním bloku vlastního partprogramu. Za blokem s funkcí M02 nebo M30 mohou být v partprogramu programovány pouze bloky jednotlivých podprogramů, které se "volají" z těla partprogramu.

Fyzické ukončení celého partprogramu včetně eventuelních podprogramů musí být označeno znakem * (hvězdičkou).

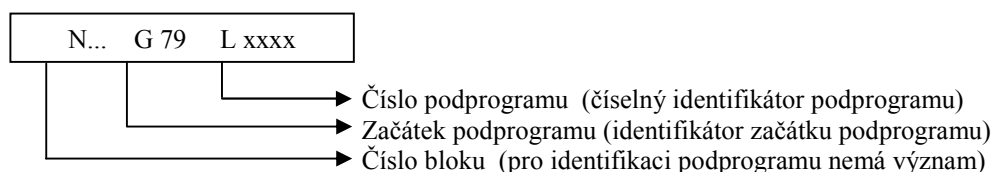
Pozn.:

Při zápisu partprogramu v editoru systému se hvězdička nezapisuje, systém si ji doplní automaticky. Hvězdičku je nutné do partprogramu zapsat prakticky pouze v případě, že se partprogram načítá ze sériové linky.

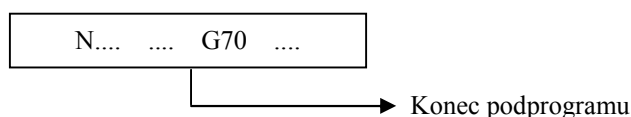
2.3.3 Podprogram (PPRG)

Podprogramem se rozumí určitá skupina programových bloků, které mají standardní úvodní a závěrečný blok. Podprogram logicky patří pouze k danému partprogramu a může být volán pouze z tohoto partprogramu.

Každý podprogram musí povinně začínat blokem, ve kterém je kromě čísla bloku uvedena pouze funkce G79 a adresa L s hodnotou, udávající číslo podprogramu. Číslo bloku nemá pro identifikaci podprogramu žádný význam. Tvar úvodního bloku podprogramu je:

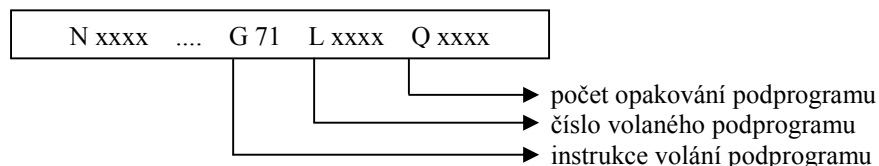


Podprogram musí být ukončen blokem, ve kterém je uvedena funkce G70. U posledního bloku podprogramu je povoleno programovat i jiné funkce, případně pohyb. Poslední blok podprogramu má tvar:



Při nahrávání partprogramu musí být příslušné podprogramy nahrány společně s partprogramem, ke kterému patří (ihned za blokem, ve kterém je programována funkce M02 nebo M30), nebo-li musí být ve stejném souboru.

Maximální počet podprogramů v partprogramu je 99. V příslušném partprogramu, který začíná znakem % a končí znakem * (vlastní partprogram s eventuálními podprogramy) nesmí být dva nebo více programových bloků označeno stejným číslem bloku (hodnota adresy N). Volání podprogramu je možno provést v kterémkoliv bloku vlastního partprogramu. Volání podprogramu má tvar :



Počet opakování podprogramu, programovaný funkcí Q nemusí být uveden. Potom se podprogram vykoná pouze jednou (stejně jako při programování Q1). Po skončení podprogramu zůstávají hodnoty technologických a pomocných funkcí ve stavech tak, jak je nastavil podprogram. Podprogram je možno editovat v témže rozsahu jako vlastní partprogram. V programovém bloku partprogramu, ze kterého je volán podprogram, je přípustné programovat i ostatní posuvové a technologické funkce. Vlastní odskok na zvolený podprogram se provede až v závěru bloku, t.j. po vykonání ostatních programovaných operací. Po provedení podprogramu se řízení vrátí na následující blok partprogramu, než ze kterého byl odskok proveden.

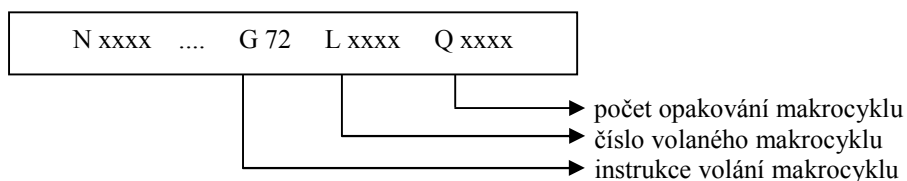
2.3.4 Příklad partprogramu s podprogramy:

```
%35          " ZACATEK PARTPROGRAMU CISLO 35 "
N10 X0 Y0 Z0 G54  " PRVNI BLOK PARTPROGRAMU "
N20 X200 G1 F300  " DRUHY BLOK PARTPROGRAMU "
....
....
N80 G71 L100 Q2   " VOLANI PODPROGRAMU CISLO 100 "
....          " PODPROGRAM BUDE VOLAN DVAKRAT "
....
N120 G71 L200     " VOLANI PODPROGRAMU CISLO 200 "
....
....
```

Makrocyklem se rozumí určitá skupina programových bloků, které tvoří partprogram pro typickou součást nebo část součásti vyráběnou na konkrétním stroji.

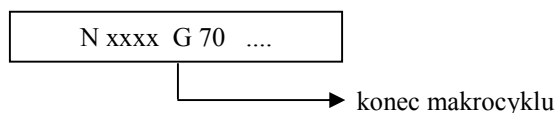
Makrocykly patří do tzv. skupiny knihovních partprogramů a jsou v systému uloženy jako samostatné soubory s povinným názvem Lxxxx, kde xxxx je číslo makrocyklu uvedené v souboru pod adresou L (G79 Lxxxx). Do systému nesmí být nahrány dva nebo více souborů s makrocykly označených stejným názvem.

Volání makrocyklu je možno provést z kteréholiv bloku libovolného partprogramu. Volání má tvar:



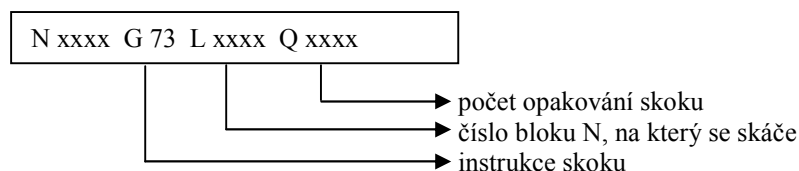
2-9

V programovém bloku partprogramu (t.j. i podprogramu) nebo makrocyklu, ze kterého je požadovaný jiný makrocyklus volán, je přípustné programovat i ostatní posuvové a technologické funkce. Vlastní odskok na volaný makrocyklus se provede obdobně jako u volání podprogramu až v závěru bloku. Každý makrocyklus musí být zakončen shodně jako podprogram:



2.3.6 Skoky v partprogramu, makrocyklu

Pomocí funkce G73 je možno programovat odskok na jiný programový blok. Blok, ve kterém programujeme odskok, má tvar:



Skoky jsou povoleny pouze na programové bloky v rámci tzv. programového celku, t.j. v rámci partprogramu, nebo podprogramu nebo makrocyklu ve směru vzad i vpřed (tj. ve směru k začátku nebo konci partprogramu, podprogramu nebo makrocyklu). Nelze tedy použít např. skoku z partprogramu do podprogramu.

Hodnota funkce Q vyjadřuje počet opakování skoku. Při Q (n) se úsek programu mezi příslušnými bloky vykoná (n+1) krát.

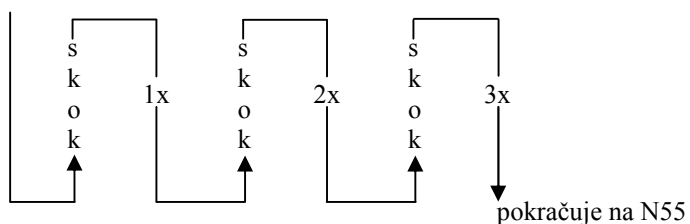
Poznámka:

Pomocné funkce L (číslo volaného PPRG, MRC a čísla bloku, na který bude proveden skok), Q (počet opakování volání PPRG, MRC nebo počet skoku při SPRB) mají platnost pouze v bloku, ve kterém jsou programovány.

Příklad:

N 25
 N 26
 N 27

 N 54 G73 L26 Q3
 N 55



Část partprogramu mezi bloky N26 a N54 se provede jednou a třikrát se zopakuje, tj. provede se celkem čtyřikrát.

V programovém bloku, ve kterém je programován odskok, je přípustné programovat i ostatní posuvové a technologické funkce. Vlastní odskok na definovaný programový blok se provede až v závěru bloku.



3. PROGRAMOVÁNÍ SOUŘADNIC

3.1 Souřadnicový systém - označení souřadnic

Řídicí systém CNC8x6 může ovládat max. 6 souvisle řízených souřadnic (X, Y, Z, 4., 5., 6.). V jednom bloku se smí programovat maximálně 6 souřadnic současně.

Souřadnice X, Y, Z tvoří základní souřadnicový systém (prostor). Osy této prostorové pravoúhlé soustavy jsou rovnoběžné s hlavními vodíci plochami stroje.

Orientace os vyhovující definici podle norem DIN 66217 a ISO/R 841 je pak dána takto: "Osa Z je vždy rovnoběžná s osou pracovního vřetene nebo je s ní souhlasná. Kladný smysl osy Z směřuje od dílce k nástroji. Osa X leží ve vodorovné rovině kolmé k ose Z a je rovnoběžná s povrchem upínací plochy stolu. Je hlavní osou pohybu v rovině. Pro případ rotujícího nástroje s horizontální osou směřuje kladný smysl pohybu osy X vpravo při pohledu od vřetene k rovině upínacího stolu. Osa Y pak doplňuje souřadnice na normální (tedy pravoúhlou, pravotočivou) souřadnicovou soustavu.

Orientace souřadnice je brána ve smyslu relativního pohybu nástroje vůči pevnému obrobku. Pohyby nástrojů se označují písmeny bez čárky (např. X). Pokud se pohybuje v dané ose obrobek vůči pevnému nástroji, jeho pohyb je v opačném smyslu a značíme jej písmenem s čárkou (např. X').

Souřadnicím označeným 4. a 5. je možno přiřadit adresní označení U, V, W, A, B nebo C. Přiřazení adres těmto souřadnicím se provede nastavením příslušných strojních konstant.

Souřadnici 6. je možno přiřadit adresní označení A, B nebo C.

Při přiřazování adres platí zásada, že nesmí být dvě souřadnice označeny stejnou adresou.

Není-li specifikováno požadované přiřazení adres, je standardně přiřazená 4. souřadnici adresa U, 5. souřadnici adresa V a 6. souřadnici adresa C. V následujících kapitolách budeme tyto souřadnice takto označovat.

3.2 Vlastnosti souvislého řízení souřadnic

Všechny souřadnice X, Y, Z, U, V jsou řízeny v polohové vazbě pro pracovní posuvy i rychloposuv. Při takto řízených souřadnicích platí při jejich posuvech jistá funkční závislost, která zaručuje dodržení pohybu po programované trajektorii.

Všechny pohyby (pracovní i rychloposuv) se provádí s plynulým lineárním dosažením požadované rychlosti na začátku pohybu a lineárním snížením požadované rychlosti na minimální rychlost na konci pohybu. Tato vlastnost systému je nazývána rozjezdem a dojezdem, je zaručována při všech druzích interpolací (kromě řezání závitů) a při override a STOPu.

Strmost rozjezdu a dojezdu (velikost zrychlení a zpomalení) je stejná pro pracovní posuvy a rychloposuv. Velikost strmosti rozjezdu a dojezdu se určí při seřízení polohových servomechanismů jednotlivých souřadnic a zadá se systému jako strojní konstanta.

Při obrábění spojitých křivek je rozjezd a dojezd mezi jednotlivými bloky vynechán. Požadovanou spojitost pohybové trajektorie mezi jednotlivými bloky vyhodnocuje systém automaticky.

Souřadnice C je rezervována pro zadávání polohy vřetene pro řízení v polohové vazbě.

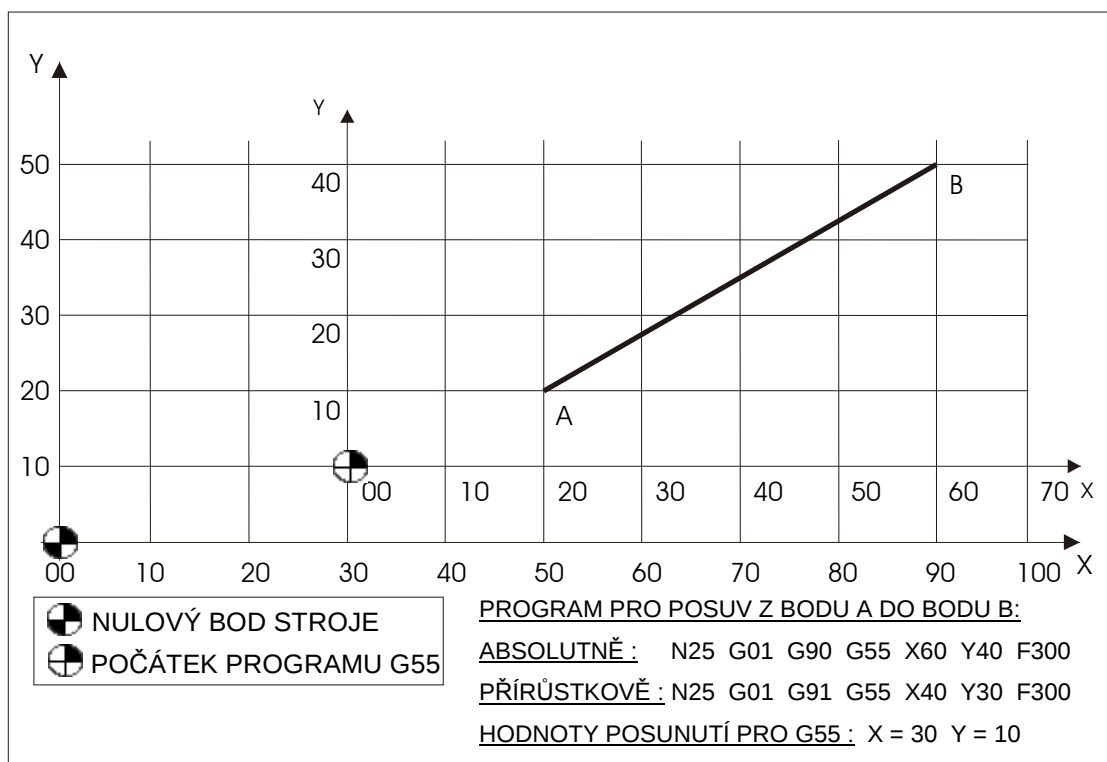
3.3 Zadání dráhy

U přímočarých souřadnic se vektorový průmět křivky do souřadnice (dráha pojezdu souřadnice) udává v mm. Systém umožňuje zpracovávat nejmenější přírůstky po 1 μm . Pro rotační souřadnice se používá programování ve stupních nebo tisícinách stupně.

Dráhu pojezdu je možno programovat volbou příslušné G-funkce ze skupiny 9. buď absolutně nebo přírůstkově (inkrementálně).

Počátek programu je programátorem zvolený bod, ze kterého je naprogramován obráběcí partprogram. Vzdálenost zvoleného počátku partprogramu oproti pevnému nulovému bodu stroje (NBS) se nazývá posun nulového bodu.

Příklad zadání programového segmentu absolutně a přírůstkově je uveden na obr. 1



obr. 1

Hodnoty souřadnic je možno programovat bez desetinné tečky nebo s desetinnou tečkou. Hodnota souřadnice programovaná bez desetinné tečky vyjadřuje u lineární souřadnice (X, Y, Z, U, V, W, I, J, K) údaj v mm, u rotační souřadnice (A, B, C) údaj ve stupních (desetinná tečka se nemusí uvádět).

Příklad:

X 100 = 100 mm = 100000 mikronů
 Y 100.35 = 100.35 mm = 100350 mikronů
 X 0.135 = 0.135 mm = 135 mikronů
 Y .002 = 0.002 mm = 2 mikrony

B 120 = 120°
 C 140.5 = 140.5°

3.4 Nulové body stroje a programu

3.4.1 Definice nulových bodů stroje (NBS)

Nulový bod stroje NBS je pevný bod stroje a tvoří počátek základní souřadné soustavy stroje. Základní souřadná soustava systému se s touto základní souřadnou soustavou stroje ztotožní při najetí do výchozích (referenčních) poloh v režimu REFERENCE. Vzdálenost referenčního bodu od nulového bodu stroje se zadává jako strojní konstanta. (Soubor TAB0.REK, konstanty 80 až 85)

Souřadná soustava programu se s touto základní souřadnou soustavou (stroje i systému) ztotožňuje pomocí funkce G53 (nebo G54) za předpokladu, že hodnoty posunutí pro funkci G53 (nebo G54) v tabulce posunutí počátků nabývají hodnoty 0.

Pozn.: Prioritní posunutí je standardně nastaveno na G53, změnou prioritního bloku lze zvolit za prioritní G54. Popis změny prioritního bloku je uveden v samostatné kapitole v příloze B - „Prioritní blok“.

V případě, že hodnoty posunutí počátku pro funkci G53 (G54) nabývají hodnot různých od nuly, je souřadná soustava programu ztotožněna pomocí funkce G53 (G54) a posunutou souřadnou soustavou daného programu s počátečním bodem NBP0.

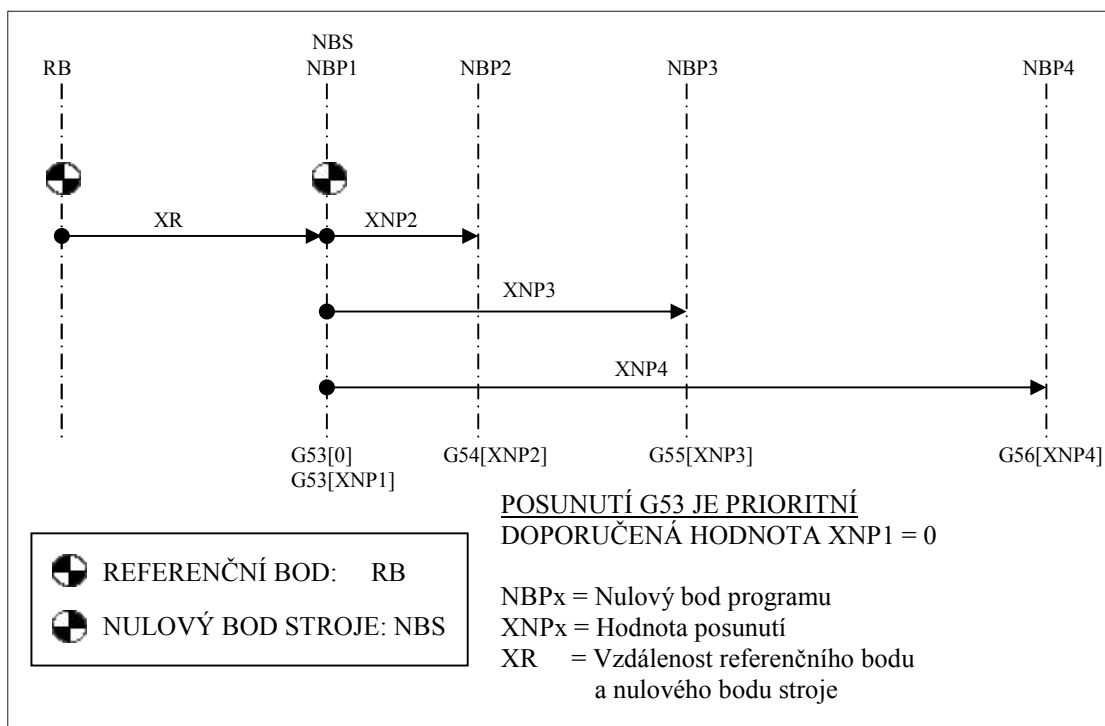
Zadané hodnoty posunutí jednotlivých nulových bodů uvedených v tabulce posunutí počátků pro funkce G53 až G59 jsou vztaženy vždy k nulovému bodu stroje NBS (bez ohledu na to, zda je zadáno pro G53 (G54) nenulové posunutí).

Doporučuje se v tabulce posunutí počátků pro prioritní funkci G53 (G54) programovat nulové hodnoty posunutí souřadnic. Tato funkce je přednostně určena pro transformaci souřadné soustavy programu do základní souřadné soustavy systému (stroje i systému).

RB - Referenční bod stroje je pevný bod stroje, který je určen sepnutím koncového spínače a nulovým průchodem odměřovacího systému.

NBS - Nulový bod stroje, pevný bod stroje, který je vzdálen o hodnotu X_R oproti referenčnímu bodu stroje RB. Hodnota X_R je zadána jako strojní konstanta v tabulce strojních konstant TAB0.REK a nelze ji obecně měnit.

NBP0 až NBP6 - Nulové body programu, obecné body, vůči kterým je vztažen program a které jsou vzdáleny o hodnotu X_{NP} oproti nulovému bodu stroje NBS (hodnota posunutí X_{NP0} platí pro G53, X_{NP1} pro G54, atd.).



obr. 2

3.4.2 Referenční bod stroje

Každá souřadná osa má pevný bod na stroji, který je označen jako referenční bod stroje RB. V okamžiku najetí na RB v režimu REFERENCE (najíetí do výchozích poloh) se do paměti absolutní polohy souřadnice v počítači zapíše hodnota X_R zadaná jako strojní konstanta, a tím dojde k ztotožnění základní souřadné soustavy systému se základní souřadnou soustavou stroje. (Hodnota X_R platí pro první souřadnici (obvykle X), Y_R pro druhou souřadnici (obvykle Y, u soustruhů Z), atd.).

3.4.3 Nulový bod programu

Systém umožňuje posunout počátek souřadné soustavy programu pomocí funkce G53 až G59 do obecného bodu NBP0 až NBP6 v pracovním prostoru stroje (obr.2).

3.4.4 Přesunutí nulových bodů (skupina G5)

Při přesunutí souřadného systému programu (nulového bodu) se vztahují všechny další absolutní míry na nový nulový bod. Při inkrementálním programování nemá posunutí nulového bodu vliv na koncový bod programované dráhy. Při absolutním programování je programovaná koncová poloha dráhy automaticky přepočítána na novou koncovou polohu vztaženou k příslušnému nulovému bodu. Indikace absolutní polohy (okamžité polohy) je vztažena vždy vůči některému bodu stroje nebo programu (referenčnímu, nulovému bodu stroje nebo nulovému bodu programu). Vůči kterému bodu je indikace vztažena, je určeno G funkcí z páté skupiny.

Hodnota posunutí výše uvedených bodů vůči NBS (nulovému bodu stroje) je zadána v následující tabulce posunutí počátku:

FCE	HODNOTY POSUNUTÍ V OSÁCH						
	1.osa	2.osa	3.osa	4.osa	5. osa	6.osa	
G53	1 _{NP0}	2 _{NP0}	3 _{NP0}	4 _{NP0}	5 _{NP0}	6 _{NP0}	Trvalá prioritní
G54	1 _{NP1}	2 _{NP1}	3 _{NP1}	4 _{NP1}	5 _{NP1}	6 _{NP1}	Trvalá funkce (může být prioritní)
G55	1 _{NP2}	2 _{NP2}	3 _{NP2}	4 _{NP2}	5 _{NP2}	6 _{NP2}	Trvalá funkce
G56	1 _{NP3}	2 _{NP3}	3 _{NP3}	4 _{NP3}	5 _{NP3}	6 _{NP3}	Trvalá funkce
G57	1 _{NP4}	2 _{NP4}					Trvalá funkce
G58	1 _{NP5}						Platí pouze v jednom bloku
G59	1 _{NP6}				5 _{NP6}	6 _{NP6}	Platí pouze v jednom bloku

Při programování posunutí nulového bodu v bloku pomocí funkce G53 až G59 se žádný pohyb souřadnice nevykoná, pokud není v tomto bloku pohyb definován jinak. V programu použitá funkce G53 až G59 musí mít před svým užitím v bloku definované hodnoty posunutí v jednotlivých souřadnicích v tabulce posunutí počátků. Hodnoty definované v tabulce posunutí počátků zůstávají trvale zachovány, dokud nejsou přepsány v editačním režimu na novou hodnotu nebo změněny interaktivním zadáním (viz Návod k obsluze) nebo pomocí funkce G92 nebo G93 z partprogramu.

Funkce G53 až G57 mají trvalou platnost, tzn., že příslušnou G funkcí zvolená souřadná soustava platí ve všech následujících blocích, dokud není přepsána jinou G funkcí 5. skupiny. Funkce G58 a G59 mají platnost pouze v bloku, ve kterém jsou programovány, v dalším bloku se systém vrací do posunuté souřadné soustavy zvolené dříve programovanou některou z funkcí G53 až G57.

3.4.5 Plnění tabulky posunutí počátků

Systémovou tabulku posunutí počátků je možné plnit třemi způsoby:

- Z partprogramu pomocí funkce G92 nebo G93
- Z panelu obsluhy edicí souboru TAB0.POS
- Z panelu obsluhy interaktivním zadáním

Plnění tabulky posunutí počátků z panelu obsluhy je podrobně popsáno v Návodu k obsluze řídicího systému CNC8x6. Nedoporučuje se kombinovat plnění posunutí z partprogramu a z panelu obsluhy. Vhodnější je vybrat si jeden ze způsobů a ten používat.

3.4.6 Plnění tabulky posunutí z partprogramu funkcí G92 a G93

Plnění tabulky z partprogramu není v praxi často používáno. Pokud se používá, nezáleží na obsahu souboru TAB0.POS, ze kterého se po zapnutí systému přepisují hodnoty posunutí do systémové tabulky posunutí. Posunutí je definované přímo v každém partprogramu.

Definici posunutí nulových bodů rozumíme přiřazení číselných hodnot posunutí počátků souřadnic pro jednotlivé G funkce 5-té skupiny. Hodnoty posunutí počátků jsou uloženy pro jednotlivé G funkce 5-té skupiny v tabulce. Tabulku posunutí počátků je možno přepisovat a zapisovat do ní následujícím způsobem:

- Nahrání hodnot tabulek posunutí počátku z programového bloku partprogramu pomocí funkce G92. Při zadání a provedení programového bloku s funkcí G92 dojde k zápisu hodnot do tabulky posunutí počátků pro příslušnou funkci G53 - G59, které jsou programovány v tomto bloku pod hodnotami příslušných souřadnic. Programový blok, kterým požadujeme posunutí zapsat, musí mít např. následující tvar:

Nxxxx G55 G92 X100.5 Y200. Z-300.650 U0 V10.

V takto zadaném bloku se do tabulky G55 zapíše posunutí pro osu X 100.5, pro osy Y 200 atd. Místo funkce G55 může být programována i jiná funkce ze skupiny G[5]. V bloku dojde k nastavení hodnot v tabulce posunutí pro pět souřadnic, hodnota zapsaná v tabulce pro šestou souřadnicí zůstane beze změny. Je-li v bloku programována funkce G92, údaje zapsané u jednotlivých souřadnic mají význam hodnot, které se zapíše do tabulky. **V tomto bloku nedojde k žádnému pohybu !** Do tabulky se zapíše pouze hodnoty u programovaných souřadnic. Pokud souřadnice není programovaná, posunutí v tabulce pro tuto souřadnicí zůstane beze změny.

- b) Automatické nastavení hodnot tabulky posunutí počátku z programového bloku pomocí funkce G93. Při zadání a provedení programového bloku s funkcí G93 dojde k automatickému nastavení hodnot posunutí počátků pro příslušnou funkci G53 - G59, při které dojde k ztotožnění nulového bodu programu pro danou funkci G53 -G59 s jistým bodem souřadnicového systému, který je vzdálen od skutečné polohy nástroje o hodnoty programovaných souřadnic v daném bloku. Tvar programovaného bloku pro tuto operaci je shodný s blokem uvedeným v bodu a) s tím rozdílem, že funkce G92 je nahrazena funkcí G93.

Operaci používání funkce G93 podle b) vysvětlíme na následujících dvou základních příkladech.

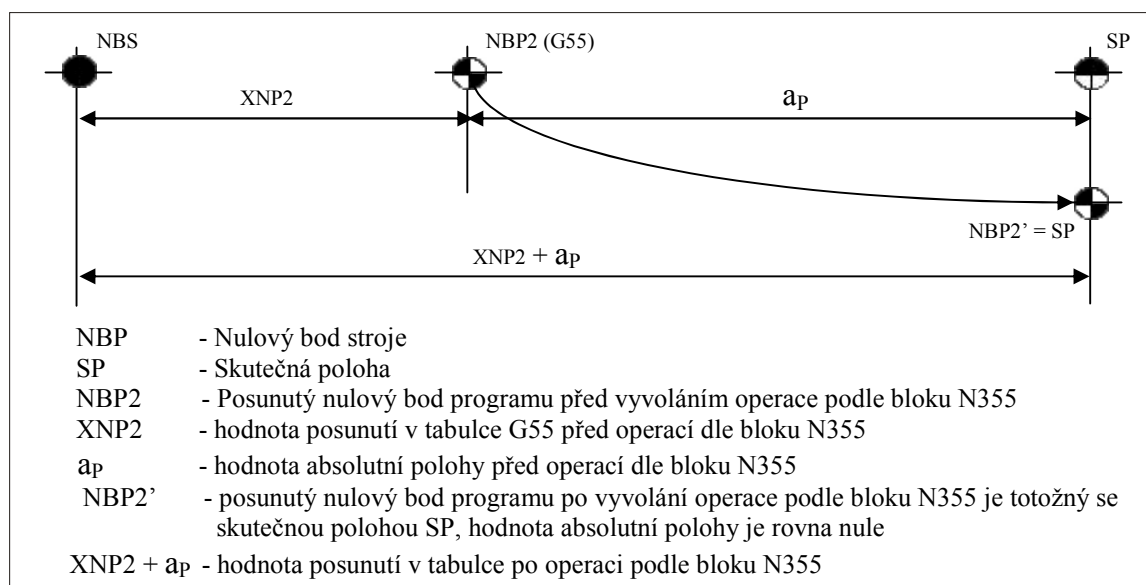
Hodnota programované souřadnice se rovná 0 (obr.3).

V tomto případě je souřadná soustava programu ztotožněna pro danou funkci G53 až G59 svým počátkem s bodem, ve kterém se souřadnice nalézá. Indikace absolutní polohy (okamžité polohy) bude v tomto případě ukazovat hodnotu 0.

Příklad bloku:

N 355 G55 G93 X0

Postup určení nové hodnoty posunutí ($X_{NP2} + a_p$):



obr. 3

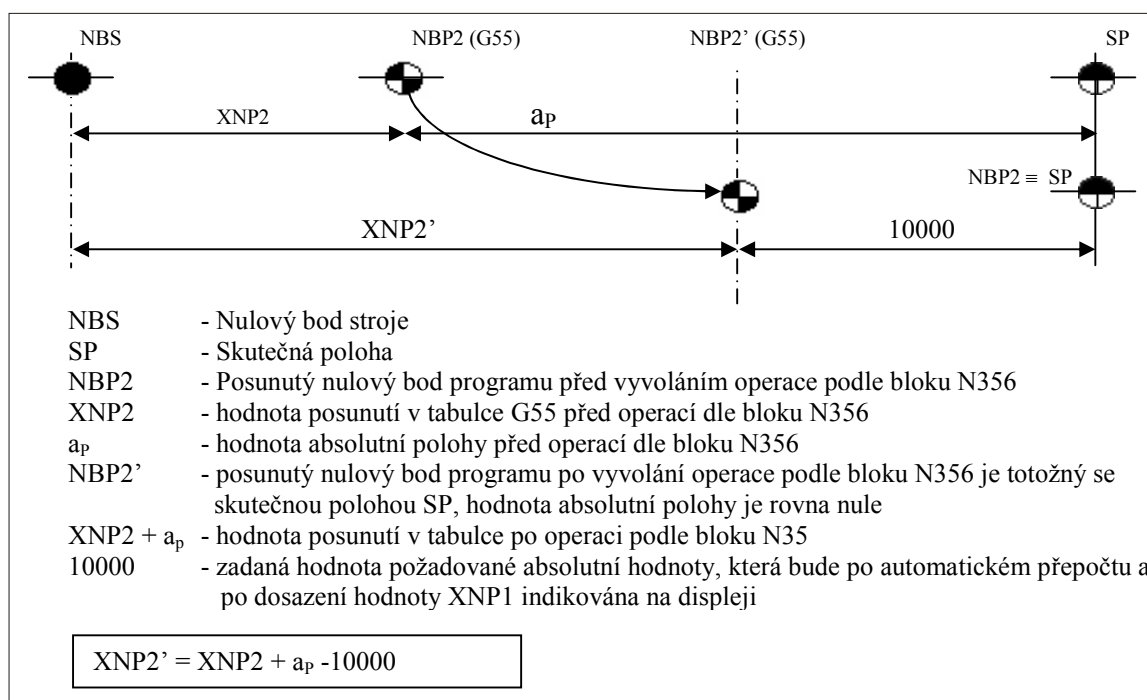
Hodnota programované souřadnice je rozdílná od 0 (obr. 4)

V tomto případě je souřadná soustava programu ztotožněna pro danou funkci G54 a G59 svým počátkem s bodem, který je vzdálen od bodu, v kterém se souřadnice nalézá o hodnoty programovaných souřadnic v daném bloku. Indikace absolutní polohy (OP-okamžité polohy) bude v tomto případě rovna programovým hodnotám souřadnic.

Příklad bloku:

N 356 G55 G93 X+10.000

Postup určení nové hodnoty posunutí ($X_{NP2'}$):



obr. 4

Upozornění:

Hodnoty, které se zapíší pomocí funkcí G92 nebo G93 v partprogramu do systémové tabulky posunutí se nezapisují do souboru TAB0.POS !

3.4.7 Plnění tabulky posunutí edicí souboru TAB0.POS

Tento způsob se používání je nejčastější. Hodnoty posunutí se pomocí editoru upraví v souboru TAB0.POS. Uložení tohoto souboru dojde k přepsu hodnot z tohoto souboru do systémové tabulky posunutí a tato posunutí platí až do další změny, která se případně v editoru provede. Posunutí uvedená v tomto souboru se do systémové tabulky přepíší i po zapnutí systému, takže jsou okamžitě platná. V partprogramu se používají pouze funkce G53 – G59 pro volbu jednotlivých posunutí.

Podrobný popis ovládání editoru pro edici souboru TAB0.POS je uveden v „Návodu k obsluze“.

3.4.8 Plnění tabulky posunutí interaktivním zadáním

Použití je stejné jako bylo uvedeno v předešlé kapitole, pouze způsob zápisu do souboru TAB0.POS je jiný – ne pomocí editoru, ale interaktivním ovládáním. Postup je popsán v „Návodu k obsluze“.

3.4.9 Volání posunů nulových bodů

K vlastnímu posunutí souřadného systému dojde během chodu programu v okamžiku výkonu bloku s programovanou G-funkcí 5-té skupiny. V tomto bloku a při G53 až G57 i v dalších blocích (mají trvalou platnost) se začnou brát jak programované souřadnice, tak poloha nástroje vzhledem k posunutému počátku. Prioritní G funkcí ze skupiny 5 je funkce G53 (G54), která se nastaví při odstartování nového programu

v automatickém režimu, nebo při dosažení konce programu. Hodnoty posunutí počátků pro funkce G53 až G57 uložené v tabulce posunutí zůstávají zachovány trvale, dokud nejsou přepsány na jiné hodnoty (při začátku ani konci programu nejsou ovlivňovány). Hodnotu posunutí nulového bodu platící pro kteroukoliv výše uvedenou G funkci je možno rovněž definovat v průběhu provádění některého aut. režimu při zastavení programu a přechodu do režimu TAB. Při plnění tabulek posunutí pomocí G92 a 93 se plněné posunutí uplatní v následujícím bloku.

Příklad:

N10 G00 X10 Y20	“Pokud není v prvním bloku zařazeno žádné posunutí, platí prioritní
	posunutí G53 (G54)
N20	“Platí stále G53 (G54)
N30 G55 ...	“Zařadí se posunutí G55
N40	“Platí posunutí G55
N50 G58 ...	“Pouze v tomto bloku se zařadí a platí posunutí G58
N60	“V tomto bloku se vrátí platné posunutí G55 (aniž by bylo programované)
N70 ...	“Platí posunutí G55

4

4. DRUHY POHYBU

4.1 Stavění souřadnic - funkce G00

Stavěním souřadnic se rozumí přemístění nástroje do koncového (programovaného) bodu rychloposuvem. Koncová poloha se programuje v absolutních nebo inkrementálních mírách (platí obecně pro všechny druhy pohybů). Toto přemístění nástroje je zadáno v bloku funkcí G00, která je současně nositelem informace pro provádění pohybu rychloposuvem. Velikost rychloposuvu je v systému zadána pevně jako strojní konstanta a v bloku se neprogramuje. Při rychloposuvu je zaručen plynulý rozjezd a dojezd na začátku a konci pohybu. V jednom bloku je možno programovat stavění jedné až šesti souřadnic.

4.2 Lineární interpolace - funkce G01

Lineární interpolace se volí funkcí G01. V jednom bloku je možno naprogramovat interpolaci mezi jednou až šesti souřadnicemi naprogramováním souřadnic koncového bodu v příslušných osách. Je tedy možná i vzájemná kombinace lineárních a rotačních souřadnic. Pro lineární interpolaci je nutná zadaná posuvová rychlost pod adresou F. Rychlost F nemusí být uvedena v bloku s G01, ale v kterémkoli předcházejícím bloku.

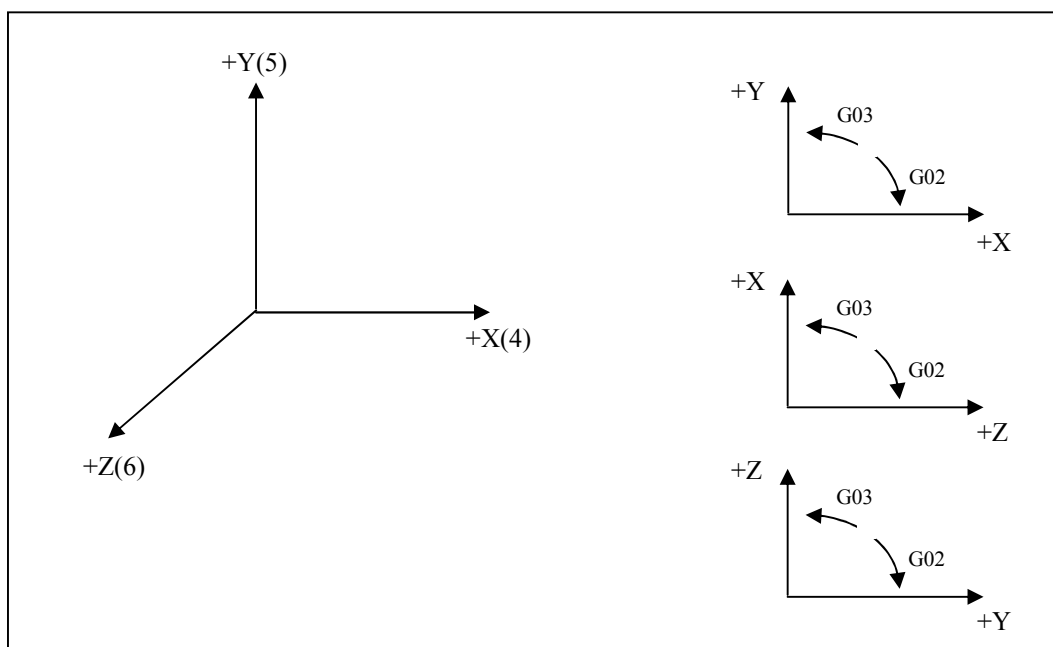
Příklad průběhu dráhy při lineární interpolaci:

Absolutní programování:	Přírůstkové programování:
N10 G01 G90 X 90.000 Y 50.000	N11 G01 G91 X 40.000 Y 30.000

Je-li výchozím bodem interpolace bod A o souřadnicích X=50, Y=20 při zařazeném posunutí G54, které je totožné s NBS a koncový bod B o souřadnicích X=90 Y=50, potom uvedené bloky N10 i N11 vykonají stejnou dráhu.

4.3 Kruhová interpolace - funkce G02, G03

Kruhová interpolace se volí buď funkcí G02 (pohyb po kružnici ve směru hodinových ručiček) nebo G03 (pohyb po kružnici proti směru hodinových ručiček). Vyjádření směru kruhové interpolace (G02 nebo G03) v libovolných rovinách pro souřadnicový (pravotočivý) systém se určuje při pohledu na rovinu kruhové dráhy (obr.5).

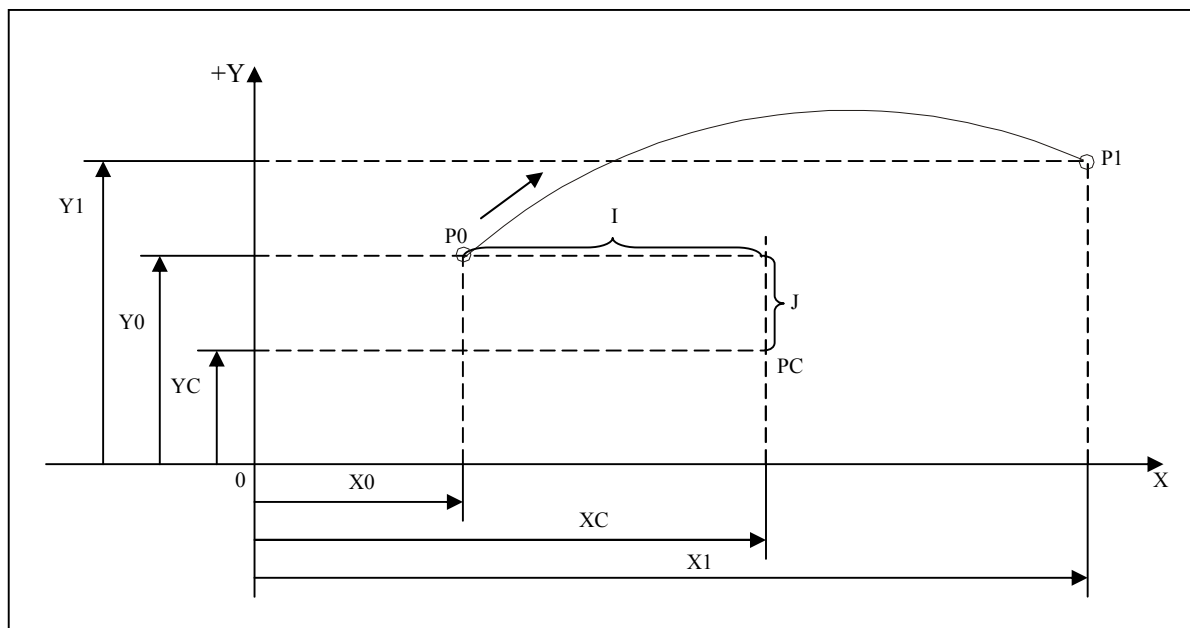


obr. 5

V závorkách jsou uváděny souřadnice v systému uvažované jako paralelní k původním. Kruhová interpolace je možná pouze v rovině. Za rovinu interpolace můžeme zvolit rovinu určenou dvěma ze šesti souřadnic X - Y, Y - Z, Z - X, 4.- 5. atd. Teoreticky se může kruhová interpolace programovat i v případech, že je některá ze souřadnic rotační, ale programování je značně složité.

Kruhová interpolace se zadává souřadnicemi **koncového** bodu kružnice nebo kruhového oblouku. Souřadnice koncového bodu musí být zadány **obě** a to i v případě, že koncový bod je totožný s počátečním. Souřadnice koncového bodu lze zadávat absolutně nebo přírůstkově.

Souřadnice středu kružnice se programují adresami I a J. Na rozdíl od koncového bodu se **souřadnice středu kružnice musí programovat pouze přírůstkově vzhledem k počátečnímu bodu kružnice** (k počátku kruhové interpolace). Střed vzhledem k první ose se programuje adresou I, střed vzhledem k druhé ose adresou J. V této souvislosti je třeba podotknout, že při kruhové interpolaci v rovině Z - X se za první osu považuje osa Z a za druhou osa X (obr.5), tedy vzdálenost středu kružnice od počátku v ose Z se programuje adresou I a v ose X adresou J !



obr. 6

- P0 - počáteční bod kruhové interpolace
- X0,Y0 - souřadnice počátečního bodu
- P1 - koncový bod kruhové interpolace
- X1,Y1 - souřadnice koncového bodu
- PC - střed kružnice (kruhového oblouku)
- XC,YC - souřadnice středu kružnice

Kruhová interpolace na obr.6 je G02 (ve směru hod. ručiček).

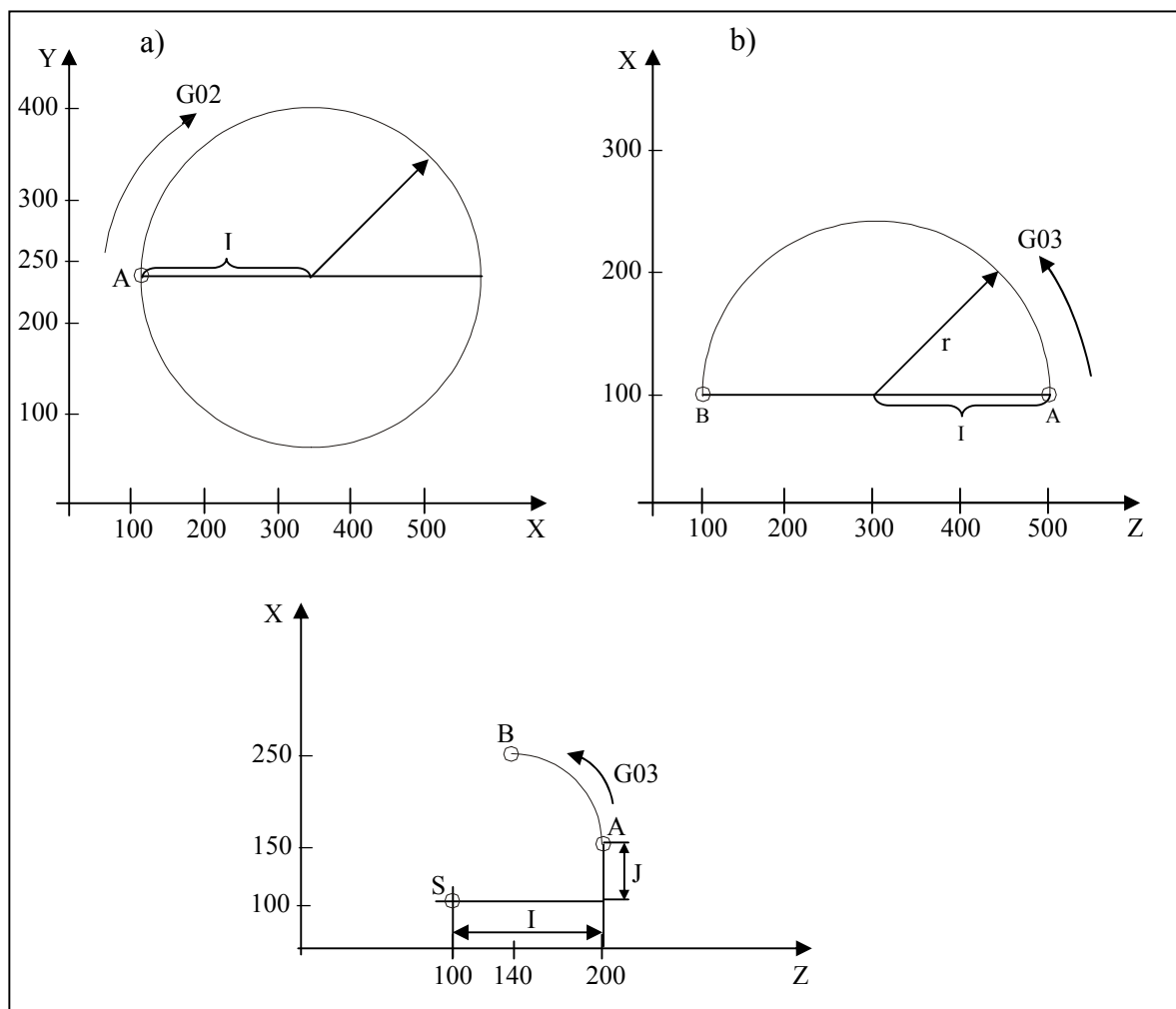
Souřadnice středu kružnice pro osu X je vzdálená o hodnotu I od počátku:

$$I = XC - X0$$

Souřadnice středu kružnice pro osu Y je vzdálená o hodnotu J od počátku:

$$J = YC - Y0$$

Znaménko u adresy J bude záporné, neboť střed kružnice se vzhledem k počátku vzdaluje v záporném směru. Pokud je adresa I nebo J nulová, nemusí se v bloku programovat.



obr. 7

Příklad programu pro celou kružnici v rovině X - Y (obr. 7a):

Absolutně:

N19 G02 G90 X200.0 Y250.0 I200.0 F120

Přírůstkově:

N19 G02 G91 X0 Y0 I200.0 F120

Příklad programu pro půlkružnici v rovině Z - X (obr. 7b):

Absolutně:

N20 G03 G90 X100.0 Z100.0 I-200.0 F100

Přírůstkově:

N20 G03 G91 X0 Z-400.0 I-200.0 F100

Příklad programu pro úsek kružnice v rovině X - Z (obr. 7c):

Absolutně:

N21 G03 G90 X250.0 Z140.0 I-100.0 J-50.0 F100

Přírůstkově:

N21 G03 G91 X100.0 Z-60.0 I-100.0 J-50.0 F100

Pozn.:

Kruhová interpolace není omezena na jeden kvadrant. Programovaná kružnice může procházet i více kvadranty.

Pokud se naprogramuje u kruhové interpolace souřadnice koncového bodu, který neleží na kružnici, systém vyhlásí chybu číslo 7.56 - "KONCOVÝ BOD NELEŽÍ NA KRUŽNICI". Koncové body i souřadnice středu je nutné do partprogramu zadávat s přesností na jeden mikron! Přesnost zadání lze částečně zmenšit nastavením strojní konstanty číslo 55 – tolerance středu kružnice.

5

5. ŘEZÁNÍ ZÁVITŮ NOŽEM

5.1 Řezání závitu bez výběhu

Při programování řezání závitů nožem funkcí G33 sváže systém pohyb v souřadnici, pro níž bylo zadáno stoupání, s pohybem vřetene. Pohyb ostatních interpolačních souřadnic je prováděn tak, aby byl výsledný pohyb v zadaných souřadnicích proveden po zadané trajektorii.

Blok závitování se programuje pod těmito adresami:

F	stoupání závitů, zadává se v mm/otáčku.
X,Y,Z,...	souřadnice koncového bodu závitu (lze programovat absolutně nebo přírůstkově), ve které je pohyb svázán s otáčením vřetene
I	úhlové posunutí začátku svázaného pohybu od NULOVÉHO PULSU
M03,M04	směr točení vřetene
S	otáčky vřetene
G33	Závitování

Maximální hodnota stoupání, zadaná přímo adresou F, je 99,99 mm/ot.

Je-li I = 0 nebo není-li programováno vůbec, začne vzájemná vazba vysílaných pulsů ze snímače otáček vřetene působit od vyslání nulového pulsu snímačem. Je-li pod adresou I programována hodnota, znamená úhlové posunutí začátku svázaného pohybu od nulpulsu. Úhlové posunutí začátku závitování má praktický význam pouze při programování vícechodých závitů.

Závitování je podobné jako programování posuvu v mm/ot funkcí G95. Jediný rozdíl je, že při programování G33 se čeká pro zahájení pohybu na NULPULS, kdežto při G95 se pro zahájení posuvu na NULPULS nečeká. Čekání na nulový puls je nutné pro zajištění opakovaného přesného najetí do závitu při vícenásobných průchodech.

Příklad 1:

Příklad uvádí část programu s rozepsaným závitováním, tj. **není** použit závitovací cyklus G84. Závitování v ose Z začíná na míře Z3.0mm a končí na míře -9.0 mm v zápichu, stoupání je 1,5mm:

```

.....
N325 G00 G95 T9 &I100 D9 S1000 M03 M42      "HLAVNI VETA
N330 G00 X26.618 Z3.                         "NAJEZD RYCHLOPOSUVEM NA ZACATEK ZAVITU
N335 G33      Z-9. I0. F1.5                 "ZAVITOVANI
N340 G00 X29.                                "VYJEZD RYCHLOPOSUVEM ZE ZAPICHU
N345      Z3.                                "NAVRAT G00 DO VYCHOZI POLOHY
N350 X26.418                                  "VYCHOZI BOD DRUHEHO PRUCHODU
N355 G33      Z-9. I0. F1.5                 "ZAVITOVANI
N360 G00 X29.
N365      Z3.
N370 X26.177                                  "VYCHOZI BOD TRETIHO PRUCHODU

```

N375 G33 Z-9. I0. F1.5 "ZAVITOVANI
N380 G00 X29. "atd.

Příklad 2:

Blok závitování v ose Z, začátek posunut o 180 stupňů od nulpulsu:

N20 M04 S500 Z200.0 F1.5 I180 G33

Příklad 3:

Závitování na kuželu:

Při závitování na kuželu se programuje navíc druhá osa, ve které se provádí posuv. U soustruhů je to osa X
Uvedený blok vytvoří na délce 100mm kuželový závit z počátečního průměru 226mm na konečný průměr 246mm.

N125 G00 G95 T9 & I100 D9 S500 M03 M42 "HLAVNI VETA
N130 G00 X226 Z3. "NAJEZD RYCHLOPOSUVEK NA ZACATEK ZAVITU
N135 G33 X246 Z-100 I0. F1.5 "ZAVITOVANI

5.2 Programování závitů s výběhem

Programování je principiálně stejné jako u závitu bez výběhu. Do bloku se pouze doplní pod **adresou J délka výjezdu** (vždy kladná hodnota, bez ohledu na směr posuvu). **Programovaná koncová poloha závitu je v tomto případě včetně délky výjezdu. Za blokem závitování musí být programován prázdný blok. V dalším bloku musí být programovány všechny souřadnice. Není-li požadavek na pohyb, musí se zopakovat koncové polohy.**

Příklad 4:

Závitování v ose Z se stoupáním 6mm a výjezdem 12mm:

N10 M03 S50 "ROZTOČENÍ VŘETENE
N20 Z5. X64. "VÝCHOZÍ POLOHA
N30 Z-100 F6.0 J12 G33 "ZÁVITOVÁNÍ s VÝJEZDEM 12mm
N40 "PRÁZDNÝ BLOK
N50 X64. Z-100. G0 "MUSÍ BÝT PROGRAMOVÁNY VŠECHNY OSY

5.2.1 Úhel výjezdu ze závitu

Úhel výjezdu ze závitu je možné zadat dvěma způsoby:

- A) strojní konstantou
- B) parametrem

A) Zadání strojní konstantou

Ve strojní konstantě č.8 (soubor TAB0.REK) se nastavuje úhel výjezdu ze závitu (maximálně 60 stupňů) a to včetně znaménka.

Jednotlivé dekády mají tento význam (zn = znaménko):

Dekády:	8	7	6	5	4	3	2	1
Zn	0	0	1	1	2	5	0	0

Do 1. až 5. dekády se zadává úhel výjezdu s přesností na tisíciny. V uvedeném příkladu je nastaven úhel 12, 500 stupňů.

V 6 dekádě se zadává pořadové číslo osy, která se podílí na výjezdu ze závitu. U soustruhů je to osa X, t.j. je zadána 1. (Osa Y je 2, osa Z je 3 atd).

7. a 8. dekáda jsou bez významu – musí být uvedeny nuly.

Znaménko určuje, je-li úhel výjezdu kladný nebo záporný (např. u vnitřních závitů může být záporný).

Tento způsob zadání úhlu výjezdu ze závitu (tj. pomocí strojní konstanty) je v praxi použitelný pouze v případě, že se požaduje vždy stejný úhel výjezdu a především stejný směr výjezdu ze závitu.

B) Zadání parametrem

Pokud potřebujeme měnit úhel výjezdu a především směr výjezdu, což je v praxi nezbytné pokud se programují vnitřní i vnější závity, musí se úhel a směr výjezdu programovat parametrem, jehož číslo je určené ve stejné strojní konstantě (č.8)

Jednotlivé dekády potom mají tento význam (zn. = znaménko):

Dekády:	8	7	6	5	4	3	2	1
zn.	0	4	0	0	0	0	0	0

7 a 8 dekáda určuje číslo parametru, ve kterém je programován úhel a směr výjezdu, ostatní dekády nemají v tomto případě (závitování s výjezdem) význam.

V příkladu je v 7 a 8 dekádě 04, tj. úhel a směr se převezme z parametru R04.

Důležitá poznámka:

Obecně se může volit libovolný parametr (**kromě R00 a R80**) pokud se závitování rozepisuje v partprogramu samostatně. **Pokud se však používají pevné cykly** (soubory PEVNECY4.NCP, PEVNECY5.NCP nebo PEVNECY6.NCP), **musí se použít pouze parametr R04**, jak je uvedeno v příkladu. Doporučuje se tedy volit, pokud je to možné, tento parametr.

Parametr je naplněn stejným způsobem jako strojní konstanta, pokud je v ní určen úhel. Například pro úhel výjezdu 20 stupňů ve směru do mínusu, přičemž výjezd bude v ose X bude parametr naplněn takto:

N10 R04= -00120.000

Pozor: Nesmí se opomenout uvést v 6. dekádě číslo osy, která „vyjíždí“ ze závitu. Pro soustruhy je to X tj. píše se 1.

Příklad 5.:

Je stejný jako příklad 4, pouze směr výjezdu je zadán nikoli strojní konstantou, ale parametrem R04. Ten může být naplněn kdykoli před vlastním závitováním.

N10 M03 S50	“ROZTOČENÍ VŘETENE
N20 Z5. X64. R04=120.0	“VÝCHOZÍ POLOHA a ZADANÍ UHLU A SMERU VYJEZDU
N30 Z-100 F6.0 J12 G33	“ZÁVITOVÁNÍ s VÝJEZDEM 12mm
N40	“PRÁZDNÝ BLOK
N50 X64. Z-100. G0	“MUSÍ BÝT PROGRAMOVÁNY VŠECHNY OSY

5.3 Závitování s vjezdem a výjezdem (výběhem)

Pozn: Platí od verze panelu 30.29

V některých speciálních případech je vhodné nebo dokonce nutné mít možnost naprogramovat závitování s vjezdem („výběh na začátku závitu“). V praxi se to může využít, pokud je z technologických nebo jiných důvodů nutné vytvořit závit s výjezdem „od konce“. Někdy může být i požadavek na závit s vjezdem i výjezdem například pro tvoření spirálové mazací drážky apod.

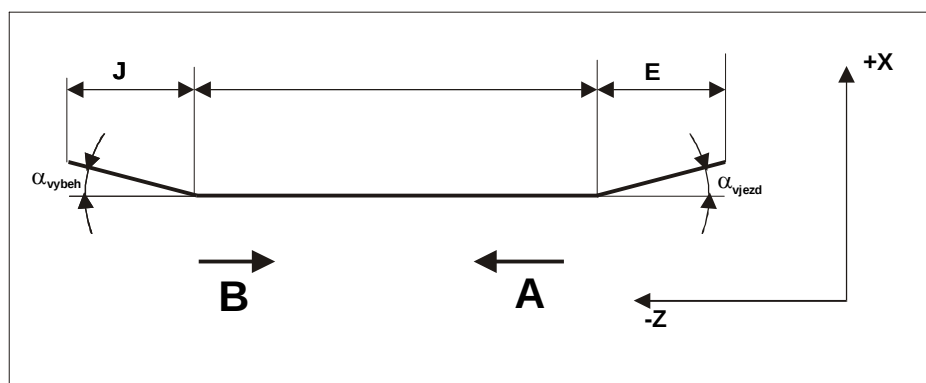
Délka „vjezdu“ se zadává adresou E. Co je vjezdem a co výjezdem ze závitů je dáno směrem pohybu. Pokud se pohybujeme ve směru A (viz obr.) je vjezdem hodnota kótovaná na obr. písmenem E a výjezdem hodnota kótovaná písmenem J. Tyto adresy by byly i v programu.

Pokud by pohyb byl ve směru B (viz obr.) vjezdem by byla hodnota zakótovaná na obr. písmenem J, ale v programu by byla adresa E ($E = \text{vjezd}$), výjezdem by byla hodnota zakótovaná na obr. písmenem E, ale v programu by byla adresa J ($J = \text{výjezd}$).

Pozn.:

Adresu E je nutné vždy programovat s desetinnou tečkou, tj. vjezd 10mm je nutno programovat E10. nebo E10.0, u adresy J to není nutné, ale může se rovněž s tečkou programovat J10 je stejné jako J10.0

Úhel vjezdu i výjezdu se zadává v rozsahu 0 až 60 stupňů. **Znaménka úhlů se programují stejná pro vjezd i výjezd** (určující směr je výjezd ze závitů), přičemž systém si sám změní směr vjezdu do závitů. Pro příklad uvedený na obrázku by byl zadán vstupní i výstupní úhel kladně.



Pravidla pro zadání úhlu vjezdu a výjezdu jsou podobná jako u závitování s výběhem.

A) Zadání strojní konstantou

Použitelné v případě, že úhel vjezdu i výjezdu je stejný a je používán jen jeden směr výjezdu a vjezdu. Jednotlivé dekády mají tento význam (zn = znaménko):

Dekády:	8	7	6	5	4	3	2	1
Zn	0	0	1	1	2	5	0	0

Do 1. až 5. dekády se zadává úhel výjezdu i vjezdu s přesností na tisíciný. V uvedeném příkladu je nastaven úhel 12,500 stupňů.

V 6. dekádě se zadává pořadové číslo osy, která se podílí na výjezdu i vjezdu do závitů. U soustruhů je to osa X, t.j. je zadána 1. (Ostatní případy nemají praktický význam).

7. a 8. dekáda jsou bez významu – musí být uvedeny nuly.

Znaménko určuje, je-li úhel výjezdu i vjezdu kladný nebo záporný (např. u vnitřních závitů může být záporný)

B) Zadání parametrem

Pokud potřebujeme měnit úhel výjezdu a vjezdu a také směr výjezdu a vjezdu, což je v praxi nezbytné pokud se programují vnitřní i vnější závitů, musí se úhel a směr výjezdu a úhel a směr vjezdu programovat parametry, jejichž čísla jsou určena ve stejné strojní konstantě (č.8)

Jednotlivé dekády potom mají tento význam (zn. = znaménko):

Dekády:	8	7	6	5	4	3	2	1
zn.	1	5	1	6	0	0	0	0

7 a 8 dekáda určuje číslo parametru, ve kterém je programován úhel a směr výjezdu,
5 a 6 dekáda určuje číslo parametru, ve kterém je programován úhel a směr vjezdu,

Ostatní dekády nemají v tomto případě význam.

V příkladu je v 7 a 8 dekáde 15, tj. úhel a směr výjezdu se převezme z parametru R15.

v 5 a 6 dekáde 16, tj. úhel a směr vjezdu se převezme z parametru R16.

Důležitá poznámka:

Obecně se mohou volit libovolné parametry ,kromě R00 a R80.

Parametry jsou naplněny stejným způsobem jako strojní konstanta, pokud je v ní určen úhel. Například pro úhel výjezdu 20 stupňů ve směru do mínusu, přičemž výjezd bude v ose X a pro úhel vjezdu 18 stupňů ve směru do mínusu, přičemž vjezd bude v ose X budou parametry naplněny takto:

N10 R15= -00120.000 R16= -00118.000

Pozor: nesmí se opomenout uvést v 6. dekáde číslo osy, která „vyjíždí“ ze závitů. Obě čísla by u obou parametrů měla být stejná. Pro soustruhy je to X tj. píše se 1. Znaménka se uvedou obě stejná (rozhoduje výběh) i když ve skutečnosti pojede vjezd opačným směrem.

Úhel pro vjezd i výjezd se programuje v rozsahu 0 až 60 stupňů. Systém si sám určí orientaci úhlu.

Příklad části programu závitování s výběhem, dělaným „od konce“, tj. je použit vjezd do závitu, výjezd není použit. Pro zadání vjezdu použit parametr R16, úhel vjezdu je 18 stupňů a je kladný (o znaménku rozhoduje úhel výjezdu, i když v tomto případě není použit!):

N10 M04 S50	“ROZTOČENÍ VŘETENE
N20 X71.798 Z-100. R16=118.0	“VÝCHOZÍ POLOHA a ZADÁNÍ UHLU A SMERU VJEZDU
N30 Z3 F6.0 E12 G33	“ZÁVITOVÁNÍ s VJEZDEM 12mm
N40	“PRÁZDNÝ BLOK
N50 X64. Z3 G0	“MUSÍ BÝT PROGRAMOVÁNY VŠECHNY OSY

Pozor:

Na rozdíl od závitování s výjezdem, kde se před závitem najede na požadovaný průměr a na něm se provede závit se u závitování s vjezdem musí najet na takovou míru v ose X, aby se po vjezdu dostal na požadovaný průměr! Hodnota se spočítá z úhlu a délky vjezdu. V uvedeném příkladu by měl být závit proveden na průměru 64mm.

5.4 Závitování s velkým stoupáním

Pokud je potřeba vytvořit závit se stoupáním větším než 99.99mm, použije se parametrické zadání stoupání u adresy F. Například pro stoupání 250mm bude zadání partprogramu následující:

```
.....
N50 R10=250.0          "zadání hodnoty 250.0 do parametru R10
N60 M03 S100          "roztočení vřetena
N70 Z200.0 FR10 I0 G33 "stoupání F zadáno parametricky v parametru R10
.....
```

Pozn.: Platí pro všechny typy závitů (bez výběhu, s výběhem i s vjezdem).

6

6. ZADÁNÍ POSUVU

Pracovní posuv po obráběné křivce se programuje pod adresou F. Posuv je zadáván způsobem podle G-funkce skupiny G6, nebo-li tato funkce určuje rozměr adresy F.

Při programování funkce G94 a G96 vyjadřuje hodnota u adresy F:

a) je-li programována bez desetinné čárky, posuv v mm/min.

Příklad:

N1 G94 F 1000 znamená posuv 1000mm/min.

b) je-li programována s desetinnou čárkou, posuv m/min

Příklad:

N2 G94 F.5 znamená posuv 0.5m/min tj. 500mm/min

N3 G94 F1.2 znamená posuv 1.2m/min tj. 1200mm/min

Systém připouští max. zadání velikosti posuvu 90000 mm/min (t.j. 90 m/min).

Při programování funkce G95 a G97 vyjadřuje hodnota u adresy F:

a) je-li programována bez desetinné čárky posuv v $\mu\text{m}/\text{ot}$

Příklad:

N4 G95 F 1000 znamená posuv 1000 $\mu\text{m}/\text{ot}$.

b) je-li programován s desetinnou čárkou posuv v mm/ot

Příklad:

N5 G95 F 1.0 znamená posuv 1mm/ot.

N6 F0.2 znamená posuv 0.2mm/ot.

Systém připouští maximální zadání velikosti posuvu na otáčku 99,999 mm/ot (F99999 nebo F99.999). Pro posuv na otáčku nutno snímat pulsy od vřetene a proto stroj se systémem umožňuje i závitování. Otáčkový posuv se může uplatnit i u ručních režimů.

Maximální možná programovatelná rychlost pro soupravu systému s daným strojem je dána vlastnostmi stroje. Při programování velikosti posuvu na otáčku je maximální hodnota programovatelného posuvu [mm/ot] závislá na max. přípustné posuvové rychlosti V_{max} a zadaných otáčkách vřetene S.

Platí vztah :

$$V = S \cdot \beta$$

kde V - posuvová rychlost v mm/ot

S - otáčky vřetene v ot/min

β - posuv v mm/ot (programován adresou F při G95)

Maximální programovatelný posuv max pro maximální posuvovou rychlost $V_{\max}$ a zadané otáčky vřetene S se určí ze vztahu :

$$\beta_{\max} = V_{\max} / S$$

a podobně maximální programovatelné otáčky $S_{\max}$ pro danou maximální posuvovou rychlost $V_{\max}$ a zadaný posuv se určí ze vztahu:

$$S_{\max} = V_{\max} / \beta_{\max}$$

Příklad :

$$V_{\max} = 2000 \text{ mm/min}$$

$$S = 1000 \text{ ot/min}$$

$$\beta_{\max} = ?$$

$$\beta_{\max} = V_{\max} / S = 2000 / 1000 = 2 \text{ mm/ot}$$

Pro rotační souřadnice, kde je rychlost zadávána v tisících pulsů rotačního snímače za minutu (stupních/min), vypočteme rychlost v milimetrech/min na poloměru R [mm] od osy otáčení v případě, že se souřadnice pohybuje sama podle vzorce :

$$F_{sk} = F \cdot 2\pi R / \text{počet tisíců pulsů na otáčku}$$

Při programování funkce M36 (ze skupiny M4) je skutečný posuv roven programovanému. Při programování funkce M37 je skutečný posuv zmenšen v poměru 1: 100 oproti programovanému.

6.1 Konstantní řezná rychlost (KŘR) G96 a G97

Konstantní řezná rychlost je způsob řízení otáček vřetene, kdy se ze zadané velikosti řezné rychlosti a okamžité polohy řídící souřadnice, neustále vypočítávají požadované otáčky vřetene.

Ve strojní konstantě **R67** se určí, která souřadnice je řídící pro konstantní řeznou rychlost. V blocích s KŘR rychlostí musí být nulový bod programu umístěn pro řídící souřadnici do osy otáčení vřetene. Protože se během bloku nemůže změnit převodový stupeň, je nutno jeho volbě věnovat pozornost předem.

Končí-li konst. řezná rychlost a v následujícím bloku není programováno S, dosadí se naposled dosažené otáčky do adresy S. Při přechodu z otáčkového na minutový posuv se toto pro adresu F neprovádí.

Systém používá pro KŘR dvě funkce:

G96 – Konstantní řezná rychlost s posuvem mm/min

G97 – Konstantní řezná rychlost s posuvem mm/ot

Řezná rychlost se zadává funkcí S, která má v tomto případě význam nikoli otáček, ale řezné rychlosti v desetinách metru/min. Například řezná rychlost 50m/min by se zadala hodnotou S500.

Příklad: G97 S500

Protože u většiny strojů, především soustruhů a karuselů, je zvykem zadávat konstantní řeznou rychlost s posuvem na otáčku funkcí **G96** a řeznou rychlost funkcí S v m/min, může se ve strojních konstantách nastavit prohození funkcí **G96**, **G97** a přepočet konstantní řezné rychlosti S. Tyto přepočty provede systém automaticky a programátor zapisuje řeznou rychlost v m/min.

Příklad:

```
N10 G0 X300 Z100 M44 M3 S100 "rychloposuv na průměr 300, otáčky 100ot/min
N15 Z-1
N20 G96 G1 F200 S90 X50 "zařadí KŘR 90m/min, sjetí na průměr 50, posuv
"F = 0.2mm/ot
```

Při programování rychloposuvu se mění otáčky rovněž podle průměru, pokud není funkce G96 odvolána programováním G94.

Pozn.1

Pokud by nebyla nastavená strojní konstanta č. 329 dle doporučení, uvedeného výše, musel by se blok N20 naprogramovat takto:

```
N20 G97 G1 F200 S900 X50 „KŘR 90m/min, sjetí na průměr 50, F = 0.2mm/ot
```

Pozn.2

U starších verzí byla pro záměnu G96/G97 a programování S v m/min používána konverze přes program KONV836.EXE, jehož vyvolání se nastavilo v CNC836.KNF, konstanta \$50.

6.1.1 Konfigurace konstantní řezné rychlosti

Strojní konstanta R67 – řídící souřadnice

Kód řídící souřadnice pro konstantní řeznou rychlost. Kód souřadnice (viz parametr R00 - R05), ve které se mění řezný poloměr, se zadá do parametru R67. Není-li konstantní řezná rychlost používána, nelze zadat 0, ale je nutno zadat kód libovolné z použitých souřadnic!

3. dekáda strojní konstanty R329 – záměna funkcí a přepočít.

3. dekáda R329	0	Standard
	1	Záměna funkcí G96 za G97
	2	Zadání konstantní řezné rychlosti (S) přímo v m/min
	3	Záměna funkcí G96 za G97 a zadání konstantní řezné rychlosti (S) přímo v m/min

Doporučené nastavení strojní konstanty pro zadávání KŘR s posuvem na otáčku G96 a S v m/min:
Strojní konstanta číslo **R329**, třetí dekáda = **3**.

Strojní konstanta R580 – verze konstantní řezné rychlosti

1,2. dekáda R580	00	Starší verze KŘR. KŘR platí jen v pohybových blocích a není možnost řízení z PLC programu. Také systém nepodporuje omezení otáček na zadanou hodnotu.
	01	Novější verze KŘR, kterou možno nastavit od verze software primárního procesoru 40.44 a sekundárního procesoru 6.369. KŘR není omezena na pohybové bloky a také ji možno plně řídit z PLC programu. Systém omezuje otáčky podle zadané hodnoty (viz dále).

6.1.2 Konstantní řezná rychlost – verze 2

KŘR není omezena na pohybové bloky a také ji možno plně řídit z PLC programu. Systém i PLC mají možnost omezit otáčky vřetene na zadanou hodnotu. KŘR verze 2, je do značné míry kompatibilní se starší verzí, ale poskytuje mnohé vlastnosti navíc. Pokud nebude výslovně upozorněno, platí pro novější KŘR vše, co bylo napsáno v předcházející části. V dalším popisu se zaměříme jen na nové vlastnosti KŘR verze 2.

Novější verzi KŘR možno nastavit od verze software primárního procesoru 40.44 a sekundárního procesoru 6.369. Aktivace se provede nastavením hodnoty **1** do strojní konstanty **R580**.

Pro obvodovou rychlost platí:

$$v = \omega \cdot r = 2 \cdot \pi \cdot n \cdot x$$

x = poloha řídicí souřadnice (korigována vzhledem ke korekcím a posunutí)
v = obvodová rychlost (KŘR)
n = otáčky vřetene

Vypočtené napětí, které se zadává pro vřeteno (rozsah 0-7FFFh)

$$U_x = \frac{v}{2 \cdot \pi \cdot x} \cdot U \cdot U_m$$

U_x = vysílané napětí na vřeteno (rozsah 0-7FFFh)
U = maximální napětí pro vřeteno pro daný převodový stupeň (poměr k max.napětí)
 U_m = maximální napětí (7FFFh)
P = maximální otáčky dané převodové řady

6.1.3 KŘR-verze 2, řízení z NC programu

Řízení z NC programu je stejné jako pro starší KŘR a platí také stejné modifikace pomocí strojních konstant **R67** a **R329**. Nová vlastnost je, že KŘR není omezena jen na pohybové bloky NC programu. Zařazení KŘR může proto být i v nepohybovém bloku. Také změna délkové korekce a změna posunutí počátku se uplatní okamžitě a nemusí se čekat na pohybový blok. Korekce a posunutí mají vliv na KŘR, protože KŘR se počítá na špičku nástroje a ne na suport (tuto vlastnost možno vypnout – viz dále). KŘR se může také uplatňovat v pomocných ručních pojezdech nebo v jiných druzích pohybu, například vlečení nebo pro pohyby řízené z PLC programu.

Další nová vlastnost je, že aktivní KŘR nikdy nepřepíná převodový stupeň, ale omezí otáčky na maximální otáčky daného převodového stupně (pokud není ještě jiné omezení otáček).

Z NC programu možno kdykoli zadat omezení otáček pro KŘR pomocí funkce **G76** a adresy **S**. Pod adresou **S** se zadávají maximální otáčky v ot/min. Systém omezuje otáčky vzhledem k menší hodnotě z maximálních otáček převodového stupně a ze zadaného omezení pomocí funkce **G76**.

Příklad 1:

```
N100 G96 S500      "zadání KŘR – okamžitý účinek na otáčky vřetene
...
N200 S600          "změna KŘR
```

Příklad 2:

```
N100 G76 S2000     "zadání omezení otáček pro KŘR na 2000 ot/min
```

6.1.4 KŘR-verze 2, řízení z PLC programu

PLC program má k dispozici pole double-wordových hodnot pro sledování stavu KŘR. Na systému je možno jednotlivé hodnoty sledovat v prohlížení paměti systému – paměťová oblast 1.

název	adresa	Popis
AKRR_ACT_V	A178h	Aktuální konstantní řezná rychlost [mm/min] (G96 nebo PLC)
AKRR_ACT_SMAX	A17Ch	Aktuální maximální otáčky [1/1000 ot/min] (G76 nebo PLC)
AKRR_ACT_DIST	A180h	Aktuální korekce poloměru [1/8 μm] (jen PLC)
AKRR_ACT_VCORR	A184h	Aktuální korekce řezné rychlosti [mm/min] (jen PLC)
AKRR_ACT_R	A188h	Aktuální poloměr [1/8μm]
AKRR_ACT_P	A18Ch	Aktuální max.otáčky dle převodové řady [1/1000 ot/min]
AKRR_ACT_U	A190h	Aktuální rozsah napětí dle převodové řady [0-7FFFh]
AKRR_ACT_S	A194h	Aktuální vypočtené otáčky [1/1000 ot/min]
AKRR_ACT_OUT	A198h	Aktuální vypočtené napětí [0-7FFFh]

PLC program má možnost zadávat velikost KŘR, maximální otáčky, korekci poloměru pro výpočet KŘR a korekci řezné rychlosti. Také může KŘR aktivovat a případně modifikovat výpočet. Pro řízení PLC program používá bity umístěné v řídicím bajtu **AKRR_CNTR**.

Název bitu	Váha		popis
EXT_AKRR_V	0	0	Velikost KŘR se řídí z NC programu (G96,G97+ S)
		1	Velikost KŘR zadává PLC program v buňce AKRR_V
EXT_AKRR_SMAX	1	0	Maximální otáčky zadává NC program (G76)
		1	Maximální otáčky zadává PLC program v buňce AKRR_SMAX
EXT_AKRR_DIST	2	0	PLC program nezadává korekci poloměru
		1	PLC program zadává korekci poloměru v buňce AKRR_DIST
EXT_AKRR_VCORR	3	0	PLC program nezadává korekci řezné rychlosti
		1	PLC program zadává korekci řezné rychlosti v AKRR_VCORR
EXT_AKRR_REQ	4	0	Aktivace KŘR se řídí z NC programu (G96,G97)
		1	Aktivace KŘR z PLC programu
EXT_AKRR_SUPPORT	5	0	Standardní výpočet KŘR
		1	Výpočet KŘR nezapočítává korekce a posunutí

název	adresa	Popis
AKRR_V	A19Ch	Velikost KŘR zadávaná z PLC [mm/min]
AKRR_SMAX	A1A0h	Maximální otáčky zadávané z PLC [1/1000 ot/min]
AKRR_DIST	A1A4h	Korekce poloměru zadávaná z PLC [1/8 μm]
AKRR_VCORR	A1A8h	Korekce řezné rychlosti zadávaná z PLC [mm/min]

7

7. KOREKCE NÁSTROJE

Pozn.: Platí od systémové verze s datem 20.10.1997 a pozdějším.
Vkládání kroužků při poloměrové korekci platí od verze panelu 30.27 (10.11.2000).

Korekce nástroje umožňují vytvořit obecný partprogram, který je použitelný pro různé průměry a délky nástrojů (fréz, soustružnických nožů apod.). Při výměně nástroje s jinými rozměry se upraví pouze příslušná korekce a partprogram zůstane beze změny.

Korekce rozeznáváme dvojího druhu:

Korekce na poloměr nástroje - je určena G-funkcí ze 3. skupiny (G41, G42 a G40)
Korekce délkové - jsou určeny adresou &

Tato kapitola pojednává o poloměrových korekcích.

7.1 Soubor TAB0.KOR a tabulka korekcí v paměti

Pozn.: Název souboru, resp. číslo za TAB není závazné. Obecně se může používat více tabulek. V tomto návodu budeme používat název TAB0.KOR.

Hodnoty korekcí (poloměrové i délkových) jsou uloženy v souboru TAB0.KOR. Systém při své činnosti pracuje s kopií tohoto souboru, uložené v tabulce korekcí v **systémové vnitřní paměti**. **Do této tabulky v paměti se soubor TAB0.KOR přepíše automaticky vždy po zapnutí systému a dále po každé edici souboru TAB0.KOR, pokud se po ukončení edice provede jeho uložení** (viz. „Návod k obsluze CNC836“).

Systémová tabulka korekcí ve vnitřní paměti má následující strukturu:

Čís. Korekce	I. údaj	II. údaj	III. údaj	IV. údaj	V. údaj
1	poloměrová korekce	délková pro první osu	délková pro druhou osu	Délková pro třetí osu	Délková pro čtvrtou osu
2	poloměrová korekce	délková X	délková Y	Délková Z	Délková 4.
atd. až do 99					
	..	..	..	..	..

Tabulka korekcí má celkem 99 položek, nebo-li systém umožňuje používat 99 různých korekcí na poloměr a délku nástroje. Každá položka tabulky obsahuje celkem 5 údajů. První údaj obsahuje hodnotu poloměrové korekce. Následující čtyři údaje jsou délkové korekce první, druhé, třetí a čtvrté osy. **Délkové korekce nelze použít pro eventuelní pátou a šestou osu.**

Korekce, uložené v souboru TAB0.KOR mají stejnou strukturu a jsou v souboru TAB0.KOR uloženy v dále popsaném tvaru. (Pozn.: Soubor začíná klíčovým slovem \$KOR, před ním může být libovolný komentář.)

Řádek začíná dvoumístným číslem korekce (je uvedena eventuelní nevýznamná nula), za kterým následuje dvojtečka. Hodnota poloměrové korekce je uvedena znakem R=, hodnoty délkových korekcí jsou uvedeny názvy souřadnic nebo pořadovým číslem souřadnic.

Pozn.:

Názvy souřadnic se používají, pokud jsou v pořadí první osa je X, druhá osa je Y, třetí osa je Z a čtvrtá osa je U. Pořadí a název souřadnic je určeno ve strojních konstantách č. 0 až 5 v souboru TAB0.REK. U soustruhů je obvykle první osa X a druhá osa Z. **V tomto případě se nesmí použít v souboru pro korekci osy Z znak Z, ale číslice 2 (druhá souřadnice).** Pro osu X je možno použít znak X (nebo číslici 1).

V následujícím příkladu souboru TAB0.KOR je uvedeno několik možných způsobů zápisu. Doporučený způsob zápisu v souboru TAB0.KOR pro tříosé frézky (X,Y,Z) je uveden u korekce číslo 1, doporučený způsob u soustruhů (X,Z) je uveden u korekce číslo 2. Pro čtyřosé stroje X,Y,Z,4 je doporučený způsob uveden u korekce číslo 3.

U korekce č. 25 není uvedena poloměrová korekce, proto bude v tabulce pro tuto poloměrovou korekci hodnota nula. U korekce číslo 95 je uveden zápis pomocí pořadových čísel souřadnic.

Pokud není příslušná korekce uvedena, bude její hodnota v tabulce (v paměti systému) nulová. V uvedeném příkladu budou nulové poloměrové i délkové korekce u všech zde neuvedených položek, t.j. kromě korekce č. 1,2,3,25 a 95 všechny ostatní. Nulové budou i korekce neuvedené pro jednotlivé osy.

\$KOR

01:	R=10.0	X=20.0	Y=30.0	Z=40.0	
02:	R=0.8	X=0.0	2=0.0		
03:	R=0.8	X=0.0	Y=120.0	Z=0.0	4=0.0
25:		X=85.0	Y=45.5	Z=0.0	
95:	R=0.0	1=12.55	2=0.0	3=0.0	4=0.0

\$KOR

01:	R=0.0	X=0.0	Y=0.0	Z=0.0	4=0.0
-----	-------	-------	-------	-------	-------

Vzorový soubor TAB0.KOR pro soustruhy obsahuje 99 položek s nulovými korekcemi ve tvaru:

\$KOR

01:	R=0.0	X=0.0	2=0.0
-----	-------	-------	-------

Pozn.1:

Pokud se používají u soustruhu poloměrové korekce, může být v každé položce uveden ještě typ použitého nástroje v tomto tvaru:

\$KOR

01:	R=0.0	X=0.0	2=0.0	P=3
-----	-------	-------	-------	-----

Číslo za P= musí být v rozsahu 1 až 9. Jiné hodnoty nejsou povoleny. Popis použití typu nástroje u soustruhů je uveden dále.

Pozn.2:

Soubory je možné kvůli přehlednosti zkrátit, pokud se nebude využívat všech 99 korekcí. Název souboru TAB0.KOR není závazný, pro soustruhy může být v systému uveden pod názvem TAB2.KOR. Důležité je, aby stejný název byl uveden také v konfiguračním souboru CNC836.KNF v parametru 21. Přestože systém umožňuje používat obecně několik různých souborů s korekcemi, doporučuje se kvůli jednoznačnosti používat (a mít v paměti) pouze jeden.

Způsoby plnění tabulky korekcí

Při zápisu (plnění) tabulky korekcí je třeba rozlišit zápis do souboru TAB0.KOR a zápis do kopie tabulky ve vnitřní paměti systému.

Tabulku korekcí ve vnitřní paměti systému je možné naplnit těmito způsoby:

- Editací souboru TAB0.KOR a **následném uložení** - popis editoru viz Návod k obsluze. Upravené hodnoty korekcí zůstanou v souboru trvale uloženy. Po každém zapnutí systému se přepíší do vnitřní paměti systému.
- Interaktivním zadáním v ručním režimu stiskem tlačítka R resp. D, resp. & - popis viz. Návod k obsluze. Korekce zadané interaktivním způsobem se zapíší jak do souboru TAB0.KOR, tak i do vnitřní paměti systému. Každým zapnutím systému se přepíší do vnitřní paměti systému.
- Plnění tabulky korekcí z partprogramu programováním funkce G92 - popis viz. Návod k programování.
POZOR: Tímto způsobem se naplní pouze tabulka ve vnitřní paměti systému. Hodnoty v souboru TAB0.KOR zůstanou beze změny. To znamená, že korekce zapsané do systémové tabulky v partprogramu pomocí funkce G92 platí pouze do chvíle, než se editací uloží soubor TAB0.KOR nebo do vypnutí systému. Po zapnutí platí hodnoty uvedené v souboru TAB0.KOR. Protože zápis do tabulky korekcí se provádí přímo v partprogramu, je vždy zaručeno, že pro daný partprogram budou vždy platit hodnoty korekcí, které se zde zapíší. Případná změna hodnot korekcí se ale musí v tomto případě provádět v partprogramu, nikoli v souboru TAB0.KOR !

Pro naplnění tabulky korekcí pomocí funkce G92 se využívá parametrů, jejichž hodnoty se přiřadí do dané položky tabulky korekcí takto:

hodnota parametru R0 je určena pro naplnění poloměrové korekce

hodnota parametru R1 je určena pro naplnění délkové korekce v první ose (obvykle X)

hodnota parametru R2 je určena pro naplnění délkové korekce v druhé ose (obvykle Y nebo Z u soustruhů)

hodnota parametru R3 je určena pro naplnění délkové korekce ve třetí ose (obvykle Z)

hodnota parametru R4 je určena pro naplnění délkové korekce ve 4. ose

Funkcí G92 se určí, že pokud je příslušný parametr v bloku programován, zapíše se jeho hodnota do příslušného údaje té položky tabulky korekcí, která je uvedena pod adresou D.

Příklad:

Chceme naplnit položku 12 tabulky korekcí těmito hodnotami:

poloměrovou korekcí 10.0 mm, délkovou korekcí v ose Y hodnotou 25,5 mm a délkovou korekcí v ose

Z hodnotou -5.0 mm. Blok partprogramu bude mít následující tvar:

N10 G92 D12 R0=10.0 R2=25.5 R3=-5.0

Pokud není některý parametr v bloku uveden, hodnota příslušného údaje v dané položce tabulky korekcí se nezmění. V našem příkladě se hodnota délkové korekce v ose X a 4. v položce 12 tabulky korekcí nezmění, t.j. zůstane tam původní hodnota.

7.2 Poloměrové korekce s ekvidistantou

Je-li nastavena strojní konstanta číslo 95, 8. dekáda na 1, je možné používat nový způsob řešení poloměrových korekcí. Může se (s výjimkami, uvedenými dále) programovat obrys obrobku neboli výkresové hodnoty. Korigovaná dráha (dráha středu nástroje u frézek, dráha středu poloměru bříty u soustruhů) leží na ekvidistantě. Při nespojitosti dvou po sobě následujících bloků dojíždí střed nástroje na průsečík ekvidistant nebo se vloží oblouk. Která z těchto možností se použije, záleží na nastavení strojních konstant (a versi – verze nižší než 30.27 neumí vkládat při nespojitosti oblouk).

Souhrnný přehled nastavení strojních konstant, týkajících se korekcí:

95 – 8 dekáda	0	Starší způsob poloměrových korekcí, popsáný v kapitole 7A (již se nedoporučuje používat – pouze kvůli kompatibilitě se staršími versemi)
95 – 8 dekáda	1	Doporučené nastavení. Poloměrové korekce s koncovými body na průsečíku ekvidistant nebo s vkládáním oblouků
339 – 8 dekáda	0	Poloměrové korekce s koncovými body na průsečíku ekvidistant. Nutno použít pro pravoúhlé stroje (stroje s přepínanými osami)
339 – 8 dekáda	1	Poloměrové korekce s vkládáním oblouku při nespojitosti, pokud je úhel větší než 180 stupňů plus tolerance zadaná v 1. až 6. dekadě této konstanty (možno použít až od verse 30.27)
339 – 7 dekáda	0	Vyřazení korekce funkcí G40 pouze v bloku, kde je programován pohyb alespoň v jedné z os korekční roviny (doporučený způsob)
339 – 7 dekáda	1	Povolení programovat funkci G40 i v nepohybových blocích (nedoporučený způsob - možno použít až od verse 30.27) – nutno počítat s tím, že se korekce odvolá v některém z následujících bloků, kde bude programován pohyb v korekční rovině. Do té doby nesmí být korekční rovina změněna.
339 - 1.až 6. dek		Zadá se úhel pro vkládání oblouků. Oblouk se vloží, pokud bude úhel mezi programovanými dráhami větší než zde zadaný.
39 - 1.až 8.dek		Toleranční úhel pro plynulou návaznost – související strojní konstanta. Její hodnota (úhel) je nastavena pro rozhodování systému o plynulém jetí. Hodnota ve strojní konstantě 339 (úhel) by měla být menší nebo maximálně rovná než je úhel zadaný v této konstantě

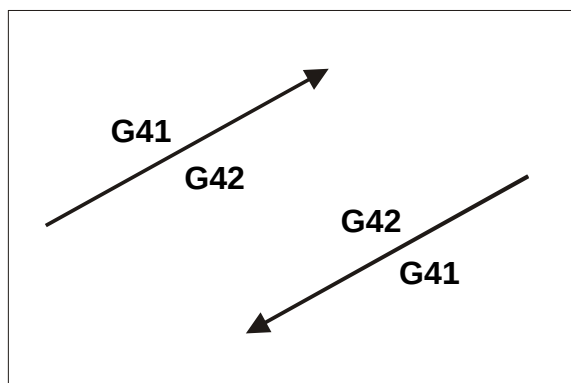
Význam těchto konstant je patrný též z následujícího textu a obrázků.

Korekce poloměrová je účinná na konci bloku, ve kterém byla vyvolána funkcí G41 nebo G42, t.j. začne platit v následujícím bloku. **Vyvolání (zařazení funkcí G41 nebo G42) a odvolání (vyřazení funkcí G40) poloměrové korekce může být pouze v bloku s lineární interpolací.** Nelze zařazovat a vyřazovat poloměrovou korekci na kružnici.

Pro programování poloměrové korekce G41, G42 platí následující pravidla:

Korekce G41/G42 je trvalá funkce a platí až do odvolání funkcí G40. Korekce je platná pro zvolenou korekční rovinu (G17,G18,G19). Pokud se korekční rovina explicitně nezvolí, platí prioritní korekční rovina G17 (rovina 1. a 2. osy – tj. obvykle XY pro frézky a XZ pro soustruhy). Pokud poloměrovou korekci používáme jen v těchto osách, nemusí se korekční rovina programovat. Eventuelní změna korekční roviny (funkce G17,G18, G19) nesmí být provedena při zařazení poloměrové korekci.

Korekce může trvat i v blocích bez pohybu v korekční rovině, neboli systém překlene až 200 bloků bez pohybu v korekční rovině a pohyb poté pokračuje, jakoby nepohybové bloky nebo pohyb v jiné než korekční rovině nebyly programované.



G41 je korekce vlevo, tj. střed nástroje se pohybuje (pokud je poloměrová korekce kladné číslo) vlevo od programované dráhy ve směru pohybu.

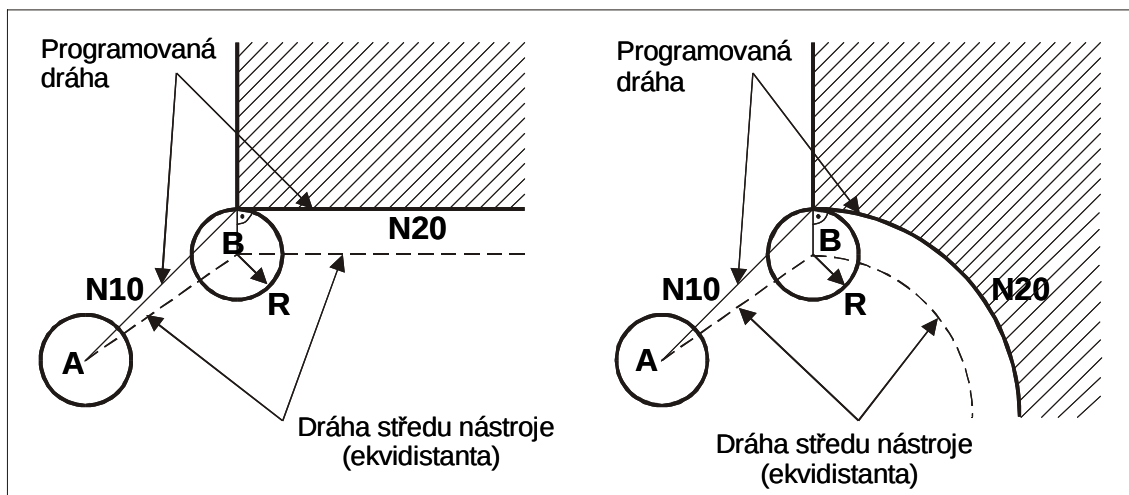
G42 je korekce vpravo, tj. střed nástroje se pohybuje (pokud je poloměrová korekce kladné číslo) vpravo od programované dráhy ve směru pohybu.

Pokud by hodnota korekce bylo záporné číslo, prohodily by se strany korekcí, tj. G41 se zápornou korekcí je to samé jako G42 s kladnou korekcí.

7.2.1 Řazení poloměrové korekce a průběh korekce

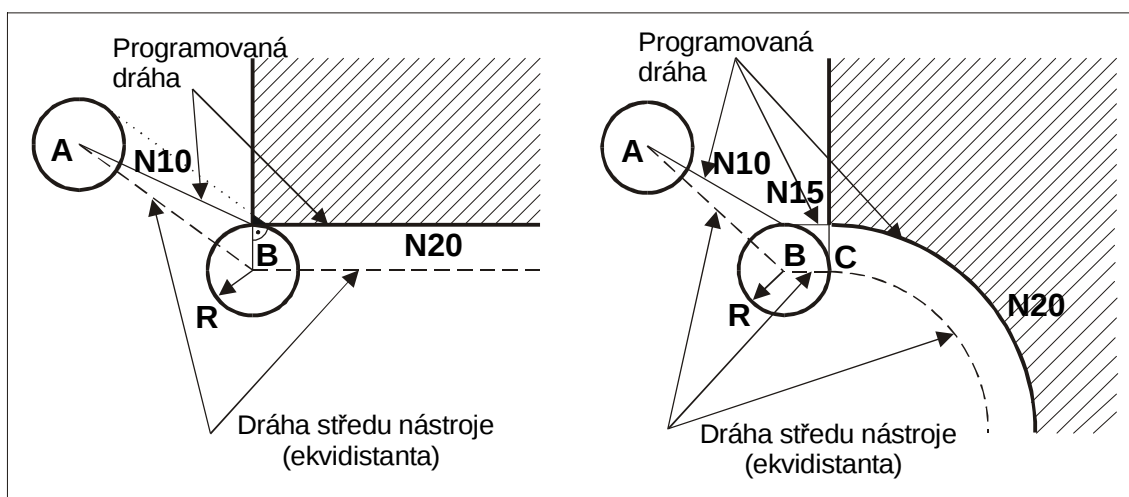
Při zařazování poloměrové korekce platí jednoznačně následující pravidlo:

Korekci lze zařadit pouze při lineární interpolaci. Dráha středu nástroje se pohybuje z počátečního bodu bloku N na kolmici k dráze bloku N+1, t.j. blok N+1 se celý provede již se zařazenou korekcí. Na obr. 1 je v bloku N10 zařazena poloměrová korekce G42, korigovaná dráha je znázorněna čárkovaně z bodu A do bodu B. Bod B, na který najíždí, leží na kolmici k dráze bloku N20. Stejným způsobem se určí bod pro nasazení korekce v případě, že následuje kružnice, jak je uvedeno na tomtéž obrázku vpravo. Bod B rovněž leží na kolmici k tečně k programované kružnici.



Obr.1

Při zařazování korekce je třeba zvolit správný směr nájezdu. Na obr. 2 vlevo je uveden chybný úhel nájezdu. Při tomto směru nájezdu na korekci by došlo k chybnému obrobení rohu, neboť obrys nástroje protne obrobek předtím, než najede na kolmici k dalšímu bloku.

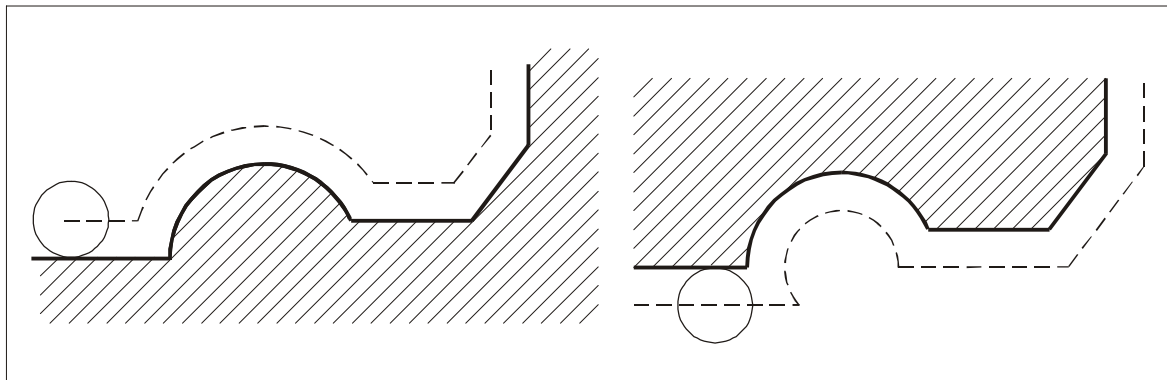


Obr.2

Vhodnější je tedy nasadit korekci ještě před materiálem, jak je uvedeno na obr. 2 vpravo. Pokud se korekce nasadí ještě před obrobkem, což je obvyklý případ, tak na úhlu nájezdu nezáleží. Nasazení korekce skončí v bodu B, do bodu C (blok N15) již program jede se zařazenou korekcí.

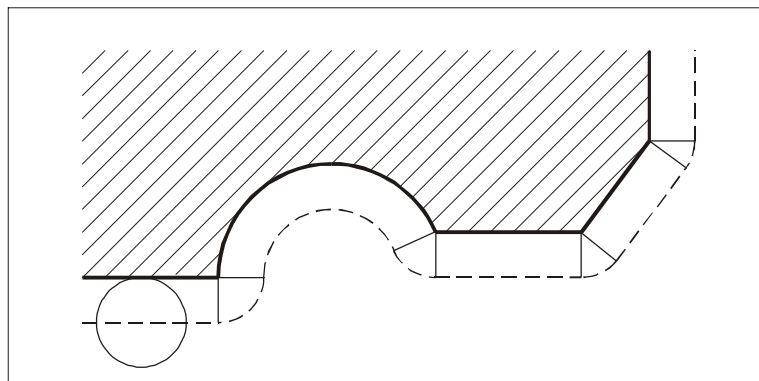
Je-li korekce zařazena, pohybuje se střed nástroje po ekvidistantách (označeny čárkovaně) vzdálených o poloměr nástroje od programované dráhy. Na obr.3 jsou uvedeny příklady korigované dráhy pro korekci vlevo i vpravo.

Takto bude provedena poloměrová korekce ve verzích nižších než 30.27 a u verzí 30.27 a vyšších v případě, že bude nastavena strojní konstanta 339, 8 dekáda na 0.



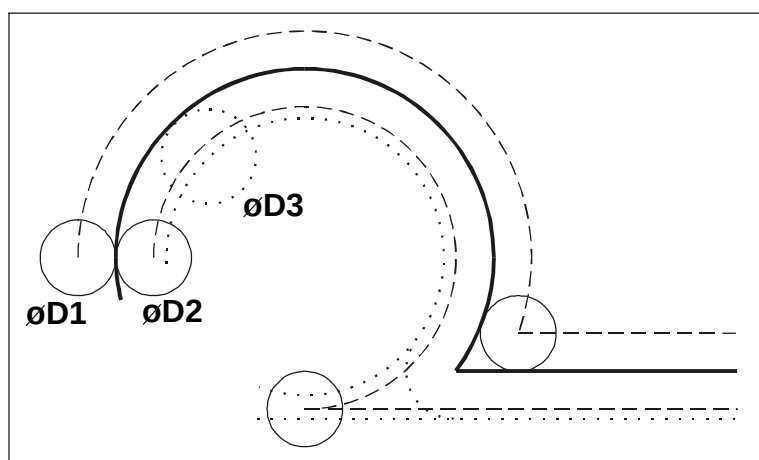
Obr.3

U verzí 30.27 a vyšších v případě, že bude nastavena strojní konstanta 339, 8 dekáda na 1, se budou vkládat při nespojitosti oblouky, jak je uvedeno na obr.4.



Obr.4

Oblouky se vloží, pokud je úhel větší než 180 stupňů (plus toleranční úhel, nastavený ve strojní konstantě 339 – viz dále). Pro menší úhly pojede korigovaná dráha na průsečíky ekvidistant, tj. stejným způsobem, jak je uvedeno na obr. 3 vlevo, nebo-li pro úhly menší než 180 stupňů je průběh korekce stejný bez ohledu na nastavení 8. dekády strojní konstanty č.339.



Obr.5

Pokud jezdí střed nástroje na průsečíky ekvidistant (339, 8dek.=0), musí se při programování dbát, aby vždy existoval průsečík ekvidistant. Neprotnou-li se ekvidistanty dvou po sobě následujících bloků, hlásí systém chybu „NENALEZEN PRUSEČÍK EKVIDISTANT“. Tento případ může nastat při nevhodné kombinaci velikosti korekce a úhlu tečen v bodě, kde se protíná dráha dvou po sobě následujících bloků. Na obr. 5 je uveden příklad při programování kružnice a přímky.

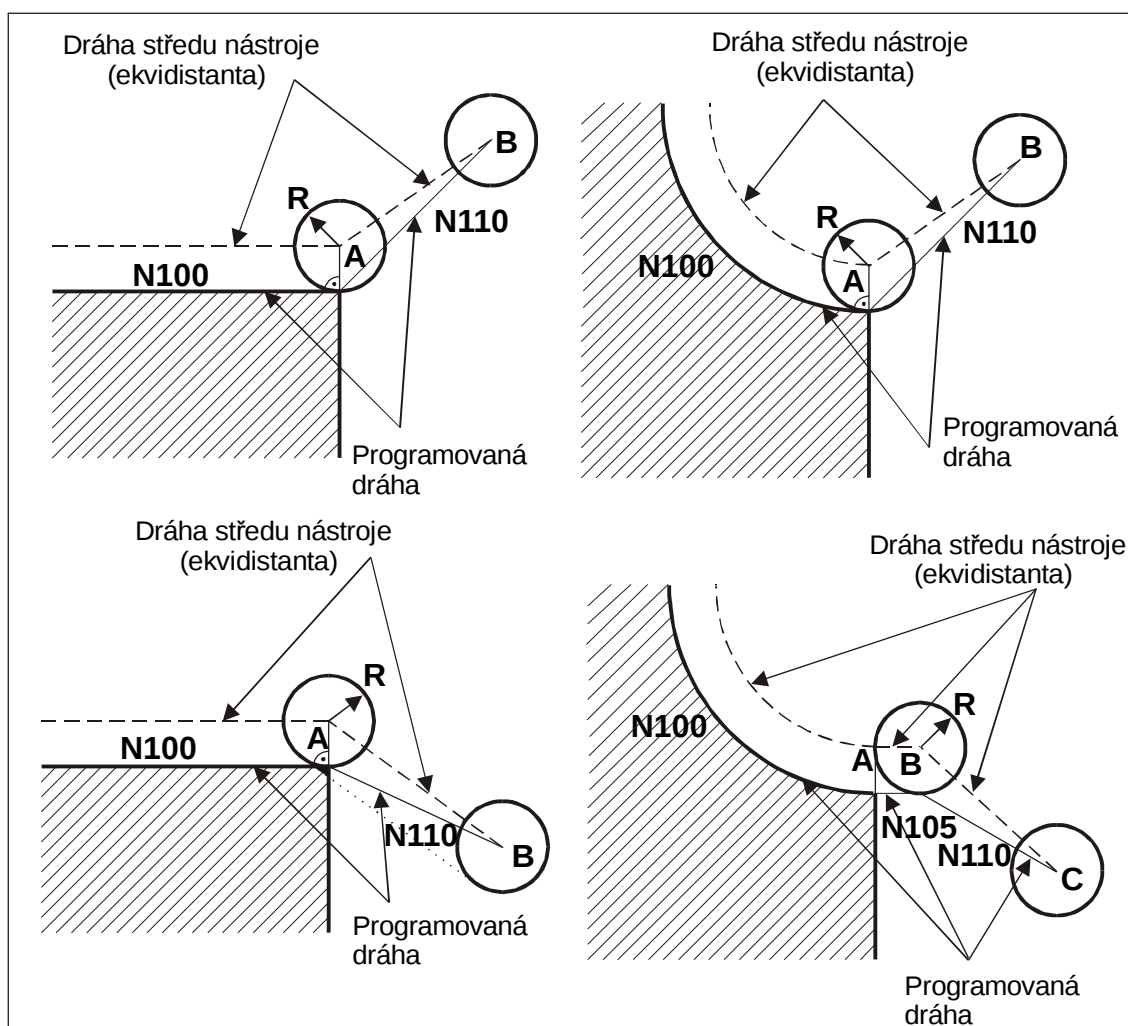
Při korekci vlevo G41 – nástroj o průměru $D1$ - nenastanou žádné problémy, ekvidistanty se vždy protnou. Při korekci vpravo G42 a nástroji o průměru $D2$ se ekvidistanty rovněž protnou i když nástroj již zajiždí relativně daleko. Při korekci vpravo G42 a průměru nástroje $D3$ se ovšem ekvidistanty již neprotnou (dráha na obr. 5 je vyznačena tečkovaně) a systém hlásí uvedenou chybu. **U nových verzí a při nastavené strojní konstantě (339, 8dek=1) se však vloží v uvedeném případě oblouk (rovněž vyznačen tečkovaně). Vkládání oblouků jednoznačně řeší tyto případy, kromě toho je i dráha nástroje kratší čímž se zkracuje čas obrábění. U nových verzí se doporučuje používat tento způsob.**

Pozn.:

V některých případech (pokud se nekládají oblouky) je sice nalezen průsečík ekvidistant, ale leží daleko. Obvykle se jedná o technologicky nevhodné případy, které se v praxi nevyskytují. Pokud se vyskytnou, je třeba zvolit vhodnější způsob programování

7.2.2 Vyřazení poloměrové korekce

Odvolání poloměrové korekce se provede funkcí G40 a je možné, stejně jako řazení G41, G42, pouze při lineární interpolaci. Koncový korigovaný bod posledního bloku před G40 leží na kolmici k tečně, která prochází koncovým programovým. Odvolání korekce je uvedeno na obr.6. Stejně jako u zařazování korekce se musí dbát na směr bloku, ve kterém je programována funkce G40. Neplatí to pro případ, že korekci odvoláváme již mimo obrobek.



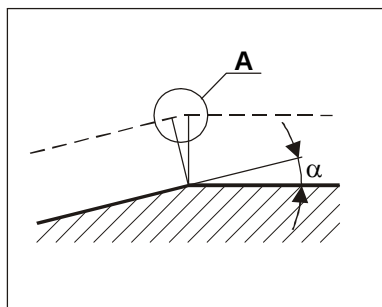
Obr.6

Na obr.6 je odvolání korekce G40 programováno v pohybovém bloku (tj. je programován pohyb alespoň v jedné z os korekční roviny). Nastavením strojní konst. 339 (7. dekáda = 1) lze programovat G40 i v nepohybovém bloku. Tento způsob se však nedoporučuje používat.

7.2.3 Limitní úhel pro vkládání oblouků

Platí pouze pro verze 30.27 a vyšší

Ve strojní konstantě 339 je možné zadat limitní úhel pro vkládání oblouků při poloměrové korekci. Použití



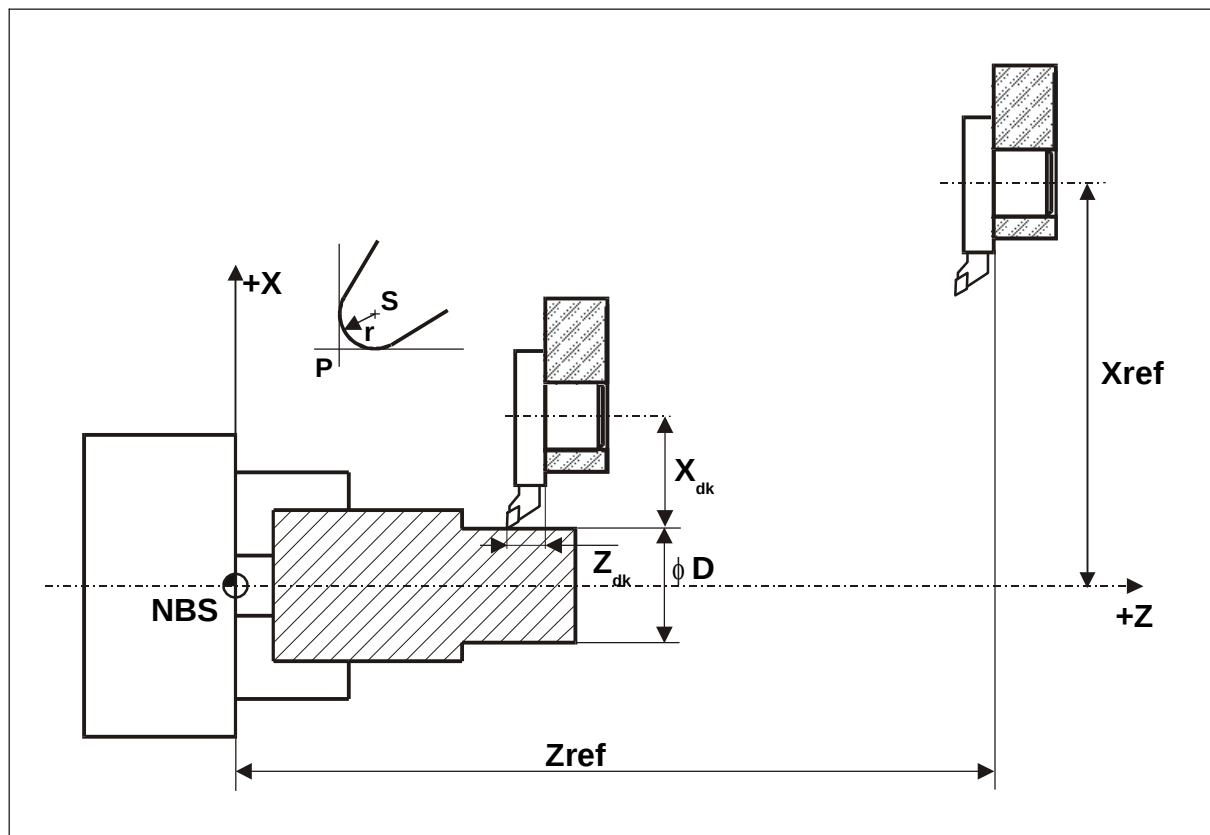
vyplyne z příkladu – dráha středu nástroje je vyznačena čárkovaně. Ve strojní konstantě 339 zadáme limitní úhel α_{limit} např. 15 stupňů. Potom v případě (viz obr. vlevo), že úhel $\alpha > \alpha_{\text{limit}}$, vloží se při nespojitosti oblouk (detail A), v případě, že úhel $\alpha \leq \alpha_{\text{limit}}$, oblouk se nevloží, ale pojede se na průsečík ekvidistant.

Nastavení strojní konstanty 339 úzce souvisí se strojní konstantou 39, kde se zadává toleranční úhel pro plynulou návaznost. Pokud je úhel dráhy mezi dvěma bloky menší než zde zadaný limit, považuje se dráha za spojitou a projede se (při programování G23) beze změny rychlosti. Limitní úhel pro vkládání oblouků při poloměrové korekci, zadaný ve strojní konstantě 339 by proto měl být menší nebo maximálně rovný úhlu, zadanému ve strojní konstantě 39. Pokud by byl limitní úhel

pro vkládání oblouku (konst. 339) větší než limitní úhel pro plynulou návaznost (konst.39), existovala by oblast úhlu návaznosti bloku, při které by byl zakázán plynulý přejezd (systém by zpomalil a zase se rozjel).

7.3 Poloměrové korekce u soustruhů

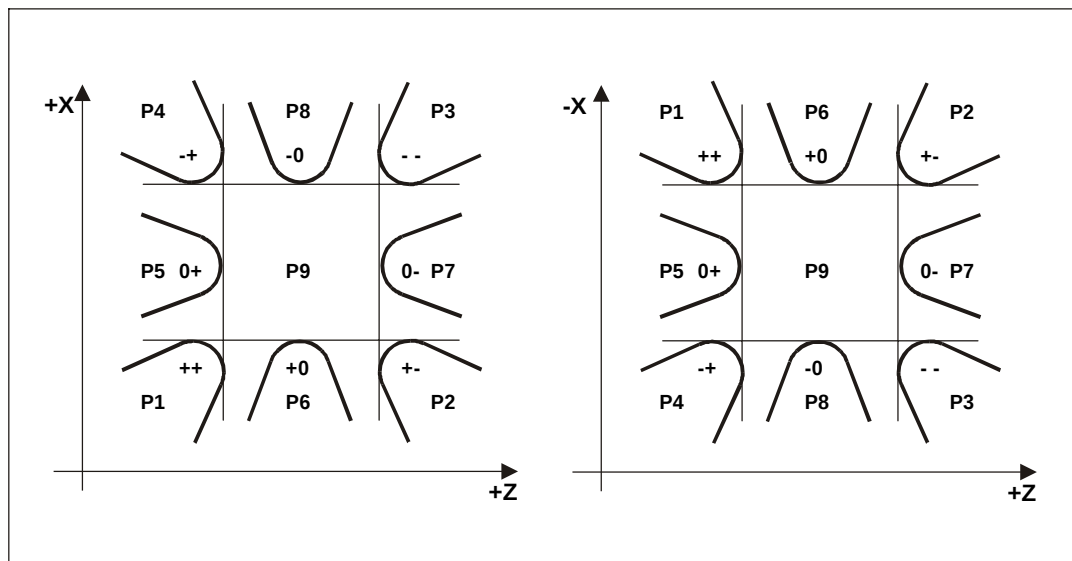
Při využívání poloměrové korekce může být programována kontura dílce (výkresové hodnoty). Při změně nástroje (poloměru břitu) se změní pouze hodnota poloměrové korekce v tabulce.



Obr.7

Při zadávání délkové korekce u soustruhů se tato korekce vztahuje k teoretickému bodu břitu P. Tento bod neleží (obvykle) ve středu břitu, ale v průsečíku tečen k poloměru břitu r , jak je uvedeno na obr. 7.

Aby systém mohl správně vypočítat projížděnou dráhu po ekvidistantě, musí být kromě poloměru bříty zadána i jeho poloha. Pro výpočet poloměrové korekce pak systém použije kromě údaje o poloměru nástroje (bříty) také údaj o tom, jak je nůž upnut v držáku. Možné polohy nože jsou uvedeny na obr. 8.



Obr.8

Pozn.:

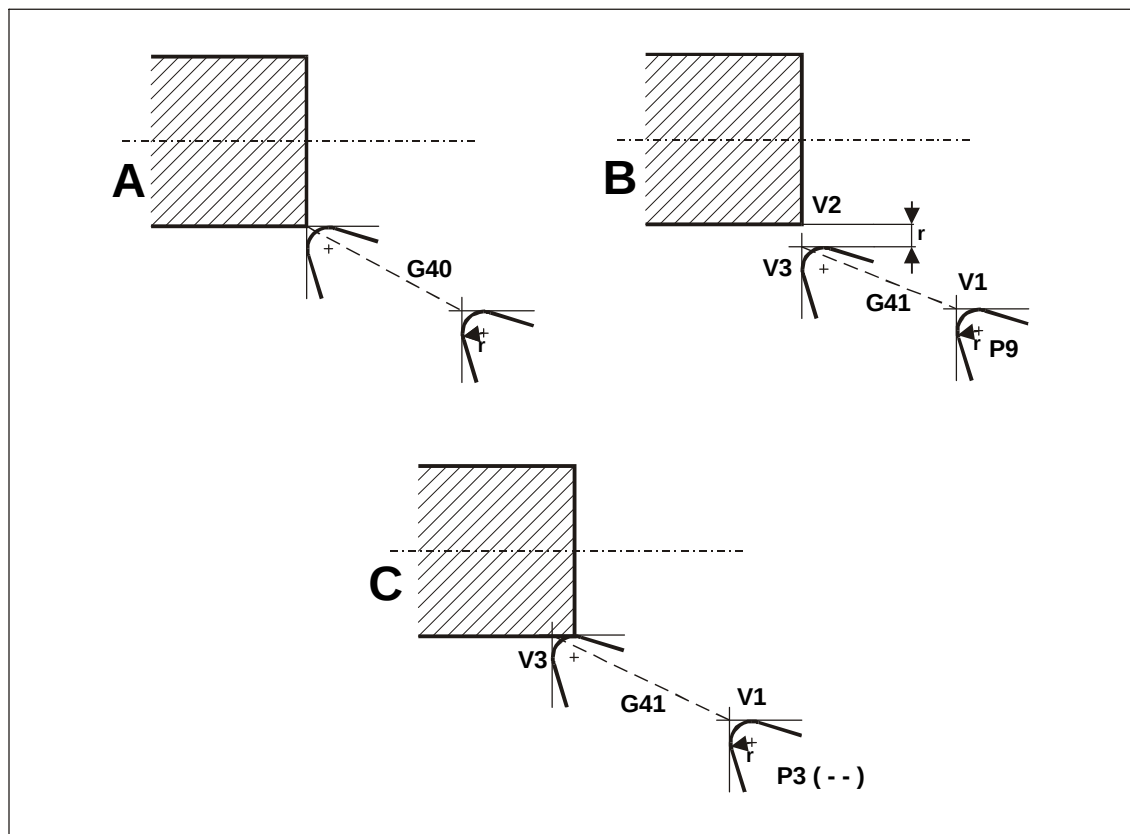
Oba obrázky jsou totožné, pouze zrcadlově otočené podle osy Z. Pro stanovení polohy nástroje se vybere ten, u kterého souhlasí směry osy X s konkrétním soustruhem. Znaménka uvedená u jednotlivých poloh bříty jsou zapsána v konfiguračním souboru CNC836.KNF v parametrech \$51 (pro první nástrojovou hlavu) a \$52 (pro eventuální druhou nástrojovou hlavu). Pokud má soustruh pouze jednu nástrojovou hlavu, na nastavení parametru \$52 nezáleží (uvedou se zde např. stejné hodnoty jako pro první nástrojovou hlavu). Znaménka určují směr aditivního posunutí teoretické špičky bříty nástroje vůči středu poloměru bříty.

Vliv aditivního posunutí, které zohledňuje použitou polohu nástroje, je uveden na obr.9. Na obr.9A je uvedena dráha bez použití poloměrové korekce. Teoretická špička bříty nástroje se pohybuje po dráze, kterou naprogramuje technolog - v tomto případě je dráha naprogramována na hranu obrobku. Pokud by programované dráhy byly pouze rovnoběžné se souřadným systémem, tj. rovnoběžné pouze s osami X a Z, nemusela by se korekce programovat resp. dráha s korekcí i bez korekce by byla totožná.

Na obr.9B je použita poloměrová korekce vlevo G41, ale v tabulce korekcí není zadán typ nástroje (není uvedeno P) nebo je uvedeno P=9. Nástroj typu P9 má teoretickou špičku ve středu bříty, proto má aditivní posunutí v obou směrech nulové. Jak je vidět z obrázku, korekce G41 se zařadí (špička se pohybuje z bodu V1 do bodu V3), ale protože koncový bod V3 není zkorrigován o aditivní posunutí, nástroj by se pohyboval mimo obrobek.

Na obr.9C je uvedeno správné použití poloměrové korekce. V tabulce korekcí musí být u dané poloměrové korekce uveden typ nástroje - v našem případě P=3, který má podle obr.8 aditivní posunutí v ose X mínus a v ose Z také mínus. O velikost poloměru s ohledem na uvedená znaménka se posune teoretická špička V3 z obrázku 9B do polohy V3 na obr.9C, což je správná poloha bříty vzhledem k obrobku. Dráha teoretické špičky je uvedena na obr. 9C čárkovaně z bodu V1 do bodu V3.

Použití poloměrové korekce se projeví především u dráhy, která není rovnoběžná ani s jednou osou.



Obr.9

7.4 Indikace u soustruhu při použití poloměrové korekce

Indikace polohy na systému zahrnuje aditivní posunutí. Jaké bude systém indikovat hodnoty v jednotlivých blocích si uvedeme na příkladu části partprogramu.

Obsah souboru s korekcemi TAB0.KOR bude následující:

\$KOR

01: R=0.8 X=120.0 Z=340.5 P=3

....

....

V souboru je pro korekci číslo D1 zapsána poloměrová korekce 0.8mm. Předpokládáme, že délkové korekce jsou nastavené pro nástroj číslo T1, poloha břitu podle obr.8 je P=3. Souřadná soustava obvyklá u soustruhů :

Vpravo Z +

Nahoru X -

Část partprogramu bude mít následující tvar (předpokládáme průměrové programování v ose X):

%1

N5 G90 G54 G40 &1100 D1 T1

N10 X 20 Z50

N20 X0 Z0 G41

N30 X20

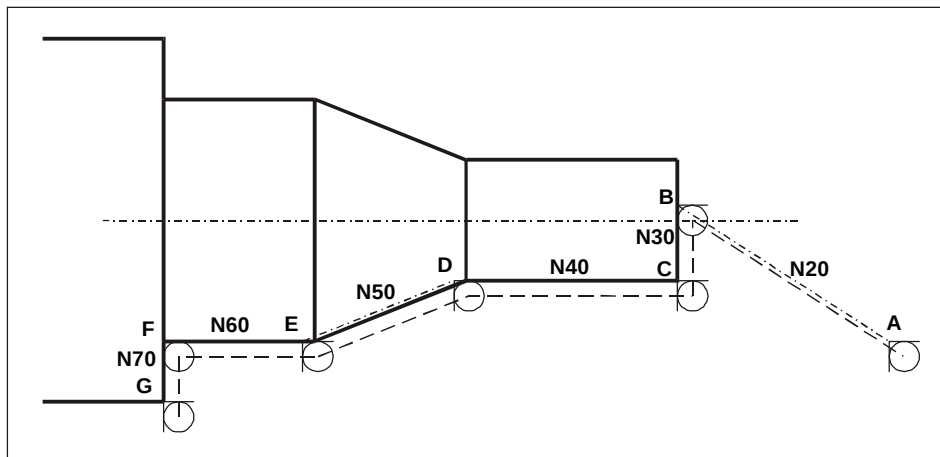
„VOLBA NÁSTROJE A KOREKČNÍ TABULKY

„NAJETÍ VÝCHOZÍ POLOHY

„ZAŘAZENÍ KOREKCE

N40 Z-50
 N50 X 40 Z-75
 N60 Z-100
 N70 X50
 N80 X60 G40

„POSLEDNÍ BLOK S KOREKČÍ
 „ODVOLÁNÍ KOREKCE



Obr.10

Na obrázku 10 je nakreslena dráha bříty dle uvedeného partprogramu (poloměr bříty je pro přehlednost mnohonásobně zvětšen).

Indikace na koncích jednotlivých bloků je uvedena v tabulce:

KONEC BLOKU	TEORETECKÁ SPIČKA V BODĚ	INDIKACE X (PRŮMĚROVĚ)	INDIKACE Z
N20	B	-1.6	0.0
N30	C	+20.0	0.0
N40	D	+20.0	-50.646
N50	E	+40.0	-75.646
N60	F	+40.0	-100.0
N70	G	+48.4	-100.0
N80		+60.0	-100.0

V bloku N70, který je posledním blokem před odvoláním korekce leží střed bříty na kolmici ke koncovému bodu a teoretická špička je o poloměr blíže k ose. Hodnota 48.4 je průměr 50 mínus 2 krát poloměr bříty 0.8. (Indikace je průměrová!)

Pozn.:

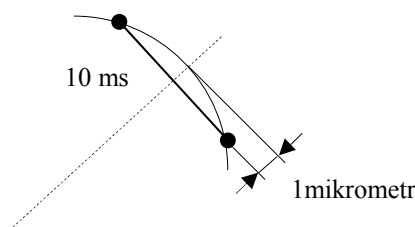
Při grafickém znázornění dráhy se kreslí teoretická špička bříty. Kromě bloku N20 a N50 (na obr.10 vyznačeny čárkovaně mezi body A-B a D-E) se korigovaná dráha kryje s programovanou.

7.5 Řízení rychlosti při poloměrových korekcích

Systém řídí rychlost jednotlivých souřadnic tak, že programovaná rychlost v bloku určuje prostorový vektor, kterého směr je vždy tečnou k dané trajektorii a jeho složky do souřadnicového systému určují rychlosti v jednotlivých osách.

Programovaná rychlost v bloku je rovna výsledné tečnové rychlosti, čím jsou zabezpečeny konstantní řezné podmínky pro obráběný materiál. Změny skutečné tečnové rychlosti od programované rychlosti mohou být způsobeny těmito faktory:

- Změna rychlosti s ohledem na zadané procento rychlosti (%F).
- Možnost zmenšení rychlosti při kruhové interpolaci s ohledem na přesnost od tvaru ideální kruhovitosti. Omezení rychlosti se projevuje na malých kruzích. Výpočtový takt ininterpolátoru systému je 10ms a proto je vlastně kruhová interpolace složená s lineárními úseky pohybu trvajících 10 ms. Systém standardně omezuje rychlost při kruhové interpolaci tak, aby byla odchylka od ideální kruhovitosti menší než 1 mikrometr. Pokud to dynamika stroje dovolí a není potřeba dodržet odchylku od ideální kruhovitosti 1 mikrometr, je možno řídit omezení rychlosti na kruhové interpolaci pomocí strojní konstanty R232 (viz příloha F).



- Při použití poloměrových korekcí systém standardně ovlivní skutečnou tečnovou rychlost tak, aby rychlost v bodu dotyku nástroje a materiálu byla ta, která je naprogramována v bloku. Tímto způsobem jsou zabezpečeny konstantní řezné podmínky. Ovlivnění rychlosti dochází jen při kruhové interpolaci. Skutečná výsledná rychlost je větší nebo menší v poměru aktuální poloměrové korekce a poloměru kružnice. Systém vlastně dodržuje konstantní úhlovou rychlost.

poloměr kružnice bez poloměrových korekcí je :
 programovaná rychlost v bloku je
 aktuální poloměrová korekce je
 skutečná tečnová rychlost je

$$R = \text{SQR}(I^2 + J^2)$$

$$F$$

$$\text{Kor}$$

$$F_{\text{sku}}$$

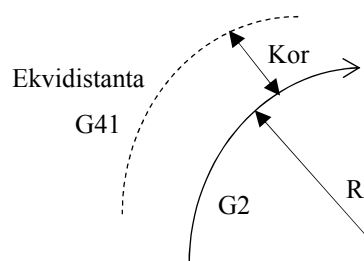
Platí pro jednotlivé případy:

Kruhová interpolace G2 a poloměrová korekce vlevo G41:

Skutečná tečnová rychlost bude **větší**, aby byla zachována programovaná rychlost v místě dotyku nástroje a materiálu. Když je nasazena korekce vlevo, musí se obrábět materiál vždy po pravé straně nástroje, jinak by se účinek ovlivnění rychlosti projevil obráceně a nebyly by zachovány řezné podmínky.

Pro skutečnou rychlost bude platit:

$$F_{\text{sku}} = F * [(R + \text{Kor}) / R]$$

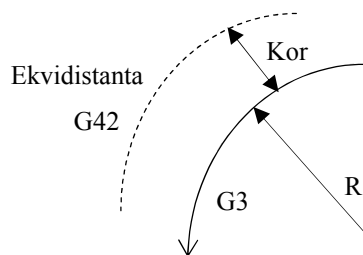


Kruhová interpolace G3 a poloměrová korekce vpravo G42:

Skutečná tečnová rychlost bude **větší**, aby byla zachována programovaná rychlost v místě dotyku nástroje a materiálu. Když je nasazena korekce vpravo, musí se obrábět materiál vždy po levé straně nástroje, jinak by se účinek ovlivnění rychlosti projevil obráceně a nebyly by zachovány řezné podmínky.

Pro skutečnou rychlost bude platit:

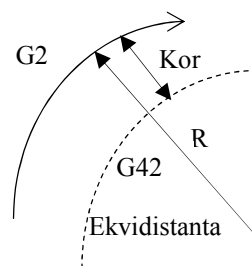
$$F_{sku} = F * [(R + Kor) / R]$$

**Kruhová interpolace G2 a poloměrová korekce vpravo G42:**

Skutečná tečnová rychlost bude **menší**, aby byla zachována programovaná rychlost v místě dotyku nástroje a materiálu. Když je nasazena korekce vpravo, musí se obrábět materiál vždy po levé straně nástroje, jinak by se účinek ovlivnění rychlosti projevil obráceně a nebyly by zachovány řezné podmínky.

Pro skutečnou rychlost bude platit:

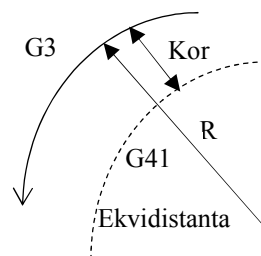
$$F_{sku} = F * [(R - Kor) / R]$$

**Kruhová interpolace G3 a poloměrová korekce vlevo G41:**

Skutečná tečnová rychlost bude **menší**, aby byla zachována programovaná rychlost v místě dotyku nástroje a materiálu. Když je nasazena korekce vlevo, musí se obrábět materiál vždy po pravé straně nástroje, jinak by se účinek ovlivnění rychlosti projevil obráceně a nebyly by zachovány řezné podmínky.

Pro skutečnou rychlost bude platit:

$$F_{sku} = F * [(R - Kor) / R]$$



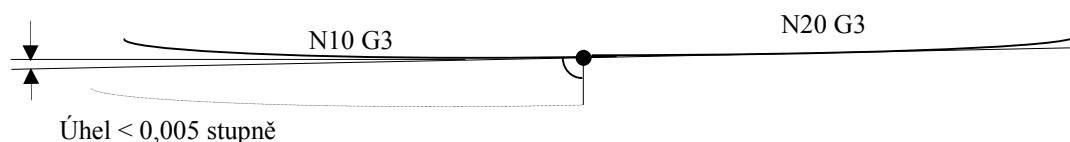
Důležitá poznámka:

Když technologické důvody nedovolí použití korekcí podle výše popsaného způsobu, je možno ovlivňování rychlosti při poloměrových korekcích vypnout pomocí páté dekády strojní konstanty R168. Když se pátá dekáda strojní konstanty R168 nastaví na hodnotu **1**, systém nedodrží konstantní úhlovou rychlost na kružnici, to znamená, že skutečná tečnová rychlost se nemění a pokud nedojde k omezení rychlosti vzhledem k požadované přesnosti (R232), je tato rychlost rovna programované rychlosti.

7.6 Test spojitosti pro poloměrové korekce

Systém při výpočtu poloměrových korekcí vypočítává průsečík ekvidistant. Při výpočtu průsečíku musí vnitřně provádět přípravu příštích bloků (maximálně 200 bloků napřed), aby zjistil ekvidistantu k pohybovým blokům v korekční rovině. Existují případy, když výpočet průsečíku je problematický. Takový problematický výpočet může nastat například tehdy, když jsou programovány 2 kruhové oblouky v následujících blocích, které mají obrovské poloměry, jejich středy jsou velmi vzdáleny od koncových měř bloků, ale leží blízko sebe. Směrnice bloků k dráze pohybu v bodě jejich napojení jsou skoro stejné. Zadáání takových bloků je velmi náročné z hlediska přesnosti, protože už malá nepřesnost (řádově mikrony) způsobí velkou odchylku při výpočtu průsečíku ekvidistant. Tato skutečnost klade velké nároky na CAD systémy pro generaci NC programů. Když není možnost dosažení požadované přesnosti, je možno pomocí strojní konstanty **R234** zadat tzv. „limit pro test spojitosti pro poloměrové korekce“.

V konstantě **R234** se zadává limit pro test spojitosti poloměrových korekcí. Když je úhel mezi směrnicemi tečen v bodu napojení sousedních bloků menší než nastavený limit v konstantě R234, systém nepočítá průsečík ekvidistant, ale vypočte bod podle kolmice k bodu napojení bloků. V tomto případě může být výsledná dráha přesnější než průsečík ekvidistant. Hodnota v konstantě R234 se zadává v tisícinách stupně. Pokud je hodnota nulová, systém počítá v poloměrových korekcích průsečík ekvidistant vždy, kromě případu, že sousední bloky navazují na sebe absolutně tečně. Konstantu se doporučuje nastavit na hodnotu cca 0.005 tisícín stupně. Pokud by hodnota v konstantě R234 byla neúměrně velká, mělo by to negativní vliv na přesnost ekvidistanty.



7.7 Zpřesňování kruhové interpolace

Od verze systému 30.16 a verze kazety 5.023 (3.1.2000) je zavedena možnost automatického zpřesňování zadání kruhové interpolace.

Problematika přesnosti zadání bloku pro poloměrové korekce s ekvidistantou

Systém CNC836 při naprogramování pohybu na kružnici používá standardně preurčený způsob zadávání. To znamená, že kružnice je zadána víc parametry, než je minimálně nutné (kromě souřadnic začátku a konce je nutno naprogramovat obě souřadnice středu I, J). Preurčený způsob zadávání umožní systému například kontrolovat správnost zadání kruhové interpolace a v případě chyby hlásit, že koncový bod neleží na kružnici.

Výpočet ekvidistanty pro poloměrové korekce využívá jednotlivé parametry ze zadání bloku a to vyžaduje v některých případech **vysoké nároky na přesnost zadání** všech parametrů bloku. Například při návaznosti dvou úseků kružnic (o velkých poloměrech) systém vnitřně počítá soustavu kvadratických rovnic a správný kořen ze všech řešení určí podle vzdálenosti ke koncovému bodu ze zadání bloku. Kdyby tato míra nebyla zadána s dostatečnou přesností, systém by mohl určit nesprávný kořen, což by se například mohlo projevit tím, že by se pro ekvidistantu vybral doplňkový kruhový úsek.

V praxi se někdy vyskytují problémy, že návrhový CAD systém nepracuje s požadovanou přesností při generaci NC programu. V tomto případě se musí odpovídajícím způsobem nastavit strojní konstanty **R234** pro test spojitosti a **R55** pro tolerance středu kružnice.

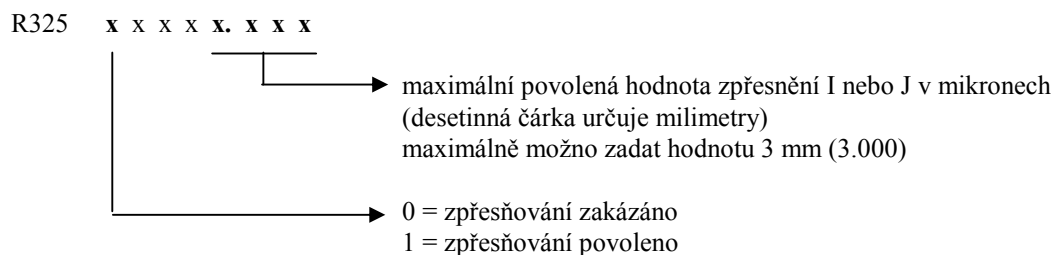
Do CNC systému je zaveden tzv. **preprocesor** pro zpracování partprogramu (který kromě jiného umožní přímé používání funkcí dialogové grafiky v NC programu). Jedna z funkcí zaváděného preprocesoru je také **automatické přepočítávání a zpřesňování zadání dráhy u kruhové interpolace**. Tato nová vlastnost má odstranit problémy s přesností zadání bloků u méně přesných návrhových CAD systémů.

Aktivace a princip zpřesňování

Nutné podmínky pro aktivaci zpřesňování jsou:

- 1.) V souboru CNC836.KNF musí být povolen preprocesor **\$63 = 1** (implicitně přednastaveno)
- 2.) Musí být nastavena strojní konstanta **R325**:

Nastavení strojní konstanty **R325**:



Preprocesor zpřesní hodnotu **I** nebo **J** až na tisíce mikrometru. V zpřesňovaném bloku se automaticky objeví nová adresa **E**, která zpřesňování provede podle vztahu:

$$I_z = I + E/1000 \quad \text{nebo} \quad J_z = J + E/1000 \quad \dots I_z \text{ a } J_z \text{ jsou zpřesněné hodnoty } I \text{ a } J$$

Osmá dekáda adresy **E** je příznak, zda má být zpřesněna adresa **I** nebo **J**. Když osmá dekáda je nastavena na hodnotu **0**, zpřesňuje se položka **I** a když je nastavena na hodnotu **1**, zpřesňuje se položka **J**. Hodnota zpřesnění v položce **E** je v **tisících mikrometru**.

Příklady:

E2.054	$I_z = I + 2.054/1000$	(například pro $I = 12.345$ bude $I_z = 12.347054$)
E-10006.992	$J_z = J - 6.992/1000$	(například pro $J = 5.088$ bude $J_z = 5.081008$)

Hodnoty pro zpřesnění vypočtené preprocesorem je možno pro informaci sledovat ve formátu listingu bloku.

7.8 Volba korekčních a interpolačních rovin

Platí od verze 30.29

Blok s kruhovou interpolací má rovinu interpolace určenou programovanými souřadnicemi. U některých interpolačních rovin v závislosti na typu stroje je nutné otočit pořadí os v kruhové interpolaci, což má vliv na programování resp. přiřazení **I** a **J** k interpolačním osám. Každý stroj podle svého typu má jednoznačně určenou množinu interpolačních rovin.

U poloměrových korekcí se pomocí funkcí G17, G18, G19 a G14, G15 a G16 volí aktuální korekční rovina ze všech možných interpolačních rovin stroje, kterých může být více než korekčních rovin. Technolog má možnost při poloměrových korekcích použít maximálně 6. korekčních rovin.

Pro určení interpolačních a korekčních rovin je určeny strojní konstanty číslo 340 a 341.

Konstanty jsou rozdělené na 8 dvojdekádových částí. V každé části (dvoumístné číslo) se určují pořadová čísla os interpolační roviny.

První číslice z dvojice (vyšší dekáda) určuje pořadové číslo první souřadnice v interpolační rovině (rozsah 1 – 6)

Druhá číslice z dvojice (nižší dekáda) určuje pořadové číslo druhé souřadnice v interpolační rovině (rozsah 1 – 6).

Současně platí:

První dvojice z konstanty 340 určuje současně korekční rovinu, volanou funkcí G17 (implicitně 12).

Druhá dvojice z konstanty 340 určuje současně korekční rovinu, volanou funkcí G18 (implicitně 31).

Třetí dvojice z konstanty 340 určuje současně korekční rovinu, volanou funkcí G19 (implicitně 23).

Čtvrtá dvojice z konstanty 340 určuje pouze interpolační rovinu – nedá se použít pro poloměrové korekce.

První dvojice z konstanty 341 určuje současně korekční rovinu, volanou funkcí G14 (implicitně 34).
Druhá dvojice z konstanty 341 určuje současně korekční rovinu, volanou funkcí G15 (implicitně 24).
Třetí dvojice z konstanty 341 určuje současně korekční rovinu, volanou funkcí G16 (implicitně 41).
Čtvrtá dvojice z konstanty 341 určuje pouze interpolační rovinu – nedá se použít pro poloměrové korekce.

Implicitní stav konstant 340 a 341 je dvojí.

Pokud jsou tam samé nuly, platí interpolační a korekční roviny dle výše uvedeného přehledu, tj. jakoby byly zapsány takto:

R340: 00233.112

R341: 00412.434

Pozn.: Desetinná tečka nemá žádný význam.

7.9 Kontrola a optimalizace ekvidistanty

Platí od verze 40.21

Kontrola správnosti ekvidistanty se provádí při volbě programu vždy pro 3 za sebou následující pohybové bloky se zařazenou poloměrovou korekcí (není rychloposuv, není řazení a vyřazování korekcí atd.). Chyba ekvidistanty se odvodí podle nesouhlasu úhlů tečen programovaných drah s tečnami vypočtených ekvidistant na obou stranách kolem prostředního bloku. Při chybě systém zahlásí upozornění: „Chyba v zadání ekvidistanty pro poloměrové korekce v bloku Nxxxxxx“. Pokud není požadována optimalizace ekvidistanty, chod programu se nijak neovlivní a možno jej odstartovat.

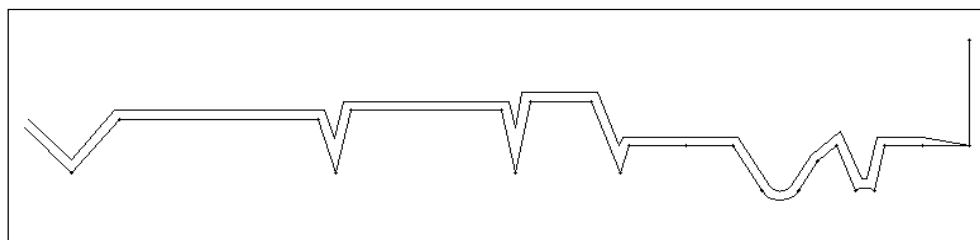
Pokud je požadována optimalizace ekvidistanty, tak se automaticky provede, přičemž jsou spočteny nové korekční vektory pro některé bloky.

Kontrola a optimalizace ekvidistanty se řídí strojní konstantou **R397** a neprovádí v případě vkládání oblouků pro poloměrové korekce (je nastavena konstanta R339).

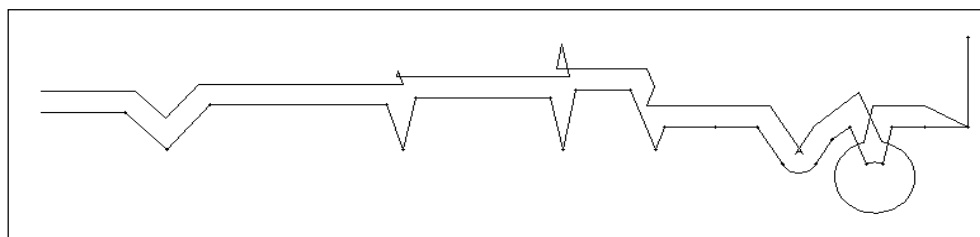
R397	hodnota	Význam
1.dekáda	0	Zařazena kontrola (a možnost optimalizace) pro správnost ekvidistanty.
	1	Vyřazena kontrola a optimalizace pro správnost ekvidistanty.
2. dekáda	0	Když je zařazena kontrola pro správnost ekvidistanty (1.dekáda R397=1), systém hlásí při chybě upozornění s číslem bloku.
	1	Blokování hlášení upozornění při vzniku chyby ekvidistanty.
3. dekáda	0	Optimalizace ekvidistanty je zakázána.
	1	Optimalizace ekvidistanty je povolena při posouzení 3 za sebou následujících pohybových bloků. (Musí být zařazena kontrola pro správnost ekvidistanty, 1.dekáda R397=1.)
4. dekáda	0	(Čtyřbloková optimalizace je zakázána.)
	1	Optimalizace ekvidistanty je povolena při posouzení 4 za sebou následujících pohybových bloků. (Musí být zařazena kontrola pro správnost ekvidistanty, 1.dekáda R397=1 a také „tříbloková optimalizace“ 3.dekáda R397=1.)

Doporučená nastavení:

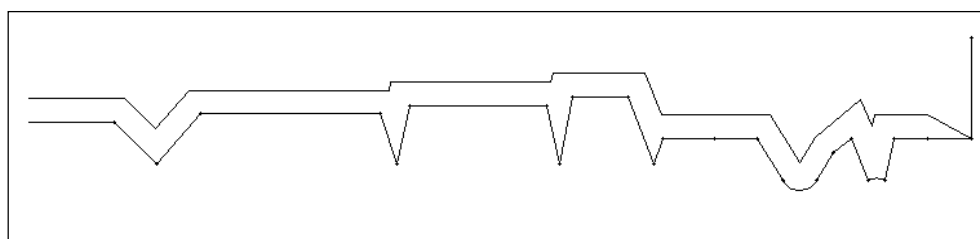
R397 = 00000.000	Probíhá jen kontrola na správnost ekvidistanty s případným upozorněním při chybě.
R397 = 00000.001	Je zakázána kontrola a také optimalizace ekvidistanty.
R397 = 00001.110	Probíhá kontrola a „čtyřbloková“ optimalizace ekvidistanty bez hlášení upozornění.



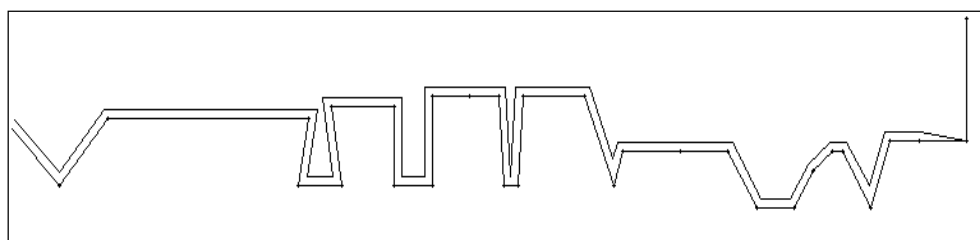
Program s malou poloměrovou korekcí.



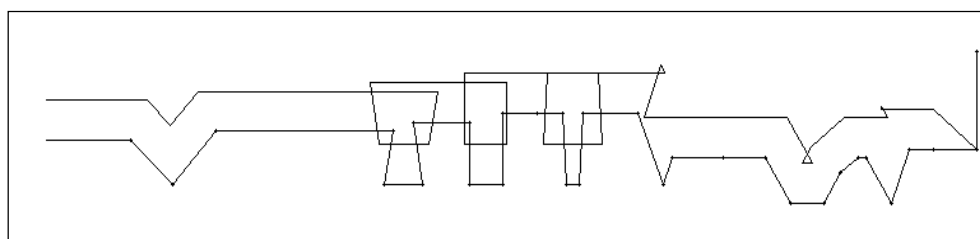
Program s velkou poloměrovou korekcí bez optimalizace.



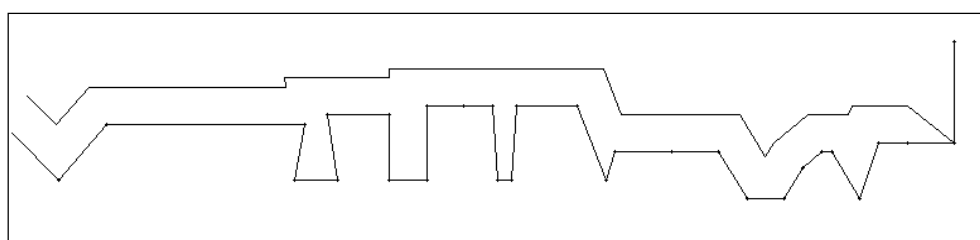
Program s velkou poloměrovou korekcí se zařazenou tříblokovou optimalizací.



Program s malou poloměrovou korekcí.



Program s velkou poloměrovou korekcí bez optimalizace



Program s velkou poloměrovou korekcí se zařazenou čtyřblokovou optimalizací.

7.10 Délkové korekce

7.10.1 Korekce na délku nástroje

Od verze panelu 30.29 se může používat korekce na délku nástroje pro všech šest souřadnic (do této verze to byly pouze 4. souřadnice).

Od uvedené verze se řídí používání délkových korekcí strojní konstantou 329 a to 7. dekadou. Její význam je následující:

A	329	[7dek.] = 0	Starší způsob, k řazení a odvolání délkové korekce používá řídicí funkce &, korekce pouze pro první 4 osy
B	329	[7dek.] = 1	Nový způsob, nepoužívá řídicí funkce &, délkové korekce možno volit pro všechny osy, řazení se provede programováním adresy Dxx, kde xx je číslo tabulky korekcí, odvolání se provede programováním D0
C	329	[7dek.] = 0	Nový způsob, používá řídicí funkce & pouze pro zařazení a odvolání korekce, pouze hodnoty 0=odvolání korekce, 1=zařazení korekce dle Dxx , kde xx je číslo tabulky korekcí

7.10.2 Délkové korekce – způsob ovládání A

Příslušná korekce na délku nástroje se volí řídicí funkcí & a adresou D. Hodnoty jednotlivých délkových korekcí jsou uloženy v tabulce korekcí a volí se adresou D. Přiřazení navolených hodnot délkové korekce z příslušné položky tabulky korekcí, volené adresou D do jednotlivých souřadnic, je řízeno tzv. "řídicí funkcí pro přiřazování délkových korekcí", programovanou pod adresou &. Funkce je čtyřdekádová (k1, k2, k3, k4), jednotlivé dekády k1,k2, k3,k4 mohou nabývat pouze číselné hodnoty 0,1,2.

Při programování &k1k2k3k4 mají jednotlivé dekády následující význam:

k1=0 - do souřadnice X se zavede nulová hodnota délkové korekce
k1=1 - k souřadnici X se přičte hodnota délkové korekce t.j. II. údaj příslušné tabulky korekcí volené adresou D.
k1=2 - od souřadnice X se odečte hodnota délkové korekce (II. údaj příslušné položky tabulky korekcí).
k2=0 - do souřadnice Y se zavede nulová hodnota délkové korekce
k2=1 - k souřadnici Y se přičte hodnota délkové korekce (III. údaj příslušné položky tabulky korekcí).
k2=2 - od souřadnice Y se odečte hodnota délkové korekce. (III. údaj příslušné položky tabulky korekcí).
k3=0 - do souřadnice Z se zavede nulová hodnota délkové korekce
k3=1 - k souřadnici Z se přičte hodnota délkové korekce. (IV. údaj příslušné položky tabulky korekcí)
k3=2 - od souřadnice Z se odečte hodnota délkové korekce. (IV. údaj příslušné položky tabulky korekcí)
k4=0 - do souřadnice 4. se zavede nulová hodnota délkové korekce
k4=1 - k souřadnici 4. se přičte hodnota délkové korekce. (V. údaj příslušné položky tabulky korekcí)
k4=2 - od souřadnice 4. se odečte hodnota délkové korekce. (V. údaj příslušné položky tabulky korekcí)

Řídicí funkce & obsluhuje pouze vybírání hodnoty délkových korekcí z příslušné položky tabulky korekcí. Číslo položky je dáno adresou D. Řídicí funkce & a funkce D jsou trvalé, t.j. platí do doby, dokud nejsou změněny.

Příklad:

N10 &1201 D12 ... K souřadnici X se přičte hodnota z druhé položky tabulky korekcí číslo 12, od souřadnice Y se odečte hodnota ze třetí položky téže tabulky, k souřadnici Z se nepřičte žádná hodnota, ke čtvrté souřadnici se přičte hodnota z páté položky tabulky č.12

Délkové korekce se vztahují výhradně na prostor XYZ a 4. souřadnici a jsou určeny výhradně pro prosté posunutí v tomto prostoru a nikoliv pro kompensování poloměru obráběcího nástroje. Mohou nabývat max. hodnoty +/- 999,999 mm

Při zařazení délkové korekce se děje pohyb na dráze, která je pouze lineárně posunuta o velikost délkové korekce v jednotlivých osách a jejíž tvar je shodný s tvarem programovaným nebo tvarem ovlivněným případně zařazenou poloměrovou korekcí. Délková korekce se zařazuje opět pouze v bloku, kde je programována lineární interpolace nebo rychloposuv. Nástroj je v následujícím programu udržován na korigované dráze, dokud se korekce neodvolá naprogramováním funkce &0, nebo dokud není naprogramována nová hodnota délkové korekce.

Na obrázku je uveden příklad pohybu nástroje při délkových korekcích. Dráha partprogramu bez délkové korekce je na obr. vytištěna silně. Tenkou čarou je znázorněn průběh partprogramu při zařazené délkové korekci. Partprogram pro všechny obrázky je stejný.

Pro průběh délkové korekce dle a) a b) naplníme tabulku korekcí např. takto:
N1 G92 D1 R1=125.0 R2=80.0

Pro průběh délkové korekce dle c) naplníme tabulku korekcí např. takto:
N1 G92 D1 R1=80.0 R2=50.0

Další bloky partprogramu mohou být napsány např. takto (předpokládáme, že souřadnice X a Y jsou v nulové poloze):

N10 X100.0 Y120.0 G1 F1000 D1 &1000	" dle obr a) "
N10 X100.0 Y120.0 G1 F1000 D1 &100	" dle obr b) "
N10 X100.0 Y120.0 G1 F1000 D1 &2200	" dle obr c) "
N20 X350.0	
N30 Y250.0	
N40 X100.0	
N50 Y120.0	
N60 X0 Y0 &0	" zrušení délkové korekce"
N70 M30	

Poznamenejme, že směr délkové korekce je v tomto příkladě určen řídicí funkcí &. Stejného výsledku bychom dosáhli, kdybychom při plnění tabulky (blok N1) zadali záporné hodnoty a do hodnot řídicí konstanty & bychom místo 1 zapsali 2 a místo 2 zapsali 1. Blok N10 by pak vypadal následovně:

N10 X100.0 Y120.0 G1 F1000 D1 &2000	" dle obr a) "
N10 X100.0 Y120.0 G1 F1000 D1 &200	" dle obr b) "
N10 X100.0 Y120.0 G1 F1000 D1 &1100	" dle obr c) "

Při zpracování jednotlivých bloků uvedeného programu je činnost systému následující :

Programování poloměrové a délkové korekce v jednom bloku partprogramu.

V jednom bloku partprogramu může být současně programována poloměrová a délková korekce i ve stejné rovině. Musí být dodržena pravidla pro programování uvedená dříve. Systém si při zpracovávání programu eviduje zvlášť poloměrovou korekci a její rozklady do jednotlivých os a zvlášť délkové korekce v jednotlivých osách. Koncové body se tedy určí jako součet poloměrové a délkové korekce.

7.10.3 Délkové korekce – způsob ovládání B

Od verze panelu 30.29 se může používat korekce na délku nástroje pro všech šest souřadnic a nemusí se programovat resp. používat řídicí funkce &.

Délkové korekce se řídí pouze programováním adresy Dxx, kde xx je číslo korekční tabulky. Korekce se zařadí ve všech osách, které nemají v tabulce nulovou hodnotu. Odvolání korekce se programuje D0.

Pozn.

U ovládání pomocí & může být v tabulce nenulová hodnota pro danou osu, ale jestli se korekce zařadí, záleží pouze na řídicí funkci &. Toto je rozdíl oproti ovládání pouze adresou D, kde rozhoduje obsah tabulky pro danou osu..

7.10.4 Délkové korekce – způsob ovládání C

Od verze panelu 30.29 se může používat korekce na délku nástroje pro všech šest.

Tento způsob je kombinací způsobu ovládání dle A a B. K řazení a odvolání se používá řídicí funkce &, ale ta nabývá pouze dvě hodnoty :

- 0 – vyřadí délkovou korekci
- 1 – zařadí délkovou korekci

Číslo korekční tabulky je dáno adresou D.

8

8. PROGRAMOVÁNÍ TECHNOLOGICKÝCH FUNKCÍ

Význam některých funkcí uvedených v této kapitole může být modifikován tvůrcem interfejsu pro daný stroj. Přesný význam a seznam použitých M-funkcí, případně P a H-funkcí musí dodat tvůrce interfejsu stroje.

8.1 Časová prodleva

Časová prodleva se programuje funkcí G04 ze skupiny G10. Tato funkce má omezenou platnost pouze v bloku v kterém byla programována. Velikost trvání časové prodlevy je definována hodnotou funkce Q která je čtyřdekádová a může tedy nabývat hodnot 0 - 9999. Jedna jednotka vyjadřuje časový úsek 10 ms a tedy je možné naprogramovat časovou prodlevu v rozmezí 10ms až 99,99 sec po 10ms.

Příklad:

Q10 10 x 10ms = 100ms
Q100 100 x 10ms = 1000ms = 1sec
Q1000 1000 x 10ms = 10000ms = 10sec

8.2 Otáčky vřetene

Otáčky vřetene se programují adresou S. Velikost otáček je možno volit ve 4 dekadách tedy v rozsahu 0 - 9999 ot/min. Zařazení otáčkového rozsahu převodového stupně je programováno funkcemi M41, M42, M43, M44.

Start otáčení ve směru CW je zadán funkcí M03, start ve směru CCW je zadán funkcí M04. Po odstartování pohybu těmito funkcemi trvá otáčení tak dlouho, dokud není programována funkce M05 (STOP vřetene) nebo M19 (STOP vřetene v orientovaném bodě).

Systém může vysílat na svém výstupu funkci S buď v BCD kódu, nebo častěji vysílá na svém výstupu analogové napětí v rozsahu +/- 10V pro buzení regulátoru pohonu vřetene.

8.2.1 Stop vřetene v orientovaném bodě a řízení v polohové vazbě.

Stop vřetene v orientovaném bodě se programuje funkcí M19. Po provedeném stopu v orientovaném bodě dojde k zastavení vřetene v definovaném bodě a k uzavření polohové vazby. Poté je možno řídit vřeteno v polohové vazbě.

Řízení vřetene v polohové vazbě slouží k natočení vřetene do libovolné polohy s přesností $1/n$ otáčky, kde n je počet vysílaných pulsů impulsního snímače na jednu otáčku vřetene. Řízení v polohové vazbě prakticky znamená přepnutí vřetene na rotační osu, rychlost se zadává ve stupních za minutu. (F100 = znamená rychlost 100 stupňů za minutu).

V jednom bloku je možno programovat natočení vřetene maximálně o 69999,999 stupňů. Hodnota pootočení se programuje většinou pod adresou C, případně A nebo B. Rozpojení polohové vazby a přechod na řízení vřetene v rychlostní vazbě nastane při programování funkcí M03, M04 nebo M05.

Programování funkce M19 je možné pouze v případě, že ji podporuje interface pro příslušný stroj.

8.3 Chlazení nástroje

Pro ovládání chlazení jsou v systému vyčleněny 2 skupiny M funkcí [M5,M6], které mohou ovládat 4 samostatné chladicí okruhy. Čísla M funkcí jsou vedena v souhrnném přehledu adres, používaných systémem CNC8x6.

Upřesnění funkcí pro konkrétní stroj musí dodat tvůrce interfejsu daného stroje (např. používají-li se dva okruhy chlazení apod.)

8.4 Přerušení a konec partprogramu

V systému lze programovat čtyři M-funkce (M00,M01,M02,M30), kterými lze ukončit nebo přerušit zpracováváný partprogram:

- | | |
|-----|--|
| M00 | - Nepodmíněný stop
Zpracováváný partprogram se přeruší po vykonání všech operací v bloku, v němž je programována tato funkce. Současně se ve skupinách M2, M5 a M6 nastaví dočasné M-funkce na hodnoty označené + (viz tabulka významu adres) t.j. M05 (stop vřetene), M00 (vypnutí chlazení 1 a 2), M52 (vypnutí chlazení 3 a 4). Po následném odstartování partprogramu tlačítkem START se hodnoty M-funkcí ve skupinách M2, M5 a M6 vrátí na své původní hodnoty, pokud není programováno jinak. |
| M01 | - Podmíněný stop
Systém se zachová shodně jako při M00 v případě, že pracuje v modifikaci režimu AUT - M01. |
| M02 | - Konec partprogramu
Konec partprogramu se zpětným návratem na začátek partprogramu. Vyšlou se závěrečné funkce M05, M09, M52. |
| M30 | - Systém se zachová shodně jako při M02. |

Pozn.:

Tvůrce interfejsu pro konkrétní stroj může chování těchto funkcí do jisté míry modifikovat.

8.5 Upnutí a uvolnění obrobku

Funkcí M10 je vyslán příkaz k upnutí obrobku a funkcí M11 je vyslán příkaz k uvolnění obrobku. Vlastní cyklus upínání a uvolnění řídí programovatelný interface systému. Funkce M10 a M11 patří do sedmé skupiny.

8.6 Výměna nástroje a obrobku

Funkcí M06 je vyslán příkaz k výměně nástroje. Vlastní cyklus výměny řídí programovatelný interface systému.

Funkcí M60 se vysílá příkaz k výměně obrobku. Vlastní cyklus výměny řídí programovatelný interface systému. Funkce M06 a M60 patří do osmé skupiny.

8.7 Pomocné M-funkce skupiny M14

V této skupině lze programovat libovolnou jednu hodnotu M funkce z hodnot, které nebyly uvedeny v ostatních skupinách (M1 až M9). Systém vysílá tuto programovanou hodnotu z této skupiny v BCD kódu.

8.8 Pomocné M-funkce skupiny M10, M11, M12, M13

Uživatelé nadefinované skupiny M-funkcí. Každá nadefinovaná skupina může obsahovat čtyři M-funkce. Kódy M-funkcí se zadávají jako strojní konstanty.

8.9 Číslo nástroje - funkce T

Funkcí T, která je 8-mi dekadová, se zadává číslo nástroje, který je na základě vyslané funkce M06 zařazen do polohy vhodné pro obrábění (stroj obsahuje automatickou výměnu nástrojů).

8.10 Pomocné funkce H, P

Tyto vyslané funkce jsou dvoudekadové; jejich hodnoty jsou vysílány v BCD kódu a jsou určeny pro ovládání manipulátoru obrobku a nástrojů nebo pro jiné uživatelem požadované využití.

9

9. PARAMETRICKÉ PROGRAMOVÁNÍ - FUNKCE R

Systém umožňuje programovat místo konkrétní hodnoty libovolné adresy příslušný parametr R. V okamžiku zpracování dané adresy programované pomocí parametru R se místo tohoto parametru dosadí hodnota, která mu byla naposledy v některém v předcházejícím bloku partprogramu přiřazena (deklarována). Parametrické programování má hlavní význam při využívání pevných cyklů, makrocyklů a podprogramů, kdy obvykle v programových blocích vlastního partprogramu jsou hodnoty jednotlivých parametrů deklarovány a z pevného cyklu, podprogramu nebo makrocyklu vyvolány. Nevylučuje se však možnost deklarace a vyvolání parametrů i v ostatních blocích partprogramu.

Další hlavní využití parametrů je při aritmetických operacích. Tato problematika je popsána v samostatné kapitole.

Práci s parametry rozdělujeme do dvou fází:

- a) deklarace parametrů
- b) vyvolání parametrů

9.1 Deklarace parametrů

Při deklaraci parametrů přiřazujeme jednotlivým parametrům konkrétní hodnoty. Deklarace parametrů v bloku partprogramu má následující tvar :

N10 R0=aaaa R1=bbbb R2=cccc R95=xxxx

kde aaaa,bbbb,cccc,....,xxxx jsou deklarované hodnoty.

Plnění parametrů R hodnotami je možno psát v bloku společně s ostatními adresami.

Příklad:

N620 G54 G0 X0 Y0 R1=864 R2=-0.864 R3=100000 R20=250 R31=1

Po provedení tohoto bloku budou v tabulce parametrů přepsány parametry č.1, 2, 3, 20 a 31 novými hodnotami. Hodnoty parametrů jsou bezrozměrné. Poznamenejme, že první i druhý parametr bude v tabulce uložen jako 0.864 (druhý se znaménkem mínus), hodnota 100000 jako 100.000, hodnota 250 jako 0.250 a hodnota 1 jako 0.001

Maximální počet parametrů je 96 (0 až 95). Maximální hodnota parametru může být v rozsahu +69999.999 až -69999.999. Deklarovaná hodnota však musí být v souladu s přípustnou hodnotou adresy, u které bude parametr vyvoláván. Parametr 95 se nesmí používat (je vyhrazen jako řídicí), parametry 90 až 94 se nedoporučuje používat, protože jsou rezervované pro eventuální rozšíření řídicích parametrů.

UPOZORNĚNÍ: při načítání partprogramu není možná kontrola na přípustnost hodnot jednotlivých parametrů a za správnou deklaraci hodnot parametrů odpovídá plně programátor. Správnost parametrů se vyhodnotí až při vykonávání partprogramu.

Příklad.:

N10 R10=123

N20 GR10

Tento zápis by při vykonání partprogramu vyhlásil chybu „Do dvoudekádové adresy nelze dosadit R o více než 2 dekádách“, protože hodnota u G může být max. dvoudekádová, ale hodnota použitého parametru má tři dekády (viz dále).

9.2 Vyvolávání parametrů

Při vyvolávání parametrů se za tento parametr dosadí jeho dříve deklarovaná hodnota. V partprogramu se místo hodnoty u příslušné adresy uvede adresa Rxx, kde xx je číslo parametru.

Příklad:

Blok, ve kterém jsou vyvolány parametry, deklarované v předešlém příkladě (blok N620):

N630 GR31 XR1 YR2 M03 SR20 TR3

Pokud by se nepoužily parametry, byl by blok zapsán takto:

N630 G01 X0.864 Y-0.864 M03 S250 T100000

Hodnoty všech adres, kromě čísla bloku N, lomítka (/), adresy R a funkce G79 lze parametrizovat.

Příklad:

Partprogram s parametrickým programováním:

```
.  
. .  
. .  
N2 R30=350. R31=250. R29=1000  
N3 R1=100. R2 =200.  
N4 R3=15 M03 M42 M08  
N5 G90 G00 XR1 TR3  
N6 G82  
N7 G91 XR1 XR2  
N8 G90 G82  
N9 Z500. M05  
. .
```

Na uvedeném příkladu je provedena jednak deklarace parametrů R v samostatných programových blocích (N02, N03), tak i v bloku společně s programováním ostatních adres (N04). V bloku N5 a N7 je provedeno vyvolání parametrů.

9.3 Desetinná tečka u deklarace parametrů

Při deklaraci parametrů je důležité si ujasnit vliv desetinné tečky. Parametry jsou obecně bezrozměrná čísla, rozměr získají až přiřazením k adrese. Parametr může být zapsán s desetinnou tečkou nebo bez ní a v obou případech bude mít stejný význam, pokud se zapíše správně. Je-li parametr zapsán bez desetinné tečky, má u rozměrových adres (např. X,Y,Z, I,J) rozměr v mikrometrech !

Rozdíl se nejlépe osvětlí na příkladu:

```
N10 G0 X100
N20 R10=100
N30 R11=100.
N40 R12=0.100
N50 XR10 YR11 ZR12
```

V bloku N10 je programována adresa X přímo hodnotou 100 (bez des. tečky). Pokud jde o přímé programování u rozměrových adres, nemusí (ale může) se desetinná tečka psát a rozměr bude v milimetrech, tj. 100mm. U parametrického programování bude výsledek následující:

V bloku N50 odjede X na míru 100 mikronů (0.1mm), Y na míru 100mm a Z na 100mikronů (0.1mm).

Při psaní partprogramů se tedy doporučuje psát deklaraci parametrů určených pro adresy, které mají charakter délky nebo vzdálenosti s desetinnou tečkou, což je přehlednější než zápis bez desetinné tečky v mikronech. Naopak bezrozměrové údaje psát (pro přehlednost) bez desetinné tečky. Týká se to např. deklarace parametru pro otáčky S, G-funkce, M funkce, T-funkce apod.

Příklad:

```
N10 R20=120 R21=40 R22=12345678
N20 SR20 GR21 TR22
```

V bloku N20 se provede vyslání 120otáček, zrušení korekcí G40 a vyslání T12345678.

Stejný význam by měl i zápis s desetinnou tečkou zapsaný takto, ale tento zápis je pro uvedené funkce méně přehledný:

```
N10 R20=0.120 R21=0.040 R22=12345.678
N20 SR20 GR21 TR22
```

Při programování rychlosti posuvu (viz kapitola 6) lze zadat F bez tečky, tj. v mm/min nebo s tečkou, tj. v m/min. Rychlost F zadaná parametricky bude mít tento rozměr:

```
N10 R10=100
N20 R10=0.100
N30 R10=100.
N40 FR10
```

Správně deklarované hodnoty jsou v blocích N10 a N20. Lze říci, že rychlost v bloku N10 je zadaná v mm/min, rychlost v bloku N20 je zadaná v m/min (obě rychlosti jsou stejné).

Rychlost zadaná v bloku N30 je v m/min a znamenala by rychlost 100m/min – tedy rychlost větší než systém umožňuje.

10

10. ŠROUBOVICE

Řídící systémy CNC8x6 umožňují vlečení souřadnic, t.j. řízení jedné souřadnice v závislosti na pohybu jiné souřadnice. Tato vlastnost je využita u šroubovice, což je případ vlečení souřadnice v závislosti na pohybu po kružnici.

Programováním šroubovice umožní vázaný pohyb kruhové a lineární interpolace.

Zadání šroubovice se programuje zadáním stoupání u souřadnice, která je vlečená (osa, která se pohybuje lineární interpolací) a funkcí G10. Stoupáním se rozumí vzdálenost, o kterou se má posunout vlečená souřadnice, po dobu celé kružnice v kruhové interpolaci řídicích souřadnic. Takto zadané stoupání není závislé na poloměru kruhové interpolace (a ani na poloměrových korekcích). Úhel odklonu se ale mění v závislosti na poloměru kruhové interpolace.

Blok, kde je programováno stoupání není pohybový a programuje se obvykle samostatně a to v kterémkoli bloku před vlastní šroubovicí. Zadané stoupání platí trvale až do další změny. Například v bloku N10 Z1 G10 je programováno stoupání 1.05 mm a vlečenou osou bude souřadnice Z.

Programování šroubovice se provádí funkcí G09, zrušení šroubovice se programuje funkcí G98. Funkce G98 musí být naprogramována v samostatném bloku. Při programování šroubovice se musí zadat počet "závitů". Počet závitů se programuje tzv. skokem s opakováním na tentýž blok.

Příklad:

```
N5 X0 Y0 Z0 G00
N10 Z1.05 G10          " zadání stoupání 1.05 mm
....
....
....
N50 G09 G02 X0 Y0 I100 G73 L50 Q20 " začátek šroubovice
N60 X200. Y0 I100
N70 G98                " konec šroubovice
N80 X... Y... Z...     " pohyb všech souřadnic
```

V bloku N50 je programována kružnice o poloměru 100mm (výchozí i koncový bod je na souřadnicích 0,0). Kružnice se zopakuje celkem 21x. (o 1 více než je programováno pod adresou Q). Funkcí G73 je programován skok na blok, programovaný pod funkcí L. V bloku N60 je programována kruhová výseč (půlkruh) a v bloku N70 je zrušení funkce šroubovice (G09).

11

11. ARITMETIKA PARAMETRŮ

(Platí od softwarové verze panelu 30.19 - 14.2.2000.)

11.1 Společné zásady

Aritmetiku parametrů vykonávají funkce **G26 G27 G28** a **G29**. V bloku, kde je jedna z těchto funkcí programována, musí být programovány příslušné řídicí parametry R5 až R8.

Funkce **G26** vykoná operaci podle řídicího parametru **R5**.

Funkce **G27** vykoná 2 operace postupně podle řídicích parametrů **R5** a **R6**. (Nejdříve R5 a potom R6).

Funkce **G28** vykoná 3 operace postupně podle řídicích parametrů **R5 R6** a **R7**. (Nejdříve R5 potom R6 a nakonec R7).

Funkce **G29** vykoná 4 operace postupně podle řídicích parametrů **R5 R6 R7** a **R8**. (Nejdříve R5, potom R6, potom R7 a nakonec R8).

Parametr **R95** je vyhrazen pro řídicí funkce.

11.2 Řízení dvojité přesnosti

Některé operace mohou načíst do svých operandů hodnoty z dvojitou přesností - čtyři slova. V tomto případě se do operandu načtou hodnoty z dvou po sobě následujících parametrů **op1** a **op1+1**, nebo **op2** a **op2+1**. Pokud je dvojitá přesnost parametrů povolena, v řídicím parametru se udává číslo parametru zvětšené o 50.

Od softwarové verze panelu 30.08 (10.9.1999) má systém k dispozici **96 parametrů**. Při parametrických operacích si proto musíme zvolit, zda budeme používat dvojitou přesnost a využijeme jen 50 parametrů nebo nebudeme používat dvojitou přesnost a využijeme všech 96 parametrů.

1. Dvojitá přesnost zakázána:

Číslo parametru: 0, 1, 2,.....,95 Základní přesnost: 32 bitů

2. Dvojitá přesnost povolena:

Číslo parametru: 0, 1, 2,.....,49 Základní přesnost: 32 bitů

50, ,52,.....,99 Dvojitá přesnost pro parametry 0,1,...,49: 64 bitů

Ovládání dvojité přesnosti parametrů je umožněno pomocí strojních konstant nebo možno použít dynamické řízení pomocí parametru **R95**.

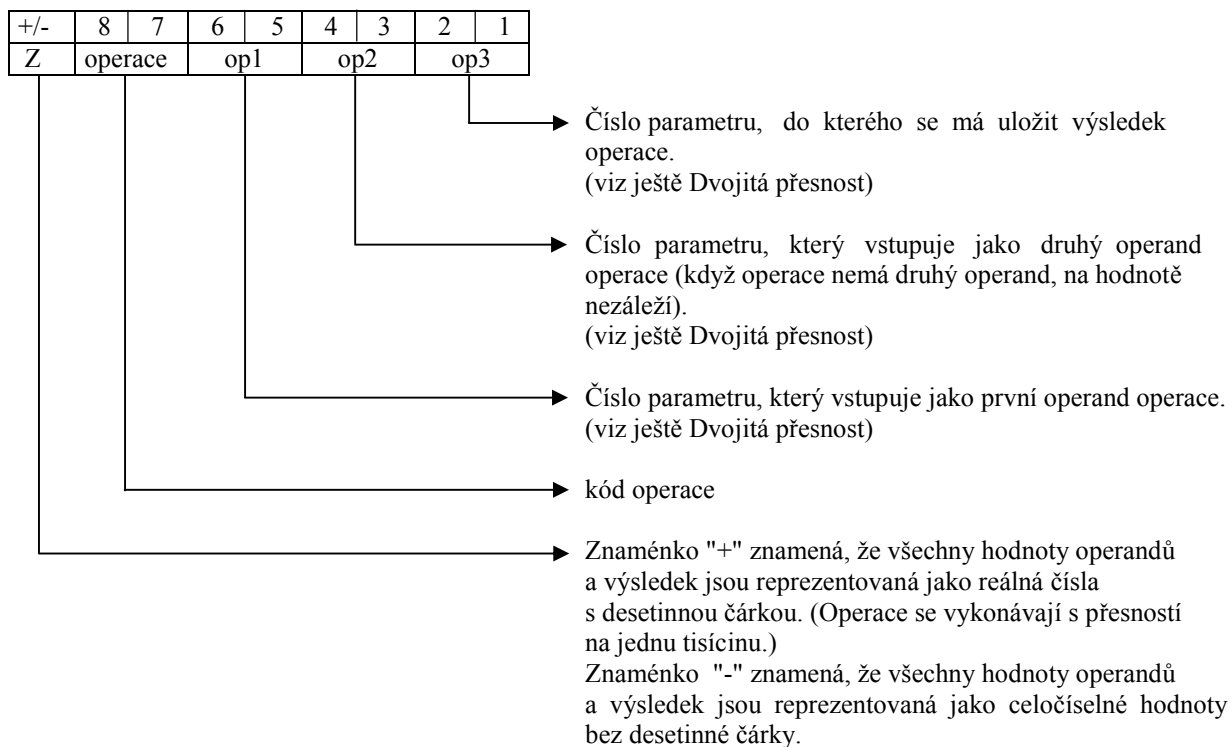
4.dekáda strojní konstanty 283:

- 0** aritmetika parametrů používá parametry 0-49 s možností dvojité přesnosti operandů
- 1** aritmetika parametrů používá parametry 0-95 bez možnosti dvojité přesnosti operandů
- 2** řízení dvojité přesnosti operandů určuje 1.dekáda parametru R95 :
 - 1.R95=0 rozsah 0-49, dvojitá přesnost povolena
 - 2.R95=1 rozsah 0-95, dvojitá přesnost zakázána

Když je 4. dekáda stroj.konstanty 283 nastavena na hodnotu 2, je možno řídit aktivaci dvojité přesnosti ve výpočtech dynamicky. Na nastavování 1.dekády parametru R95 doporučujeme použít instrukce:

R5=48110095 G26 ;funkce SET 1.R95 (nastaví 1.dekádu parametru R95 na hodnotu 1)
 R5=48210095 G26 ;funkce CLR 1.R95 (nastaví 1.dekádu parametru R95 na hodnotu 0)

11.3 Struktura řídicích parametrů



V dalším popisu operací budou hvězdičkou označeny operandy a výsledky, které mohou mít dvojitou přesnost.

OPERACE	KÓD	POPIS
Kopírování $op1^* \rightarrow op2^*$	0	Překopíruje hodnotu parametru podle op1* do parametru podle cíl* . Hodnota parametru op1 zůstane nezměněna. Hodnota op2 a Z nemají vliv.
Sčítání $op1^* + op2^* \rightarrow cíl^*$	1	Sčítání hodnoty parametru podle op1* s hodnotou parametru podle op2* a uložení výsledku do parametru podle cíl* .
Odčítání $op1^* - op2^* \rightarrow cíl^*$	2	Odčítání hodnoty parametru podle op2* od hodnoty parametru podle op1* a uložení výsledku do parametru podle cíl* .
Násobení $op1^* \cdot op2^* \rightarrow cíl^*$	3	Násobení hodnoty parametru podle op1* s hodnotou parametru podle op2* a uložení výsledku do parametru podle cíl* .
Dělení $op1^* / op2^* \rightarrow cíl^*$	4	Vydělení hodnoty parametru podle op1* s hodnotou parametru podle op2* a uložení výsledku do parametru podle cíl* .
Druhá mocnina $op1^2 \rightarrow cíl^*$	5	Druhá mocnina hodnoty parametru podle op1 a uložení výsledku do parametru podle cíl* .
Druhá odmocnina $\sqrt{(op1^*)} \rightarrow cíl^*$	6	Druhá odmocnina hodnoty parametru podle op1* a uložení výsledku do parametru podle cíl* .
Absolutní hodnota $ op1^* \rightarrow cíl^*$	7	Absolutní hodnota parametru podle op1* a uložení výsledku do parametru podle cíl* .
Inverzní hodnota $-op1^* \rightarrow cíl^*$	8	Inverzní hodnota parametru podle op1* a uložení výsledku do parametru podle cíl* .
Binární posun $op1^*.2op2 \rightarrow cíl^*$	9	Binární posun hodnoty parametru podle op1 . Posun se provede podle parametru op2 . Hodnota op2 může být ± 32 (± 64 pro dvojitou přesnost)
Převrácená hodnota $1/op1^* \rightarrow cíl^*$	10	Převrácená hodnota parametru podle op1* a uložení výsledku do parametru podle cíl* .
Funkce 2^X $2^{op1} \rightarrow cíl^*$	11	Funkce 2^X podle op1 a uložení výsledku do cíl . Operand op1 nemůže mít dvojitou přesnost a musí být reálné číslo. Výsledek je reálné číslo a může mít dvojitou přesnost.
Funkce $\log_2 x$ $\log_2(op1) \rightarrow cíl^*$	12	Funkce $\log_2 x$ podle op1 a uložení výsledku do cíl . Operand op1 může mít dvojitou přesnost a musí být reálné číslo. Výsledek je reálné číslo a nemůže mít dvojitou přesnost.
Funkce x^y $(op1)^{op2} \rightarrow cíl^*$	13	Umocnění hodnoty podle op1 na hodnotu podle hodnoty v op2 a uložení výsledku do cíl . Operandy op1 i op2 nemohou mít dvojitou přesnost a musí být reálná čísla. Výsledek je reálné číslo a může mít dvojitou přesnost.
Funkce $\sqrt{(x^2 + y^2)}$ $\sqrt{(op1^2 + op2^2)} \rightarrow cíl^*$	14	Funkce $\sqrt{(op1^2 + op2^2)}$ Možno použít pro reální i celočíselné hodnoty.
Větší $op1^* > op2^* \rightarrow cíl^*$	15	Pokud hodnota parametru podle op1 je větší než hodnota parametru podle op2 , uloží se do parametru podle cíl* hodnota 73 , jinak 78 .
Větší nebo rovno $op1^* \geq op2^* \rightarrow cíl^*$	16	Pokud hodnota parametru podle op1 je větší nebo rovná než hodnota podle op2 , uloží se do parametru podle cíl* hodnota 73 , jinak 78 .
Menší $op1^* < op2^* \rightarrow cíl^*$	17	Pokud hodnota parametru podle op1 je menší než hodnota podle op2 , uloží se do parametru podle cíl* hodnota 73 , jinak 78 .
Menší nebo rovno $op1^* \leq op2^* \rightarrow cíl^*$	18	Pokud hodnota parametru podle op1 je menší nebo rovná než hodnota podle op2 , uloží se do parametru podle cíl* hodnota 73 , jinak 78 .
Porovnání $op1^* = op2^* \rightarrow cíl^*$	19	Pokud hodnota parametru podle op1 je rovná jako hodnota podle op2 , uloží se do parametru podle cíl* hodnota 73 , jinak 78 .

Podmínka "rovno" $op1 * = op2 * \rightarrow$ pokračuj	20	Pokud hodnota parametru podle op1 je rovná jako hodnota podle op2 , pokračuje výpočet na následující parametrické operace
Podmínka "nerovno" $op1 * \neq op2 * \rightarrow$ pokračuj	21	Pokud hodnota parametru podle op1 není rovná jako hodnota podle op2 , pokračuje výpočet na následující parametrické operace.
Konec podmínky $\rightarrow$ pokračuj	22	Ukončení podmínek pro výpočet parametrických operací
Převod int $\rightarrow$ real $real(op1) \rightarrow$ cíl	23	Převod celočíselné hodnoty podle parametru op1 na reálnou hodnotu podle cíl
Převod real $\rightarrow$ int $int(op1) \rightarrow$ cíl	24	Převod reálné hodnoty podle op1 na celočíselnou hodnotu podle cíl
Přesun odměřování ODM $\rightarrow$ cíl*	25	Přesun odměřování souřadnice podle op1* (a op2*) a uložení do parametru podle cíl* . Do operandů op1 (op2) se zadává pořadové číslo souřadnice (1,2,...,6).
Přesun programované hodnoty souřadnic PROG $\rightarrow$ cíl*	26	Přesun programované souřadnice podle hodnoty souřadnic op1* (a op2*) a uložení do parametru podle cíl* . Do operandů op1 (op2) se zadává pořadové číslo souřadnice (1,2,...,6).
Funkce $\tan(x)$ $\tan(op1) \rightarrow$ cíl*	27	Funkce $\tan(x)$ podle op1 a uložení výsledku do cíl . Operand op1 nemůže mít dvojitou přesnost a musí být reálné číslo. Výsledek je reálné číslo a může mít dvojitou přesnost.
Součet geometrické řady $1 + k + k^2 + \dots + k^{n-1}$ $(k^n + 1) / (k - 1) \rightarrow$ cíl* $k \dots op1 *$ $n \dots op2 *$	28	Součet geometrické řady $1+k+k^2+\dots+k^{n-1} = (k^n-1)/(k-1)$ Kvocient k je dán parametrem op1 . Počet členů n je dán parametrem op2 . Operand op1 i op2 nemohou mít dvojitou přesnost a musí být reálná čísla. Výsledek je reálné číslo a může mít dvojitou přesnost.
Podmínka "větší" $op1 * > op2 * \rightarrow$ pokračuj	29	Pokud hodnota parametru podle op1 je větší než hodnota podle op2 , pokračuje výpočet na následující parametrické operace
Podmínka "větší nebo rovno" $op1 * \geq op2 * \rightarrow$ pokračuj	30	Pokud hodnota parametru podle op1 je větší nebo rovna než hodnota podle op2 , pokračuje výpočet na následující parametrické operace
Podmínka "menší" $op1 * < op2 * \rightarrow$ pokračuj	31	Pokud hodnota parametru podle op1 je menší než hodnota podle op2 , pokračuje výpočet na následující parametrické operace
Podmínka "menší nebo rovno" $op1 * \leq op2 * \rightarrow$ pokračuj	32	Pokud hodnota parametru podle op1 je menší nebo rovna než hodnota podle op2 , pokračuje výpočet na následující parametrické operace
PROGRAM spuštění programu od op1 po cíl	33	Spuštění programu od parametru op1 po parametr cíl (včetně). Parametry mohou být naplněny ve stejném bloku. Za funkcí 33 nesmí být programována další parametrická operace
Přesun parametrů přesun od op1 po op2 do cíl	34	Přesun parametrů od op1 (včetně) po op2 (včetně) do oblasti podle parametru cíl Pro op1=99 se vynuluje oblast od cíl po op2
Výměna parametrů výměna od op1 po op2 do cíl	35	Výměna parametrů od op1 (včetně) po op2 (včetně) s oblastí podle parametru cíl
Vepsaný oblouk $\sin(op1) \rightarrow$ cíl	36	Funkce využívá větší oblast parametrů
$\sin(op1) \rightarrow$ cíl	37	Funkce $\sin(x)$ podle op1 a uložení výsledku vynásobeného *10000000 do cíl .
$\cos(op1) \rightarrow$ cíl	38	Funkce $\cos(x)$ podle op1 a uložení výsledku vynásobeného *10000000 do cíl .
$\arcsin(op1/op2)$	39	Funkce $\arccos(x/y)$ podle op1/op2 a uložení výsledku do cíl .
$R \cdot \cos((I \cdot B/N) + A)$	40	Funkce pro výpočet děr na kružnici

op1+1 -> cíl	41	pro op2=0 inkrementace hodnoty op1 a uložení do cíl pro op2<>0 zvětší hodnotu podle op1 o číslo v op2 a uloží do cíl
op1-1 -> cíl	42	pro op2=0 dekrementace hodnoty op1 a uložení do cíl pro op2<>0 odečte hodnotu podle op1 o číslo v op2 a uloží do cíl
SAVE	44	Úchova všech parametrů
RESTORE	45	Obnova parametrů od op1 po op2 včetně do cíl Pro op1=op2=cíl=0 obnovení všech 96 parametrů
LICHÝ(op1) -> cíl	46	Pokud hodnota parametru podle op1 je lichá, uloží se do parametru podle cíl hodnota 73 , jinak 78 .
SUDÝ(op1) -> cíl	47	Pokud hodnota parametru podle op1 je sudá, uloží se do parametru podle cíl hodnota 73 , jinak 78 .
LOGIC(dek) -> dek(cíl)	48	Logické operace pro dekády parametrů 8. a 7. dekáda je kód pro logiku = 48 6. dekáda je kód operace: 1 = SET, 2=CLR, 3=NOT, 4 = OR, 5= AND, 6= XOR, 7=TEST 5. dekáda určuje pořadové číslo dekády pro logickou operaci 1,2,...,8 4. a 3. dekáda určují 1.operand pro operace OR, AND, XOR a TEST 2. a 1. dekáda určují 2.operand a současně cíl pro operace OR, AND, XOR. Operace SET, CLR a NOT pracují pouze s parametrem podle cíl . Pro operaci TEST se do parametru podle cíl uloží hodnota 73 nebo 78 .
READ(TAB) -> cíl	49	Přesuny z tabulek do parametrů 8. a 7. dekáda je kód pro přesuny = 49 6. dekáda je kód tabulky: 1=tab.korekcí, 2=tab.posunutí 3=tab.strojních konstant, 4=registr bloku BWRK, 5=registr bloku AWRK 5. dekáda určuje sloupec v jednotlivých tabulkách 1,2... pro registry bloku je: 1=funkce M, 2=funkce G 3=funkce P, 4= H, 5= S, 6= T, 7= Q, 8=D 4. a 3. dekáda určuje zdroj v tabulkách. Pro tab.korekcí je 1-99, pro tab.posunutí je 53-59, pro stroj. konstanty je 0-99 + sloupec*100, pro registry bloku se zadává skupina funkcí M a G 2.a 1. dekáda určuje cílový parametr pro zápis hodnoty z tabulky
WRITE(op) -> TAB	50	Přesuny z parametrů do tabulek 8. a 7. dekáda je kód pro přesuny = 50 6. dekáda je kód tabulky: 1=tab.korekcí, 2=tab.posunutí 5. dekáda určuje sloupec v jednotlivých tabulkách 1,2... 4.a 3. dekáda určuje zdrojový parametr pro zápis hodnoty do tabulky 2. a 1. dekáda určuje cíl v tabulkách. Pro tab.korekcí je 1-99, pro tab.posunutí je 53-59

MESSAGE	51	Zobrazení technolog. informačního hlášení 8. a 7. dekáda je kód pro zprávy = 51 6. a 5. dekáda je kód hlášení 1,2,...,99 4. a 3. dekáda je číslo parametru, který může být zobrazen ve zprávě (v textu je znak samotného \$). Hodnota zobrazeného parametru je načtena v okamžiku průchodu této instrukce a počas zobrazení není aktualizována. (Kromě tohoto parametru může být zobrazován také parametr průběžně aktualizován - v textu je nutno zadat jeho číslo \$xy) 2. dekáda je čas zobrazení (pokud je požadován) ve vteřinách 1,2,...,9 1. dekáda je řídící pro způsob zobrazení: 0= zrušení hlášení 1= trvalé hlášení (pokud nebude zrušeno) 2= hlášení je zobrazeno po dobu trvání jednoho bloku 3= hlášení je zobrazeno po dobu určenou pro čas zobrazení zadanou v 2. dekáde
---------	----	---

11.4 Příklad využití aritmetiky parametrů

Pro výklad použití aritmetiky parametrů použijeme standardně dodávaný makrocycklus L9001.NCP pro vrtání děr v pravidelných roztečích a sloupcích. V systémech CNC8x6 je v současné době možno využívat 51 matematických operací. V příkladu nelze podrobně všechny popsat, nicméně princip používání je stejný a příklad jistě poslouží jako námět pro tvorbu vlastních programů s využitím aritmetiky parametrů.

Makrocycklus pro vrtání děr používá tyto vstupní parametry:

R09 = číslo pevného cyklu 81 - 89

R10 = Rozteč děr v ose X (= rozteč sloupců)

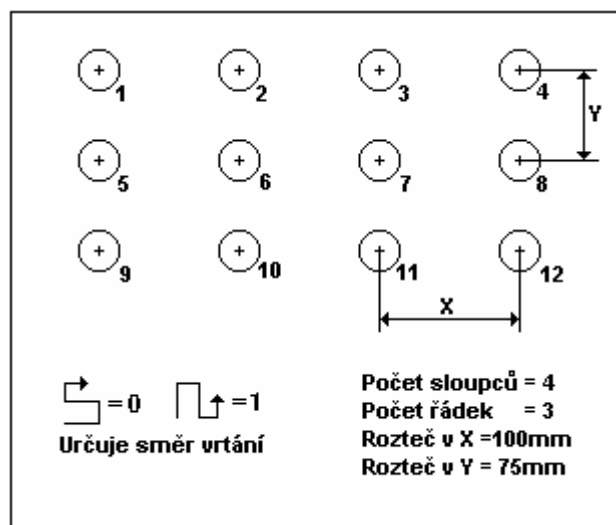
R11 = Rozteč děr v ose Y (= rozteč řádek)

R12 = Počet sloupců

R13 = Počet řádek

R14 = 0 ... vrtání po řádcích (začíná v ose X)

1 ... vrtání po sloupcích (začíná v ose Y)



Makrocycklus předpokládá výchozí bod nad první dírou, tj. programátor musí zajisti v partprogramu nájezd nad první díru.

Poznamenejme ještě, že znaménko u parametru R10 a R11 určuje směr pohybu od první díry, proto se může začít v libovolném rohu pomyslného čtverce resp. obdélníka. Makrocycklus se vrací po ukončení opět do výchozího bodu.

Dále se předpokládá, že minimální počet zadaných řádek je 2 a rovněž minimální počet zadaných sloupců je 2.

Pro náš příklad (viz obr.) budou parametry nastaveny takto:

R09 = 81 „pevný cyklus G81

R10 = 100.0

R11 = 75.0

R12 = 4 „počet sloupců

R13 = 3 „počet řádků
R14 = 0 „vrtání po řádcích (viz čísla u děr)

Předpokládáme souřadnou soustavu X,Y, kladné směry vpravo a nahoru.

Pozn.:

Pokud se podíváme na parametry např. ve volbě indikace, budou parametry, zadané bez tečky zformátovány do tvaru s desetinnou tečkou takto:

R09=0.081, R12=0.004, R13=0.003

Hodnota je ale stejná – nicméně s těmito čísly se musí v některých aritmetických operacích pracovat jako s reálnými čísly (před kódem operace je znaménko mínus).

V dalším textu provedeme podrobný komentář k aritmetickým operacím.

N1 G79 L9001

V bloku N10 se provede deklarace různých pomocných parametrů, které budou potřeba pro další výpočty, deklarují se zde také různé konstanty, potřebné pro další výpočty.

N10 **R15=0** " Pomocný pracovní parametr
 R16=0 " Pomocný pracovní parametr
 R17=0 " Pomocná deklarace konstanty 0
 R18=31 " Pomocný parametr pro počet rotací
 R19=0 " Pomocný pracovní parametr
 R20=0 " Čítač (pracovní parametr)

V bloku N20 se rozhodne podle vstupního parametru R14, zda se začne vrtat po řádcích nebo po sloupcích, tj. v jaké ose začne první pohyb. Použije se kód operace 19 – porovnání. Porovnává se vstupní parametru R14 (op1) s konstantou 0, kterou jsme si nadeklarovali do parametru R17 (op2). Výsledek aritmetické operace porovnání se uloží do parametru R06 (cíl). Pokud vyjde rovnost, uloží se do R06 hodnota 73, při nerovnosti hodnota 78.

N20 G26 R5=19141706 " je-li R14=0 pak se vrtá po řádcích (X)

V bloku N30 se nahradí formální parametr R06 skutečnou hodnotou, vypočtenou v předešlém bloku, tj. provede se G73 L1105 nebo G78 L1105. Funkce G73 provede skok na blok N1105, funkce G78 neprovede nic, program bude pokračovat na dalším řádku. V našem příkladu se provede skok na blok N1105.

N30 GR06 L1105 " G73 L1105 (= X) nebo G78 L1105 (= Y)

VRTANÍ PO SLOUPCÍCH (Y)

N105

N900 G70 „Konec vrtání po sloupcích

VRTÁNÍ PO ŘÁDCÍCH (X)

V bloku N1105 provede celkem 4 matematické operace (G29) podle řídicích parametrů R05 až R08. Úschova všech parametrů (kód operace 44) se provede do pomocné pracovní paměti systému. Tato operace se používá, pokud chceme zachovat např. vstupní parametry beze změny a přitom tyto parametry využít pro výpočty. Navrácení parametrů se provede kódem operace 45. (viz dále). Kódem operace 00 se zkopíruje vstupní parametr R13 (op1) do parametru R19 (cíl). Operand op2 není při kopírování využit a může zde být libovolné číslo (obvykle se dává 00). Dále je použit kód operace 42 – dekrementace. Touto operací se zmenší obsah parametru R12 (op1) o hodnotu op2. Pozor – operand op2 u této operace nemá význam čísla parametru (není to odkaz na parametr) ale je to přímo hodnota, o kterou se má op1 zmenšit! V prvním případě se zmenší obsah R12 o 2, v druhém případě se zmenší R13 o 1. Tímto výpočtem si přednastavíme počet opakování pojezdů od díry k díře v ose X a Y. U těchto operací musí být uvedeno znaménko mínus, tj. operandy jsou reprezentovány jako reálná čísla (0.004 – 0.002). Pokud by bylo uvedeno znaménko plus, provedlo by se $0.004 - 2.000 = -1.996$, což by byla chyba.

N1105 G29 R05=44000000 " Úschova všech parametrů
R06=00130019 " Ušchová počet řádek do R19
R07=-42120212 " R12 = R12 - 2 (počet opakování)
R08=-42130113 " R13 = R13 - 1 (počet opakování)

V bloku N1108 se vykoná pevný cyklus G81 (jsme nad první dírou). V parametru R09 je hodnota 81 (0.081) Parametry pro pevný cyklus musí být samozřejmě rovněž nastaveny. V tomto příkladě je pro jednoduchost neuvádíme.

N1108 GR09 " Vrtání první díry (PEVNY CYKL G81)

V bloku N1110 se vykoná inkrementální pohyb (G91) rychloposuvem v ose X o hodnotu rozteče, zadanou ve vstupním parametru R10, v našem případě o 100mm. Na konci pojezdu se provede opět vrtání pevným cyklem G81.

N1110 XR10 G91 GR09 G00

V bloku N1120 se provede odvolání PC funkcí G80, jinak by se provedlo v tomtéž místě ještě jedno vrtání. Dále se provede operace porovnání (kód 19). Proveďte se porovnání obsahu R20 (op1) což je pracovní čítač na začátku nastavený na 0, s obsahem parametru R13 (počet řádků upravený v bloku N1105 na počet opakování). Pokud nejsou všechny řádky hotové, dosadí se do R21 hodnota 78 (neprovede se skok v bloku N1140). Do bloku N1140 se ovšem program dostane, až se provede počet opakování (skok na blok N1110) podle parametru R12. Parametr R12 byl v bloku N1105 spočítán na počet opakování pro jednu řadu děr.

N1120 G73 L1110 QR12 G26 R5=19201321 G80 "if (PocetOpak=Citac)...Konec

Do bloku N1140 program projde po vykonání jedné řady.

N1140 GR21 L1170 G80

Posun přírůstkově na další řadu podle parametru R11 a vyvrtání díry cyklem G81 (R09=81).

N1150 YR11 GR09

V bloku N1160 se změní znaménko v R10 (osa X), protože druhou řadu pojede opačným směrem. Použije se operace 08 – inverzní hodnota operandu op1, výsledek se uloží do R10 (cíl). Dále se inkrementuje pracovní čítač v parametru R20 - používáný v bloku N1120 pro operaci porovnání. Nakonec se provede skok do bloku N1110 pro vrtání další řady, pokud se opakování neukončí.

N1160 G27 R5=08100010
R6=-41200020
G73 L1110 QR13 G80 "skok na dalsi radu

Bloky následující jsou určeny pouze pro výpočet návratu do výchozí polohy první díry.

Je-li počet řádek LICHÝ - vrací se v ose X i Y.

Je-li počet řádek SUDÝ - vrací se pouze v ose Y.

Počet řádek je v parametru v R19.

Pro rozhodnutí, je-li počet řádek sudý nebo lichý se použije kód operace 47, která do parametru R17 nastaví hodnotu 73(skok) pokud je počet řádek sudý nebo 78(pokračuje) pokud je počet řádek lichý. V našem příkladě je počet řádek lichý (3) a musí se z díry 12 vrátit na 1 v obou osách. Pokračujeme tedy na bloku N1185.

N1170 G26 R5=47190017 G80 " jestliže (R19 = sudý) pak R17=73 jinak R17=78
N1185 GR17 L1200 R15=0 " skok na L1200 při sudém počtu řádek

V bloku N1190 se vypočte návrat v ose X, t.j. přepíše nulu v R15 vypočtenou hodnotou. Použije se kód násobení (opět jako reálné číslo – znaménko minus).

N1190 G28 R5=-41120012 " Z počtu opakování spočte počet roztečí sloupců
R6=-03121015 " R12*R10=R15 (počet roztečí)*(rozteč X) = dráha

R7=08150015**" Změna znaménka dráhy**

Podobným způsobem vypočte návrat v ose Y.

$R13 * R11 = R14 \dots (\text{počet řádek} - 1) * (\text{rozteč } Y) = \text{přírůstek návratu v } Y$.

N1200 G27 R5=-03131114 R6=08140014

Vypočtené dráhy jsou uloženy v R14 a R15.

N1210 YR14 XR15**" návrat do výchozího bodu**

V bloku N1230 vrátí zpět parametry, které uschoval v bloku N1105. Pokud op1, op2 i cíl je 0, provede navrácení všech 96 parametrů. Parametry op1, op2 a cíl se dá řídit, které parametry (a kam) se mají navrátit.

N1230 G90 G26 R05=45000000**" Vráť zpět uschované parametry****N1300 G70****" konec makrocyclu**

*

12

12. PROGRAMOVÁNÍ A DEFINICE PEVNÝCH CYKLŮ

12.1 Tvorba pevných cyklů

Pevné cykly jsou normou doporučené makrocykly, které provádějí standardní vrtací, frézařské nebo soustružnické cykly. Chování pevného cyklu se ovlivňuje pouze nastavením parametrů, podle kterých pevný cyklus provádí svoji činnost. Zásady pro tvorbu pevných cyklů jsou stejné pro frézky i soustruhy.

Pevné cykly jsou v paměti systému CNC8x6 uchovány jako normální partprogram a možno je vytvářet a editovat. V zálohované paměti (na disku) může být i více souborů s pevnými cykly, ale systém při volbě partprogramu načte ten soubor, jehož jméno je uvedeno v parametru **\$17 konfiguračního souboru CNC836.KNF**. Další podmínkou automatického načtení pevných cyklů je nastavení **3. dekády strojní konstanty 99 na hodnotu 1**. (viz. návod na obsluhu kapitoly: "Strojní konstanty" a "Konfigurace").

Tvorbu a editaci pevných cyklů možno provádět ze systému CNC8x6. Doporučuje se využít systémový editor (vstup ze základního menu přes tlačítko Systém a Systém-edice), neboť při editaci a uložení souboru s pevnými cykly v prostředí partprogramů (vstup ze základního menu přes tlačítko Programy a Edice) se může – pokud byla po zapnutí systému provedena volba libovolného partprogramu - hlásit chyba „Partprogram je již v paměti“, neboť pevný cyklus se načítá do paměti automaticky při volbě programu. Při editaci pevných cyklů přes systémový editor se neprovede po uložení syntaktická kontrola. Eventuelní chyby se proto projeví až při volbě programu.

Soubor s pevnými cykly začíná klíčovým slovem: " \$PC " a končí znakem *.

Při tvorbě pevných cyklů se často využívá aritmetika parametrů. Parametrické operace jsou podrobně vysvětleny v kapitole "Aritmetika parametrů".

Pro programování pevných cyklů platí následující pravidla:

Pohyby náležející pevným cyklům se provádí v bloku, kde je prvně programovaná jedna z funkcí G81 – G89 a to za interpolacemi, ale před závěrečnými M-funkcemi. Vrtací cyklus se pak provede v každém dalším bloku (funkce G81-G89 již nemusí být programovány) až do bloku, kde je programována funkce odvolání pevného cyklu G80. V bloku s G80 se již vrtací cyklus neprovede.

Pevné cykly nemění programované technologické M-funkce, G-funkce ani hodnoty dalších adres (např. číslo tabulky korekcí D, rychlost F apod.) a parametrů, nebo-li po vykonání pevných cyklů se vrátí hodnoty funkcí, adres i parametrů, které byly programovány před voláním pevného cyklu, i když je pevný cyklus změněn.

Například v pevném cyklu G84 je použita reverzace vřetena. Před voláním pevného cyklu se naprogramuje např. M3, pevný cyklus po dojetí do dna díry reverzuje vřeteno, tj. vyšle M4, ale po ukončení pevného cyklu (přesněji po startu bloku následujícího za pevným cyklem) se opět roztočí vřeteno ve směru M3. Podobně pokud se v pevném cyklu např. změni rychlost F, po ukončení pevného cyklu bude rychlost nastavena na F, které bylo programováno před vyvoláním pevného cyklu.

Pozn.:

Toto je hlavní a prakticky jediný rozdíl mezi pevnými cykly a makrocykly. Po vykonání makrocyklů zůstanou hodnoty G a M funkcí i adres takové, jak je eventuálně nastaví případně změni makrocykly.

Příklad:

```

N10 ...
N20 X50 G0
N30 R26=400 R27=200 R30=5.0 R31=-85.0 R32=25.0 „deklarace parametrů pro pevný cyklus
N40 M3 M41
N50 X100 G00 G81
N60 X200
N70 X300
N80 X400
N90 X500
N100 X600 G80

```

Pevný vrtací cyklus G81 se provede poprvé na konci bloku N50 (po dojetí na míru X100) a potom v každém dalším bloku. Poslední vrtací cyklus se provede v bloku N90 po dojetí na míru X500, neboť v bloku N100 je již programováno odvolání pevných cyklů funkcí G80. Pozor na umístění funkce volání pevného cyklu (zde G81). Nemůže být již např. v bloku N30 (plnění parametrů) nebo N40 (zařazení M3). Pokud by byla funkce G81 např. v bloku N30, provedlo by se vrtání již v tomto bloku (na poslední najeté míře X50) a další vrtání v téže poloze X v bloku N40.

12.2 Příklady frézkových vrtacích pevných cyklů

Vrtací pevné cykly jsou dodávány výrobcem, nicméně uživatel si je může upravit, případně si vytvořit další vlastní. S výhodou se dá využít aritmetika parametrů, popsáná v samostatné kapitole.

V systémech CNC8x6 se (od verze 30.29) dodávají standardní pevné cykly v souboru **PCYKLYDG.NCP**. Starší verze používají soubor **PEVNECYK.NCP** případně jiný upravený. Které pevné cykly bude systém používat je dáno nastavením parametru **\$17 v souboru CNC836.KNF**, kde je zadán **název souboru s pevnými cykly**.

Pokud se využívá dialogová tvorba partprogramu a v ní volba pevných cyklů, doporučuje se využívat **PCYKLYDG.NCP**

Popis pevných cyklů ze souboru **PCYKLYDG.NCP**

Pevný cyklus začíná v referenční rovině RA. Hloubka vrtání je prováděna do roviny RB. Pevný cyklus končí v rovině RC. Souřadnice referenčních rovin RA, RB a RC musí být programovány v **absolutních souřadnicích**. Přírůstek hloubky q pro vrtání hlubokých děr je programován přírůstkově. Vlastní prováděcí program pevných cyklů je v systému vytvořen v parametrické formě. V partprogramu, ze kterého je příslušný pevný cyklus volán, musí být deklarovány souřadnice rovin, přírůstek pro vrtání hlubokých děr, časová prodleva, velikost posuvu a otáček jednotlivými parametry takto:

Parametr	Popis
R26	Deklaruje rychlost posuvu (adresu F), který je v rámci PC prováděn
R27	Deklaruje velikost otáček (adresu S) pro daný PC (v souladu s dříve zadanou funkcí M41 až M44)
R28	Deklaruje přírůstek q pro vrtání s výplachem (G83) a vrtání s odlamováním třísky (G86)
R29	Deklaruje velikost časové prodlevy (funkce Q)
R30	Deklaruje souřadnici roviny RA („odkud vrtat“)
R31	Deklaruje souřadnici roviny RB („kam vrtat“)
R32	Deklaruje souřadnici roviny RC („kam vyjet“)

Doporučení pro zadávání parametrů:

Při zadávání parametrů je důležité si uvědomit vliv desetinné tečky na hodnotu parametru. Hodnota parametru je obecně bezrozměrné číslo. Rozměr získá až přiřazením ke konkrétní adrese. Pokud není uvedena desetinná tečka, má parametr $R_{xx}=200$ hodnotu 200. Pokud bude hodnota 200 přiřazena k parametru, který určuje otáčky, bude rozměr 200ot/min. Pokud bude hodnota 200 přiřazena k parametru, který určuje rychlost posuvu, bude rozměr 200mm/min, pokud bude hodnota 200 přiřazena k parametru, který určuje délkové míry nebo polohu, bude rozměr 200mikronů (pozor – nikoli milimetrů!).

Zapsání samotné desetinné tečky si lze představit jako přidání tří nul k hodnotě parametru. $R_{xx}=200$. by tedy znamenalo 200000 otáček, 200000mm/min (=200m/min) a 200000mikrometrů (=200mm).

Z praktického hlediska se tedy doporučuje psát hodnotu parametru pro otáčky bez desetinné tečky, hodnotu parametru pro rychlost buď bez tečky pokud chceme zadávat v mm/min nebo s tečkou pokud chceme zadávat v m/min a hodnotu parametru pro délky a polohy s desetinnou tečkou, neboť je vhodnější zadávat míry v mm než v mikronech.

Možnosti zadávání parametrů pro pevné cykly uvádí následující tabulka.

Parametrické zadání	Možnosti zápisu hodnoty do parametru	Rozměr zadané hodnoty	Poznámka
FR26 (rychlost)	R26=200	200 mm/min	Pro milimetrový posuv (G94)
	R26=0.200	0,2 m/min (=200mm/min)	Pro milimetrový posuv (G94)
	R26=200	200 mikronů/ot	Pro otáčkový posuv (G95)
	R26=0.200	0,2 mm/ot	Pro otáčkový posuv (G95)
SR27 (otáčky)	R27=300	300 ot/min	
	R27=0.300	300 ot/min	
ZR28 (přírůstek)	R28=25.	25 mm	
	R28=25.0	25 mm	
	R28=25000	25000mikronů (=25mm)	
QR29 (čas. prodl.)	R29=250	250 * 10ms = 2,5 sec.	Jedna jednotka = 10ms
	R29=0.250	250 * 10ms = 2,5 sec.	
ZR30 (Rovina)	R30=5. 5	5,5mm	
	R30=5500	5500 mikronů (=5,5mm)	
ZR31 (Rovina)	dtto		
ZR32 (Rovina)	dtto		

Pozn.:

Dialogové zadávání pevných cyklů formátuje zadání všechny parametrů do tvaru s desetinnou tečkou. Podrobněji viz kapitola Dialogová tvorba.

Příklad použití pevného cyklu :

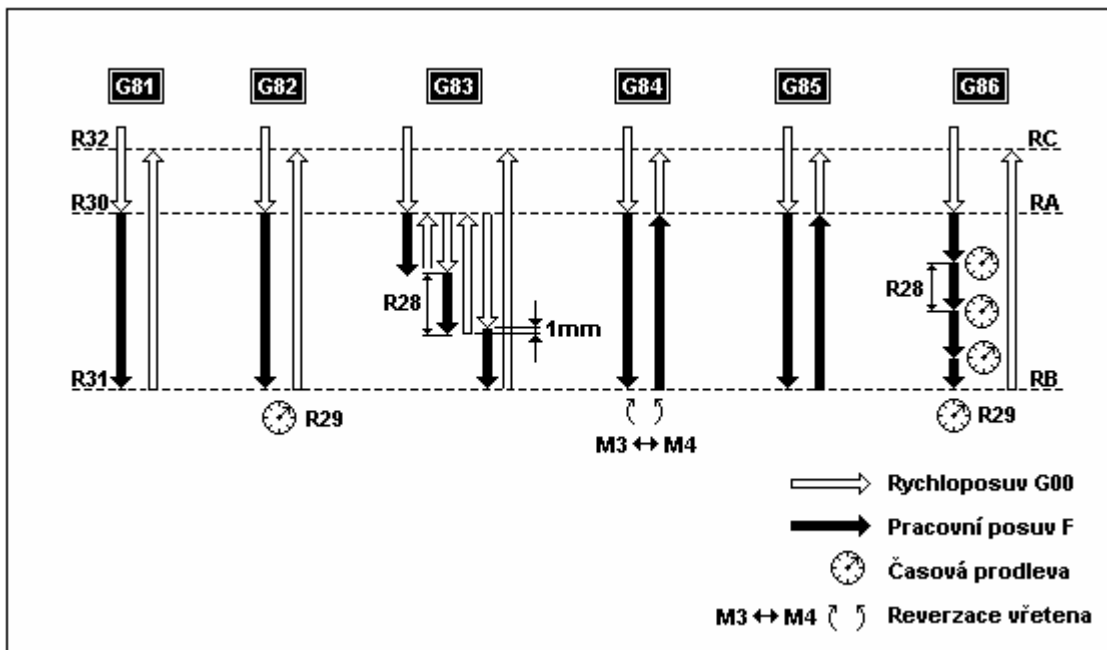
(pro pevný cyklus G82 se zadá posuv 460mm/min, 1000ot/min, časová prodleva 3 sec. Vrtá se z polohy 5.0 mm do polohy -50.0 mm a po ukončení se vyjede na míru 30.0. Celkem se vyvrtají 3 díry. První díra v bloku N4 v poloze X100. Tato poloha se najela již v bloku N1. Druhá díra se vyvrtá v poloze X200 (přejezd v bloku N5 je programován přírůstkově G91 o 100mm). Třetí díra se vyvrtá v poloze X300 (opět přírůstkově o 100 dál). Přírůstkové programování G91 se vztahuje pouze na přejezdy. Pevný cyklus má parametry rovin zadán vždy absolutně! Před pevným cyklem je zadáno S100 ot/min. Pevný cyklus změni otáčky (parametrem R27) na 1000 ot/min. Po ukončení pevného cyklu se otáčky vrátí na S100ot/min tj. přejezdy mezi jednotlivými vrtacími cykly v blocích N5 a N6 budou provedeny se 100ot/min

```

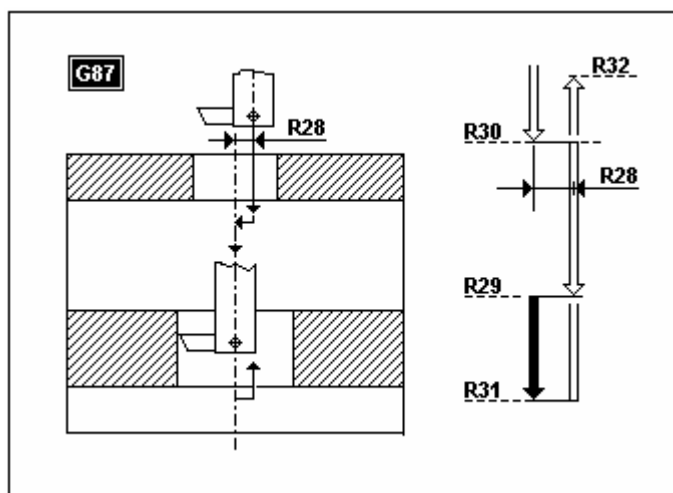
N1    G00    G90    X100.
N2    Z50.    S100    M43    M04
N3    R26=460 R27=1000 R29=300 R30=5. R31=-50. R32=30.
N4    G82
N5    G91    X100.
N6    X100.
N7    G90    Z500.    M05    G80

```

Pevné cykly – soubor PCYKLYDG.NCP



Pevný cyklus G87 s „odskokem“ v ose Y



Výpis vrtacích pevných cyklů ze souboru PCYKLYDG.NCP, který používá také dialogová tvorba partprogramů.

Pozn.1: V dialogové tvorbě partprogramů není zahrnut pevný cyklus G87.

Pozn.2: V systémech může být aktualizovaný stav souboru s pevnými cykly.

```
" $PC - PEVNE CYKLY FREZARSKE VRTACI V OSE Z
" (Kompatibilni s dialogovou grafikou od verze panelu 30.29)
" Datum revize: 20.2.2001
"
" Pozn1.:
" V pevnem cyklu G84 pouzivanem pro zavitevani zavitevaci
```

```

"   hlavičkou je použito implicitně G94 (milimetrový posuv)
"   Pokud chcete používat otákový posuv, změňte G94 na G95
"
"   Pozn2.:
"   Pokud nechcete zadávat posuvovou rychlost F a otáčky S
"   parametricky (R26 a R27) před voláním pevného cyklu,
"   zrušte ve všech pevných cyklech FR26 a SR27 !
"   Rychlost F a otáčky S se pak zadají před voláním cyklu.
"
"   Použité parametry:
"
"   R26 = Rychlost vrtání [mm/min]
"   R27 = Otáčky [ot/min]
"   R28 = Prírůstek vrtání (nebo odskok v Y pro G87)
"   R29 = Časová prodleva (1=10ms) (nebo mezirovina RD pro G87)
"   R30 = Rovina RA (absolutní poloha odkud vrtat)
"   R31 = Rovina RB (absolutní poloha kam vrtat)
"   R32 = Rovina RC (absolutní poloha kam vyjet po ukončení PC)
"
"-----
" Vrtání bez časové prodlevy
"-----
N1    G79 L81
N2    G00 G90 ZR30   FR26 SR27
N3    G01           ZR31
N4    G00 G70 ZR32
"-----
" Vrtání s časovou prodlevou na konci díry
"-----
N1    G79 L82
N2    G00 G90 ZR30   FR26 SR27
N3    G01           ZR31   G04 Q929
N4    G00           ZR32
N5    G70
"-----
" Vrtání s proplachováním
"-----
N1    G79 L83
N10   FR26 SR27
      G27 R5=15313024      "if R31 > R30 then R24=73
      R6=07280028          "abs. hodnota přírůstku vrtání
N20   GR24 L240
" Vrtání ve směru minus
N40   G27 R5=00300029
      R6=00300025
      R24=1.000
N50   G26 R5=16312923      " if R31 >= R29 then R23 je 73 else 78
N60   GR23 L140
N70   ZR25 G00 G26 R5=02292829 " R29=R29-R28
N80   G26 R5=15293123      " if R31 > R29 then R23 je 73 else 78
N90   GR23 L110 Q9998      " Obskoci blok 100
N100  G27 R5=00310029      " R29=R31
      R6=00320030          " poslední vrt.- vyjede dle R32
N110  ZR29 G01
N120  ZR30 G00
N130  G26 R5=01292425 G73 L50 Q9998 " R25=R29+R24 (R24 = 1mm)
N140  G70
" Vrtání ve směru plus
N240  G27 R5=00300029 R6=00300025 R24=1.000 "R30 -> R29
N250  G26 R5=18312923      " if R31 <= R29 then R23 je 73 else 78
N260  GR23 L340

```

```

N270 ZR25 G00 G26 R5=01292829 " R29 = R29 + R28
N280 G26 R5=17293123 " if R31 < R29 then R23 je 73 else 78
N290 GR23 L310 Q9998 " Obskoci blok 300
N300 G27 R5=00310029 " R29 = R31
      R6=00320030 " posledni vrt.- vyjede dle R32
N310 ZR29 G01
N320 ZR30 G00
N330 G26 R5=02292425 G73 L250 Q9998 " R25 = R29 - R24 (R24 = 1mm)
N340 G70
"-----
" Vrtani s reverzací vřetene (zavitování hlavičkou)
"-----
N1 G79 L84
N2 G00 G90 ZR30 FR26 SR27
      R08=4 " Pomocná pro porovnání na M4
      R09=3 " Pomocná pro porovnání na M3
      G26 R05=33100016 " Program 10 - 16
      R10=49510220 " Zapiš 2.skup M-fci do R20
      R11=20200900 " if R20=R09 (R20=M3?) then pokračuj
      R12=00080007 " Do R7 da M4
      R13=22000000 " Konec podmínky
      R14=20200800 " if R20=R08 (R20=M4?) then pokračuj
      R15=00090007 " Do R7 da M3
      R16=22000000 " Konec podmínky
N3 G01 ZR31 G94 M49 " G95 pro otákový posuv
N4 MR07 ZR30 " reverzace vřetena (R07)
N5 G70 G00 ZR32
"-----
" Vrtání i vyjezd pracovním posuvem
"-----
N1 G79 L85
N2 G00 G90 ZR30 FR26 SR27
N3 G01 ZR31
N4 ZR30
N5 G70 G00 ZR32
"-----
" Vrtání s odlamováním třísky časovou prodlevou
"-----
N1 G79 L86
" Rozhoduje o směru vrtání - if R30 > R31 - vrta do minusu
" - if R30 < R31 - vrta do plusu
" Prírůstek vrtání se zadává vždy kladně
"
N3 FR26 SR27
      G27 R5=07280028 "abs. hodnota přírůstku vrtání
      R6=16313034 "if R31<R30 then skok
N4 GR34 L100
" smer do minusu
N10 G00 G90 ZR30 "najede rychloposuvem na rovinu RA
      G26 R5=00300024 "R30 -> R24
N20 G27 R5=02242824 "R24=R24-R28 (v cyklu odcita přírůstek)
      R6=15312423 "if R31>R24 then R23=73 else R23=78
N30 GR23 L60 "skok G73 nebo prázdná instrukce G78
N40 G01 ZR24 G04 Q929 "vrta dle R24, pak časová prodleva
N50 G73 L20 Q9998 "skok na blok N20
N60 ZR31
N70 G70 G00 ZR32
" Smer do plusu
"
N100 G00 G90 ZR30 "najede rychloposuvem na rovinu RA
      G26 R5=00300024 "R30 -> R24

```

```

N200 G27 R5=01242824      "R24=R24+R28 (v cyklu pricita prirustek)
      R6=17312423        "if R31<R24 then R23=73 else R23=78
N300 GR23 L600            "skok G73 nebo prazdna instrukce G78
N400 G01 ZR24 G04 QR29    "vrta dle R24, pak casova prodleva
N500 G73 L200 Q9998       "skok na blok N200
N600 ZR31
N700 G70 G00 ZR32
"
"-----
" Vrtani s odskokem v ose Y
"-----
" R28 = odskok v ose Y
" R29 = mezirovina RD
" ostatni parametry stejne s ostatnimi PC
"
N1 G79 L87
N2 G26 R5=49510224      "R24=uschova smer otaceni vretene
      G00 G90 ZR30      "Rychloposuvem na rovinu RA
N3 G26 R5=08280025      "R25=inverzni hodnota odskoku R28
      G91 YR28          "Odskok v ose Y mimo osu vrtani
      M19              "Orientovany stop
N4 G90 ZR29            "Absolutne na mezirovinu RD
N5 G91 YR25            "Navrat v Y do osy vrtani
N6 MR24 SR27           "Roztoceni vretene
N7 G01 G90 ZR31 FR26    "Vrtani do roviny RB
N8 M19                 "Orientovany stop
N9 G00 G91 YR28         "Odskok v ose Y
N10 G90 ZR32            "Vyjezd do roviny RC
N11 G70 G91 YR25        "Navrat v Y do osy vrtani
"-----
" Prazdny pevny cykl - pro uzivatele
"-----
N1 G79 L88
N2
N3 G70
"-----
" Prazdny pevny cykl - pro uzivatele
"-----
N1 G79 L89
N2
N3 G70
*
```

12.3 Možnosti úprav pevných cyklů

Uživatel si může dodávané pevné cykly upravit podle potřeby či zvyklostí. Uvedeme některé možnosti, se kterými jsme se setkali v praxi. Někteří uživatelé nechtějí zadávat rychlost F a otáčky S parametricky v pevném cyklu (parametry R26 a R27), ale chtějí je zadat před voláním pevného cyklu klasicky naprogramováním S a F. V tomto případě stačí z pevných cyklů odstranit všechny zápisy FR26 a SR27, jak je uvedeno na příkladu pevného cyklu G81:

Původní stav G81:

N1 G79 L81
N2 G00 G90 ZR30 FR26 SR27
N3 G01 ZR31
N4 G00 G70 ZR32

Upravený stav:

N1 G79 L81
N2 G00 G90 ZR30
N3 G01 ZR31
N4 G00 G70 ZR32

Další možností je použít v pevném cyklu G84 otáčkový posuv (G95) místo milimetrového (G94), který má tu výhodu, že lze změnit otáčky, pokud nevyhovují řezné podmínky, aniž by se změnily geometrické poměry při řezání závitu. Tuto výhodu však do jisté míry při G94 eliminuje používání závitovacích hlaviček, které svým napružením mohou překlenou rozdíl v rychlosti posuvu a otáčkách při změně otáček. Změna rychlosti %F je blokována funkcí M49.

Starší typy pevných cyklů nevyužívaly roviny RC pro výjezd z pevného cyklu a končily cyklus ve stejné rovině, kde cyklus začínal, tj. v rovině RA. Pokud byste měli k dispozici hotové partprogramy, které s rovinou RC nepočítají (nepoužívají parametr R32), stačí v pevných cyklech nahradit všechny ZR32 za ZR30.

Další možností je vlastní tvorba pevného cyklu, který nemusí ani souviset s vrtáním, ale je možné si vytvořit např. pevný cyklus pro nájezd do polohy pro výměnu nástroje a jeho výměnu. V souboru PCYKLYDG.NCP jsou k dispozici pro uživatele volné cykly G88 a G89. Můžete se inspirovat následujícím příkladem:

N10 G79 L88
 N20 G54 G0 G40 Z200. M5 „odjede do bezpečnostní polohy a stopne vřeteno
 N30 G59 Y0 „odjede do polohy pro výměnu, poloha zadána v posunutí G59
 N40 TR12 „vyhledá nástroj, jehož číslo je v parametru R12
 N50 M06 „provede výměnu nástroje
 N60 G70

12.4 Pevné cykly pro soustruhy

Pro tvorbu a používání pevných cyklů pro soustruhy platí stejné zásady jako pro pevné cykly frézarské. Podrobněji popíšeme případné rozdíly.

Pro soustruhy se dodávají tři soubory s pevnými cykly: PEVNECY4.NCP, PEVNECY5.NCP a PEVNECY6.NCP. Soubory PEVNECY5.NCP a PEVNECY6.NCP obsahují navíc oproti souboru PEVNECY4.NCP možnost střídavého závitování, přičemž PEVNECY5.NCP je určen pro používání průměrového programování a soubor PEVNECY6.NCP pro poloměrové programování.

Pozn.:

Soubor PEVNECY4.NCP se musí zeditovat „ručně“ podle toho, používá-li se poloměrové nebo průměrové programování. Implicitně je nastaven na průměrové programování.

Řádky, které se upravují mají následující tvar:

R14=+01333333 " Pro průměrové progr: R33=R33+R33 (01333333) !

Pokud chceme poloměrové programování, musí se upravit takto:

R14=+00000000 " Pro průměrové progr: R33=R33+R33 (01333333) !

Pro zadávání parametrů platí stejné zásady, týkající se použití desetinné tečky, jako u vrtacích cyklů pro frézky. Týká se to navíc také např. počtu třisek, které se zadává bez desetinné tečky (případně s tečkou ale na místě setin a tisícín – viz dále). U soustružnických pevných cyklů je důležité také znaménko u některých parametrů, které udává směr posuvu. Při průměrovém programování se parametry určující míry v ose X programují průměrově (např. i tloušťka třísky!).

V příkladech jsou tyto možnosti označeny takto:

(+/-) Znaménko určuje směr

(Ø) Průměrové zadání hodnoty

12.5 Příklady soustružnických pevných cyklů

12.5.1 Podélné hrubování - G81

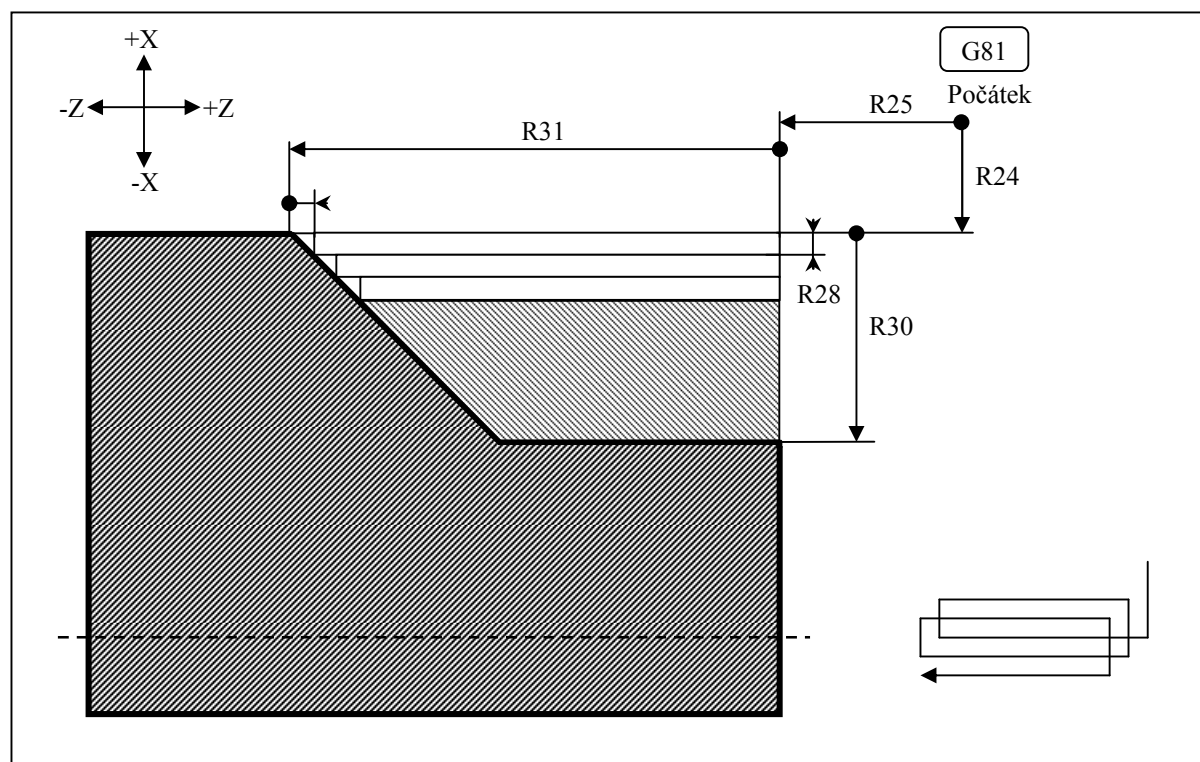
Použité parametry:

R24	(+/-)	(Ø)	Kolik mm před dílcem v ose X začíná pevný cyklus
R25	(+/-)		Kolik mm před čelem v ose Z začíná pevný cyklus
R26			Rychlost posuvu
R27			Počet opakovaných přejezdů po konečné dráze (začištění)
R28		(Ø)	Síla (tloušťka) třísky v mm
R29	(+/-)		Zkrácení vzdálenosti Z na jednu třísku
R30	(+/-)	(Ø)	Velikost oblasti hrubování v ose X
R31	(+/-)		Velikost oblasti hrubování v ose Z

Symbol (+/-) v tabulce znamená, že uvedená míra znaménkem určuje směr. Symbol (Ø) v tabulce znamená, že uvedená míra se při průměrovém programování programuje průměrově a při poloměrovém programování poloměrově.

Hrubování je možno použít ve všech čtyřech zrcadlových rovinách. Roviny hrubování jsou určeny znaménky parametrů R30 a R31. Znaménka parametrů R24, R25 a R29 jsou také závislá na použité rovině hrubování. Možno říct, že všechny tyto parametry (R24, R25, R29, R30, R31) jsou programovány přírůstkově vzhledem k počátečnímu bodu, podobně jak je tomu u adres I a J při programování kružnice.

Na obrázku je příklad hrubování, ve kterém se předpokládá, že parametry R24, R25, R30, R31 mají záporné hodnoty.



Podélné hrubování G81 se začíšťováním (průměrové programování)

Výpis pevného cyklu G81 ze souboru PEVNECY5.NCP

```

" HRUBOVANI PODELNE - G81
N1  G79  L81
N2  G90  G24          " Absolutne
                        R00=+0
                        R01=+1
                        R02=+15113010
                        R03=+17113010
                        " Modifikace instrukce porovnani

G26  R05=+33070021    " PROGRAM 7 - 21

                        R07=+00280039
                        R08=+00800085
                        R09=+31300000
                        R10=+00020003
                        R11=+08280028
                        R12=+22000000
                        R13=+26010283
                        R14=+01333333
                        R15=+01332445
                        R16=+01453030
                        R17=+01452811
                        R18=+01342538
                        R19=+01383131
                        R20=+00310044
                        R21=+00330037
                        " Uschova R39=R28
                        " Uschova R35,R36=R30,R31
                        " IF R30<R00 THEN
                        "     R03=R02
                        "     R28=-R28
                        " ENDIF
                        " Naplni X prog. do R33 a Z prog. do R34
                        " Pro prumerove progr: R33=R33+R33 (01333333)
                        " R45=R33+R24
                        " R30=R45+R30
                        " R11=R45+R28 (prvni triska)
                        " R38=R34+R25
                        " R31=R38+R31
                        " R44=R31 (uchova pro zacistovani)
                        " R37=R33

N50  G00  FR26          XR11  " Jede v X na miru
      G26  R05=+01312931      " R31=R31+R29 Zkraceni
N6   G01          ZR31      " Triska
      G27  R05=+00370009      " R09=R37
      R06=+00030007          " R07=R03 (modifikovana instrukce)
N7   G29  R05=+00110037      XR09  " Vyjezd v X
      R06=+01112811          " Kopirovani R11 do R37
      R08=+00340009          " R11=R11+R28 (dalsi triska)
      R09=R34                " Je-li R11 > R30 pak R10=73 jinak R10 je 78
N8   G00          ZR09      " Navrat v Z
      GR10 L50 Q9998          " Podmineny skok na blok N50

N10  G29  R05=+02464646      " R46=0
      R06=+19274610          " Je-li R27 = R46(0) pak R10=73 jinak R10 je
78                                "
      R07=+07350040          " R40=abs(R35) (R30)
      R08=+00270047          " R47=R27
N11  GR10 L30 Q9998          " Skok je li bez zacistovani
      G28  R05=+03294090      " R40,41=R29.R40
      R06=+04903940          " R40=R40,41/39 (28)
      R07=+01444009          " R09=R44+R40
N12  G00  FR26          XR30  " Jede v X na posledni miru
      G26  R05=+00440011      " R11=R44
N13  G01          ZR09      " Zacistovaci triska
      G26  R05=+00450010      " R10=R45
N14          XR10  ZR11      " Celo
      G26  R05=+00330010      " R10=R33
N15          XR10
      G28  R05=+00340010      " R10=R34
      R06=+02470147          " R47=R47-R01 dekrement

```

```

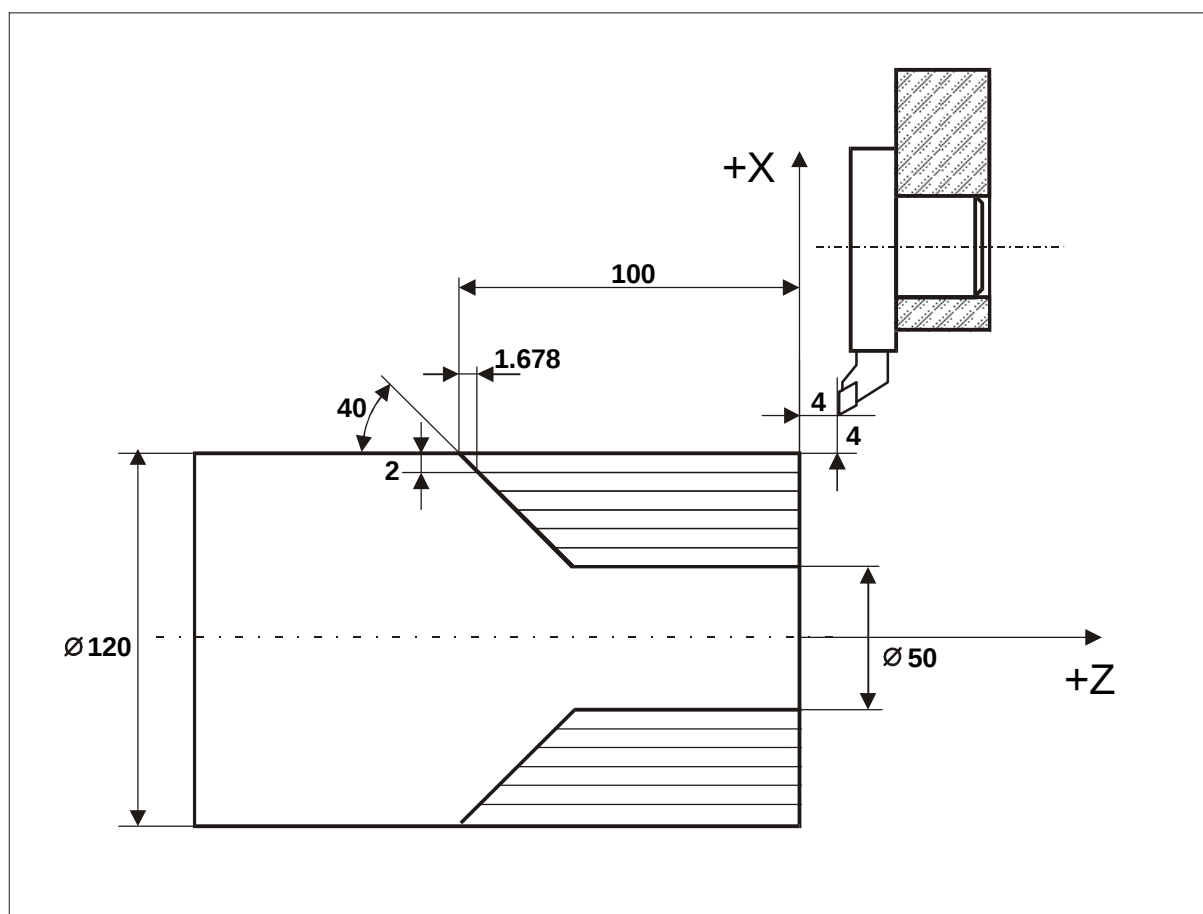
      R07=+15474611      " Je-li R47 > R46(0) pak R11=73 jinak R11 je
78
      N16 G00            ZR10  " Navrat v Z
          GR11 L12      Q9998
N17 G70

N30 G00  FR26          XR30  " Jede v X na posledni miru
      G26  R05=+01312931      " R31=R31+R29 Zkraceni
N31 G01  G01          ZR31  " Posledni triska
      G26  R05=+00330009      " R09=R33
N32          XR09        " Navrat v X
      G26  R05=+00340009      " R09=R34
N33 G00          ZR09  " Navrat v Z
N35 G70
"

```

12.5.2 Příklad použití pevného cyklu G81 pro podélné hrubování

Na následujícím obrázku je okótovaný výkres součásti. Uvedený příklad programu předpokládá průměrové programování, souřadný systém dle obrázku (kladný směr osy X nahoru, Z vpravo). Výchozí bod pevného cyklu zobrazuje soustružnický nůž.



Program pro hrubování by byl zapsán takto:

```

N10 G54 &1100 D1 T1 M3 S500      "Hlavní věta
N20 X0 Z300
N30 X128 Z4          " Nájezd do výchozí polohy
N40 R24= -8.000      " Kolik mm před dílcem v X, (znaménko = směr)
      R25= -4.000      " Kolik mm před dílcem v Z, (znaménko = směr)
      R26=200          " Rychlost posuvu
      R27=0.002       " Počet opakovaných přejezdů po konečné dráze
      R28=4.000       " Tloušťka třísky v mm
      R29=1.678       " Zkrácení vzdálenosti Z na jednu třísku (znaménko = směr)
      R30= -70.000    " Velikost hrubování v ose X, (znaménko = směr)
      R31= -100.000   " Velikost hrubování v ose Z, (znaménko = směr)
      G81
N50 G80
N60 M30

```

Pozn.:

Při průměrovém programování se míry v X programují průměrově. Tloušťka třísky 2mm se proto programuje R28=4.0, podobně ještě parametry R24 a R30.

Při poloměrovém programování se uvedené parametry programují poloměrově, tj. byly by dvakrát menší než v uvedeném příkladě (R28=2.0).

Uvedeným způsobem se programují i ostatní soustružnické pevné cykly.

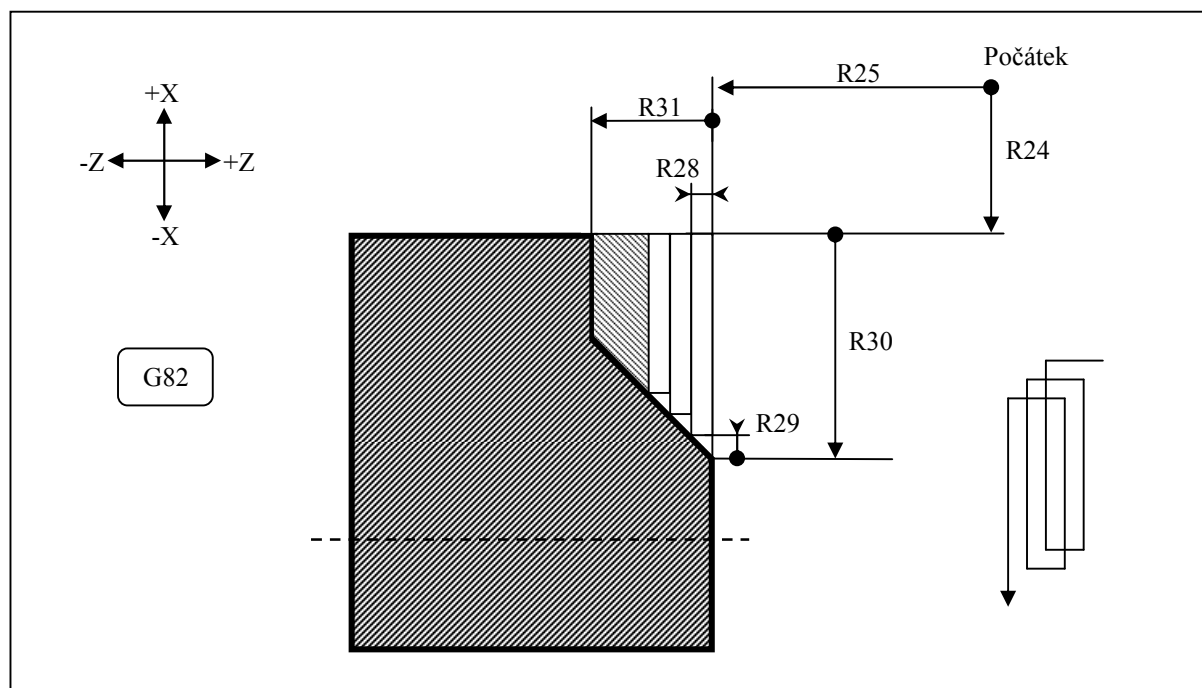
12.5.3 Příčné hrubování - G82

Použité parametry:

R24	(+/-)	(Ø)	Kolik mm před dílcem v ose X začíná pevný cyklus
R25	(+/-)		Kolik mm před čelem v ose Z začíná pevný cyklus
R26			Rychlost posuvu
R27			Počet opakovaných přejezdů po konečné dráze (začištění)
R28		(Ø)	Síla (tloušťka) třísky v mm
R29	(+/-)		Zkrácení vzdálenosti X na jednu třísku
R30	(+/-)	(Ø)	Velikost oblasti hrubování v ose X
R31	(+/-)		Velikost oblasti hrubování v ose Z

Hrubování je možno použít ve všech čtyřech zrcadlových rovinách. Roviny hrubování jsou určeny znaménky parametrů R30 a R31. Znaménka parametrů R24, R25 a R29 jsou také závislá na použité rovině hrubování. Možno říct, že všechny tyto parametry (R24, R25, R29, R30, R31) jsou programovány přírůstkově vzhledem k počátečnímu bodu, podobně jak je tomu u adres I a J při programování kružnice.

Na obrázku je příklad hrubování, ve kterém se předpokládá, že parametry R24, R25, R30, R31 mají záporné hodnoty.



Příčné hrubování G82 se začišťováním (průměrové programování)

Výpis pevného cyklu G82 ze souboru PEVNECY5.NCP

"

N1	G79	L82	
N2	G90	G24	" Absolutne
		R00=+0	
		R01=+1	
		R02=+15113110	" Modifikace instrukce porovnani
		R03=+17113110	
G26	R05=+33070021		" PROGARM 7 - 21

R07=+00280039	" Uschova R39=R28
R08=+00800085	" Uschova R35,R36=R30,R31
R09=+31310000	" IF R31<R00 THEN
R10=+00020003	" R03=R02
R11=+08280028	" R28=-R28
R12=+22000000	" ENDIF
R13=+26010283	" Naplni X prog. do R33 a Z prog. do R34
R14=+01333333	" Pro prumerove progr: R33=R33+R33 (01333333) !
R15=+01342545	" R45=R34+R25
R16=+01453131	" R31=R45+R31
R17=+01452811	" R11=R45+R28 (prvni triska)
R18=+01332438	" R38=R33+R24
R19=+01383030	" R30=R38+R30
R20=+00300044	" R44=R30 uchova pro zacistovani
R21=+00340037	" R37=R34

```

N50   G01   FR26           ZR11  " Jede v Z na miru
      G26   R05=+01302930      " Zkraceni R30=R30+R29
N6     G27   R05=+00370009      " Triska
      R06=+00030007      " R09=R37
      ZR09   " R07=R03 (modifikovana instrukce)
N7     G29   R05=+00110037      " Vyjezd v Z
      R06=+01112811      " R37=R11
      R08=+00330009      " R11=R11+R28 (dalsi triska)
      ZR09   " Je-li R11 > R31 pak R10=73 jinak R10 je 78
      " R09=R33
N8     G00           XR09  " Navrat v X
      GR10  L50      Q9999  " Podmineny skok na blok N50

N10    G29   R05=+02464646      " R46=0
      R06=+19274610      " Je-li R27 = R46(0) pak R10=73 jinak R10 je 78
      R07=+07360040      " R40=abs(R36) (R31)
      R08=+00270047      " R47=R27
N11    GR10  L30      Q9999  " Skok je li bez zacistovani
      G28   R05=+03294090      " R40,41=R29.R40
      R06=+04903940      " R40=R40,41/39 (28)
      R07=+01444009      " R09=R44+R40
N12    G01   FR26           ZR31  " Jede v Z na posledni miru
      G26   R05=+00440011      " R11=R44
N13           XR09  " Zacistovaci triska
      G26   R05=+00450010      " R10=R45
N14           ZR10  XR11  " Celo
      G26   R05=+00330010      " R10=R33
N15           ZR10
      G28   R05=+00340010      " R10=R34
      R06=+02470147      " R47=R47-R01 dekrement
      R07=+15474611      " Je-li R47 > R46(0) pak R11=73 jinak R11 je 78
N16    G00           XR10  " Navrat v X
      GR11  L12      Q9999  " Podminene opakovani
N17    G70           " Konec

N30    G01   FR26           ZR31  " Jede v Z na posledni miru
      G26   R05=+01302930      " R30=R30+R29 Zkraceni
N31           XR30  " Posledni triska
      G26   R05=+00340009      " R09=R34
N32           ZR09  " Navrat v Z
      G26   R05=+00340009      " R09=R34
N33    G00           XR09  " Navrat v X
N35    G70           " Konec
"
"
"

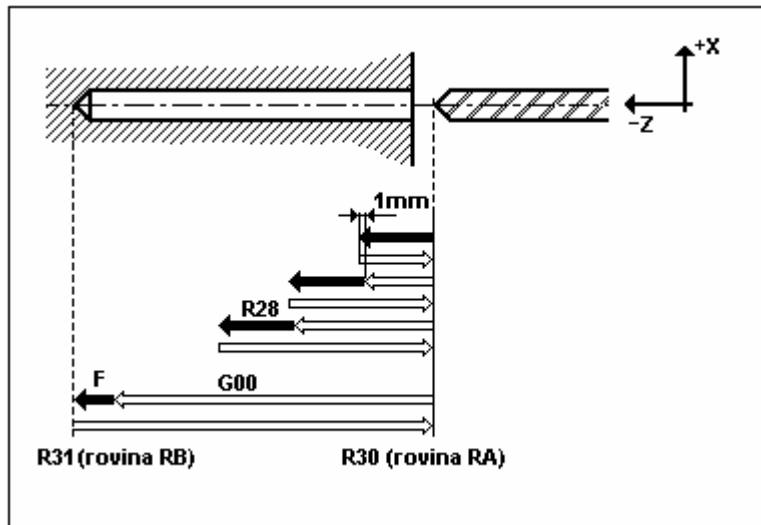
```

12.5.4 Vrtání hlubokých děr - G83

Použité parametry:

R26			Rychlost posuvu
R27			Velikost otáček (adresa S) v souladu s převodovým stupněm
R28			Přírůstek vrtání - q
R29			Nepoužit
R30	(+/-)		Deklaruje souřadnici roviny RA
R31	(+/-)		Deklaruje souřadnici roviny RB

Na obrázku je příklad cyklu G83:



Výpis pevného cyklu G83 ze souboru PEVNECY5.NCP

```
"
" VRTANI HLUBOKYCH DER S PROPLACHOVANIM - G83
"
"
N1  G79 L83
" VRTANI V OSE Z VE SMERU MINUS
"
N210 G26 R5=15313024      "IF R31 > R30 THEN R24=73 (VRTANI VE SMERU PLUS)
N220 GR24 L240
"
N40  G27 R5=00300029 R6=00300025 R24=1.000
N50  G26 R5=16312923      " IF R31 >= R29 then R23 je 73 else
78
N60  GR23 L140
N70  ZR25 G00 G26 R5=02292829      " R29 = R29 - R28
N80  G26 R5=15293123      " IF R31 > R29 then R23 je 73 else 78
N90  GR23 L110 Q9998      " Obskoci blok 10

N100 G26 R5=00310029      " R29 = R31
N110 ZR29 G01 FR26      " vrta pracovni rychlosti
N120 ZR30 G00      " rychloposuvem vyjede do roviny RA
N130 G26 R5=01292425 G73 L50 Q9998 " R25 = R29 + R24 (R24 = 1mm)
N140 G70
"
" VRTANI V OSE Z VE SMERU PLUS
"
N240 G27 R5=00300029 R6=00300025 R24=1.000 "R30 -> R29
N250 G26 R5=18312923      " IF R31 <= R29 then R23 je 73 else
78
N260 GR23 L340
N270 ZR25 G00 G26 R5=01292829      " R29 = R29 + R28
N280 G26 R5=17293123      " IF R31 < R29 then R23 je 73 else
78
N290 GR23 L310 Q9998      " Obskoci blok 300

N300 G26 R5=00310029      " R29 = R31
N310 ZR29 G01 FR26      " vrta pracovni rychlosti
N320 ZR30 G00      " rychloposuvem vyjede do roviny RA
```

N330 G26 R5=02292425 G73 L250 Q9998 " R25 = R29 - R24 (R24 = 1mm)
 N340 G70
 "
 "

12.5.5 Řezání závitu na válcové ploše - G84

Použité parametry:

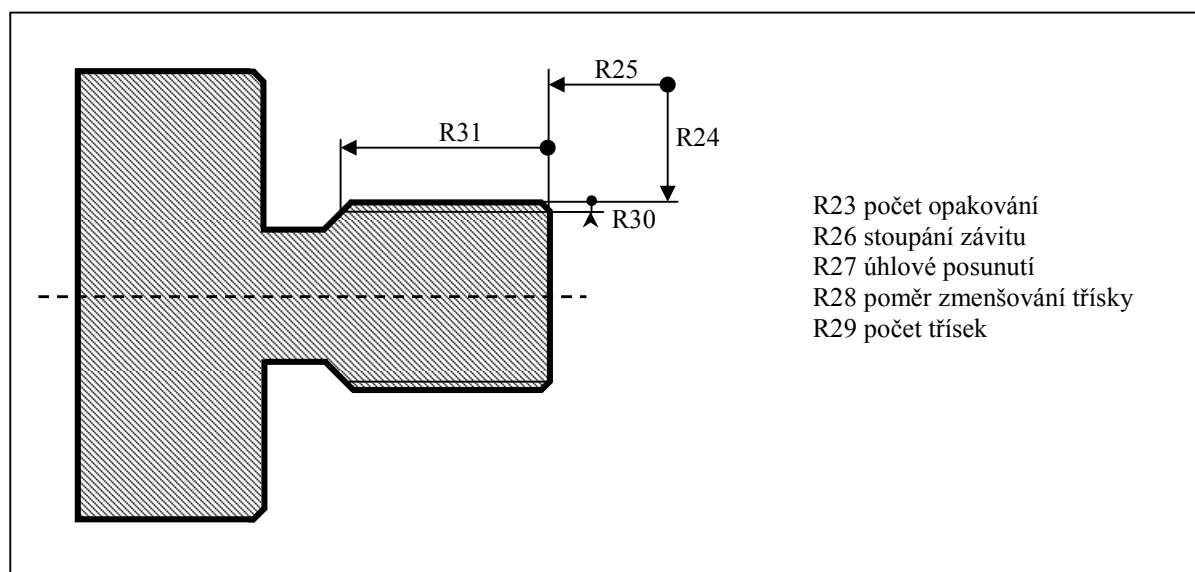
R19			Kolik materiálu nechat při střídavém hrubování závitu na stěně závitu (např. 0.05mm). Tento přírůstek se odebere při posledních n-průchodech dle parametru R21
R20			Má význam pouze pro střídavé závitování: Pro metrický závit se zadá R20=0 (nebo 60 stupňů). Pro Whitworthův závit se zadá R20=55 (stupňů) Pro lichoběžníkový závit se zadá R20=30 (stupňů) Jiné hodnoty než zde uvedené provedou rovněž metrický závit Pozn.: Zatím implementován pouze metrický závit !!!
R21			Způsob řezání závitu (0=v ose úhlu závitu, různé od nuly=střídavé, číslo udává kolik třísek před koncem skončit se střídáním)
R22			Pro závit bez výjezdu nepoužít
R23			Počet opakovaných přejezdů po konečné dráze
R24	(+/-)	(Ø)	Kolik mm před dílcem v ose X začíná PC - znaménko udává směr
R25	(+/-)		Kolik mm před čelem v ose Z začíná PC - znaménko udává směr
R26			Stoupání závitu
R27			Úhlové posunutí začátku závitování
R28			Kolikrát je následující tříska menší než předešlá (kvocient < 1), doporučená hodnota kvocientu 0.8
R29			Počet třísek, kterými má být závit vyříznut Pozn.: při střídavém závitování se jednou třískou myslí dva průchody na stejné hloubce závitu !!!
R30	(+/-)	(Ø)	Hloubka závitu v ose X - znaménko udává směr
R31	(+/-)		Délka závitu v ose Z - znaménko udává směr

Pozn.: Parametry R19 a R20 jsou nutné pouze pro střídavé závitování (R21 není 0).

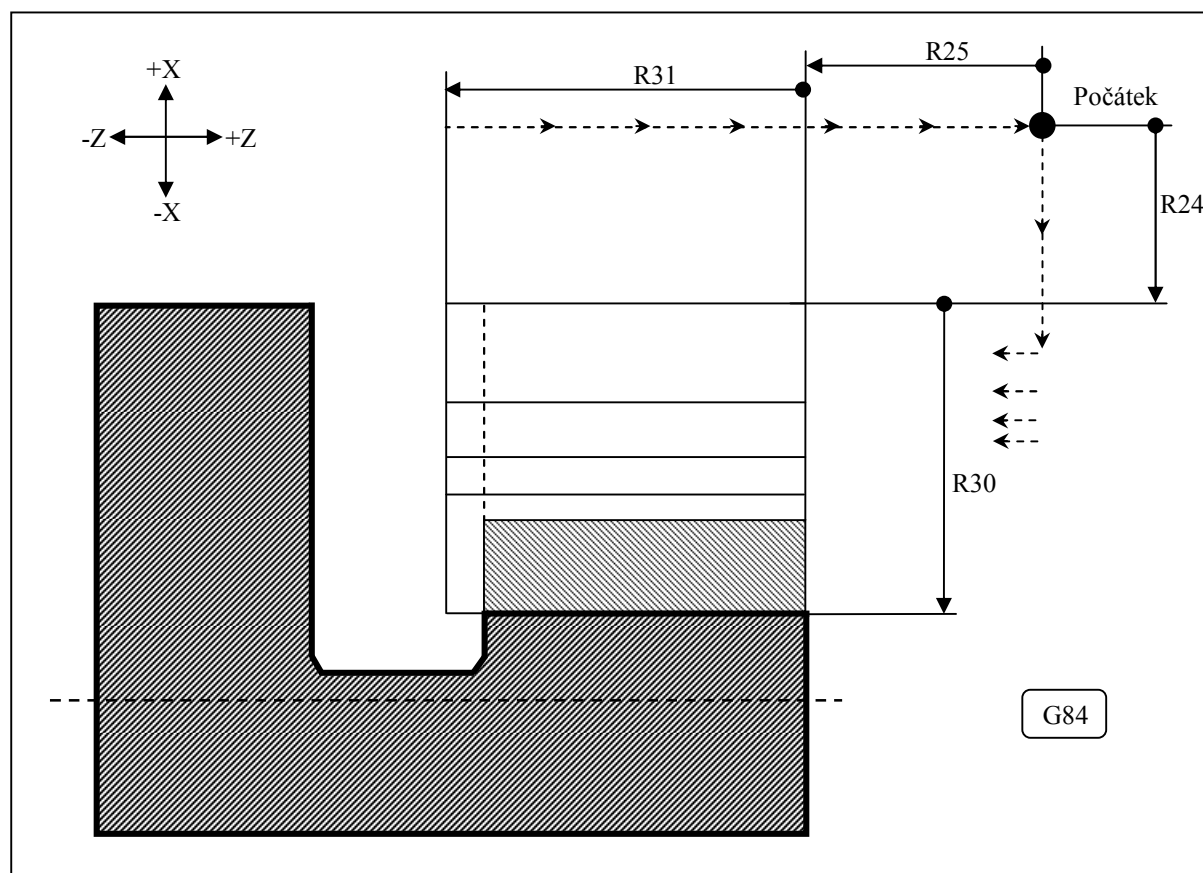
Parametr R28 (kvocient) udává, kolikrát je následující tříska menší než předešlá tříska v závitování. Třísky při závitování se postupně zmenšují podle této konstanty a tvoří geometrickou řadu s kvocientem menším než jedna. Například při síle třísky 0,5 mm a kvocientu 0,8 budou následující tloušťky:

	0,5 mm
0,50 . 0,8 =	0,4 mm
0,40 . 0,8 =	0,32 mm
0,32 . 0,8 =	0,256 mm

Závitování je možno použít ve všech čtyřech zrcadlových rovinách. Roviny závitování jsou určeny znaménky parametrů R30 a R31. Znaménka parametrů R24, R25 jsou také závislá na použité rovině závitování. Možno říci, že všechny tyto parametry (R24, R25, R30, R31) jsou programovány přírůstkově vzhledem k počátečnímu bodu, podobně jak je tomu u adres I a J při programování kružnice.



Na obrázku je příklad závitování, ve kterém se předpokládá, že parametry R24, R25, R30, R31 mají záporné hodnoty.



Řezání závitů na válcové ploše (průměrové programování)

```

" REZANI ZAVITU NA VALCOVE PLOSE - G84
"
N1  G79  L84
N2  R00=0 R22=0          " Pro zavít bez vyjezdu zajisti 0
      G28  R05=+19210002 " IF R21=0 THEN R02=73 else R02=78
      R06=00190047      " R19 -> R47 (prídavek potreba jen pro strídave)
      R07=00260048      " do R48 nenulovou hodnotu jako príznak strídani

N3  GR02 L5              " je-li rezani v ose, tak skok na N5
" -----
" REZANI STRIDAVE
" -----
N28  G90  G24  FR26      " Absolutne
      R00=+0            " R00=0
      R01=+1            " R01=1
      R02=-2            " R02=-2
      R03=+2.000        " Pomocna pro deleni 2

      G26  R05=+33060020 " Program 06-20
      R06=+00300044      " Uschova hloubku zavitu do R44
      R07=+23290039      " R39=real(R29)
      R08=+28283940      " R40=soucet geometricke rady, kvocient R28
      R09=+07300039      " R39=abs(R30)
      R10=+04394040      " R40=R39/R40 = 1. spona
      R11=+31300000      " IF R30<R00 THEN
      R12=+08400040      " R40=-R40
      R13=+22000000      " ENDIF
      R14=+26010283      " Naplni X prog. do R33 a Z prog. do R34
      R15=+01333333      " Pro prumerove progr: R33=R33+R33 (01333333) !
      R16=+00800085      " Uschova R35,36=R30,31
      R17=+01332411      " R11=R33+R24
      R18=+01113030      " R30=R11+R30 koncova mira
      R19=+01114011      " R11=R11+R40 (prvni triska)
      R20=+01342538      " R38=R34+R25
N30  G29  R05=+01383131  " R31=R38+R31
      R06=+01022910      " R10=R02+R29 (pocet opakovani)
      R07=+04260337      " R37=R26/R03: stoupani/2
      R08=+00400046      " R40 -> R46 trisku bude scitat do R46

N40  R18=34.000          " Konstanta pro vzorec metricky zavít.
      R20=69999.999      " Jen pro porovnani, aby poprve spocital posun
      G29  R05=+02102121 " R21=R10-R21, R21 je citac strídavych pruchodu
      R06=+01210121      " R21=R21+R01 zvetsi o 1
      R07=+01210121      " R21=R21+R01 zvetsi jeste o 1
      R08=+03182641      " (34 x S) do parametru R41 (pro vzorec)
"
" -----
"----- Prvni pruchod zavitu -----
" -----
" Vypocte posunu P v ose Z pro metricky zavít podle vorce:
" P = posun, S = stoupani, T = hloubka trisky
" Pozn.: 34 x S se nemeni a je predpocitano predem do R41
" P=((34 x S) - (55.426 x T))/96

N55  R17=96.000          " Konstanta pro vzorec metricky zavít.
      R18=34.000          " Konstanta pro vzorec metricky zavít.
      R19=55.426          " Konstanta pro vzorec metricky zavít.
      R32=-1              " Polomerove progr. R32= 0 !!!

```

```

" Prumerove progr. R32=-1 (binarni posun vpravo)

G29 R05=09463242 " Pri prumerovem progr. deli trisku 2 (bin. posun)
    R06=03194245 " (55.426 x T) do parametru R45
    R07=07450045 " Absolutni hodnota R45 (vzdy kladny)
    R08=02414543 " ((34 x S) - (55.426 x T)) do R43

N56 G29 R05=21002018 " Je R20 ruzne od 0? (posun neni 0 ?)
    R06=04431720 " ANO: ((34 x S) - (55.426 x T))/96 do R20
    R07=22000000 " Konec podminky

" V R20 je vypocteny posun, od posunu se odecte jeste pripadny pridavek
" Nebo je v R20 jiz 0.

N57 G28 R05=+02204703 " Odecte jeste hrubovaci pridavek a da do R03
    R06=+00030044 " R03 uschovano jeste do R44
    R07=+02340303 " od programovane hodnoty odecte posun

N60 G00 XR11 ZR03 " Jede v X a Z na miru
    G27 R05=+00330009 " R33 -> R09
    R06=+00230039 " R23 -> R39 pocet po stejne draze
" N61/ G73 L70 Q9998
" N62 M0
N70 G33 IR27 ZR31 " G33 zavitovani
N80 G00 XR09 " Vyjezd v X
    G26 R05=+00340009 " R34 -> R09, Pocatecni mira v Z
N90 G00 ZR09 " Navrat v Z
    G28 R05=+00440003 " Vybere uschovane R44 (posun) a da do R03
    R06=+01340303 " R03 = R34 + R03 ... prog. hodnota + posun
    R07=+19480008 " if R48 = 0 then G73 else G78 do R08

N110 GR08 L160 Q9998 " (je-li priznak = 0, uz nedela druhy pruchod)
"-----
" Druhy pruchod na stejne hloubce X, posunutý na druhou stranu zavitu
"-----

N120 G00 XR11 ZR03 " Jede v X a Z na miru
N130 G33 IR27 ZR31 " G33 - zavitovani
    G26 R05=00330009 " R09=R33
N150 G00 XR09 " Vyjede v ose X
N160 G26 R05=+33120017 " PROGRAM OD R12 DO R17
    R12=+03402840 " R40=R40xR28 - nasobeni kvocientem
    R13=+01404646 " triska se nascitava do R46
    R14=+01114011 " R11=R11+R40 (dalsi triska)
    R15=+00340009 " R09=R34
    R16=+02210121 " R21=R21-R01, zmensi citac pruchodu o 1
    R17=+15210008 " if R21 > 0 then G73 else G78 do R08
N180 GR08 L200 Q9998 " Citac > 0 tak skok na N200
N185 G26 R05=19480008 " priznak stridani v R48 je jiz 0 ?
N186 GR08 L200 Q9998 " ANO, tak skok na N200

N190 G28 R05=+00000048 " Vynuluje priznak stridani v R48
    R06=+00000020 " Vynuluje R20 (vypocteny posun)
    R07=+00000047 " Vynuluje R47 (pridavek hrubovaci)
N200 G00 ZR09 " Navrat v Z
    G73 L55 QR10 " Podmineny skok na blok N55
N210 G00 FR26 XR30 " Jede v X na posledni miru
    G27 R05=+00330009 " R09=R33
    R06=+02390139 " R39=R39-R01
N220 G33 IR27 ZR31 " G33 Zavitovani na miru
N230 G00 XR09 " Navrat v X
    G27 R05=+00340009 " R09=R34

```

```

        R06=+16390010      " IF R39 >= R00 THEN R10=73 ELSE R10=78
N240 G00 ZR09              " Navrat v Z
        GR10 L210 Q9998    " Podmineny skok
N250 G70                  " Konec
"-----
" REZANI V OSE
"-----
N5  G90  G24  FR26          " Absolutne
        R00=+0              " R00=0
        R01=+1              " R01=1
        R02=-2              " R02=-2

        G26  R05=+33060021  " Program 06-21
        R06=+23290039      " R39=real(R29)
        R07=+28283940      " R40=soucet geometricke rady, kvocient R28
        R08=+07300039      " R39=abs(R30)
        R09=+04394040      " R40=R39/R40 = 1. spona
        R10=+31300000      " IF R30<R00 THEN
        R11=+08400040      " R40=-R40
        R12=+22000000      " ENDIF
        R13=+26010283      " Naplni X prog. do R33 a Z prog. do R34
        R14=+01333333      " Pro prumerove progr: R33=R33+R33 (01333333)
!
        R15=+00800085      " Uschova R35,36=R30,31
        R16=+01332411      " R11=R33+R24
        R17=+01113030      " R30=R11+R30 koncova mira
        R18=+01114011      " R11=R11+R40 (prvni triska)
        R19=+01342538      " R38=R34+R25
        R20=+01383131      " R31=R38+R31
        R21=+01022910      " R10=R02+R29 (pocet opakovani)

N50 G00 XR11              " Jede v X na miru
        G27  R05=+00330009  " R09=R33
        R06=+00230039      " R39=R23 pocet po stejne draze
"N51/G73 L6 Q9998
"N52 M0
N6  G33  IR27              ZR31  " G33 Zavitovani
N7  G00              XR09      " Vyjezd v X
        G28  R05=+03402840  " R40=R40.R28 - nasobeni kvocientem
        R06=+01114011      " R11=R11+R40 (dalsi triska)
        R07=+00340009      " R09=R34
N8  G00              ZR09      " Navrat v Z
        G73  L50 QR10        " Podmineny skok na blok N50
N9  G00  FR26              XR30  " Jede v X na posledni miru
        G27  R05=+00330009  " R09=R33
        R06=+02390139      " R39=R39-R01
N10 G33  IR27              ZR31  " G33 Zavitovani na miru
N11 G00              XR09      " Navrat v X
        G27  R05=+00340009  " R09=R34
        R06=+16390010      " IF R39>=R00 THEN R10=73 ELSE R10=78
N12 G00              ZR09      " Navrat v Z
        GR10 L9  Q9998      " Podmineny skok
N13 G70                  " Konec

```

12.5.6 Řezání závitu na válcové ploše s výjezdem - G85

Použité parametry:

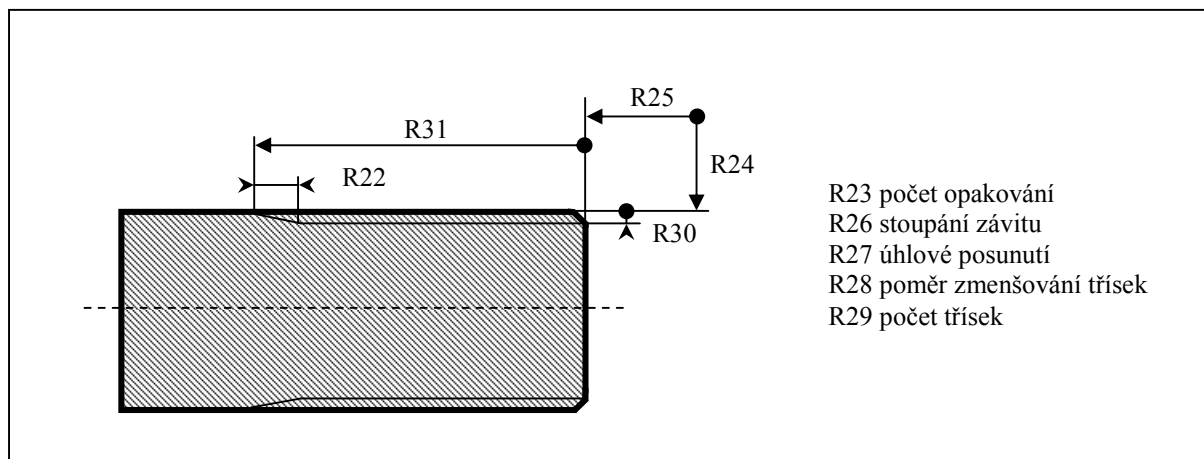
R04			Úhel výjezdu ze závitu – popis parametru viz kapitola závitování !!!
R19			Kolik materiálu nechat při střídavém hrubování závitu na stěně závitu (např. 0.05mm). Tento přírůstek se odebere při posledních n-průchodech dle parametru R21
R20			Má význam pouze pro střídavé závitování: Pro metrický závit se zadá R20=0 (nebo 60 stupňů). Pro Whitworthův závit se zadá R20=55 (stupňů) Pro lichoběžníkový závit se zadá R20=30 (stupňů) Jiné hodnoty než zde uvedené provedou rovněž metrický závit Pozn.: Zatím implementován pouze metrický závit !!!
R21			Způsob řezání závitu (0=v ose úhlu závitu, různé od nuly=střídavé, číslo udává kolik třísek před koncem skončit se střídáním)
R22			Délka výjezdu
R23			Počet opakovaných přejezdů po konečné dráze
R24	(+/-)	(Ø)	Kolik mm před dílcem v ose X začíná PC - znaménko udává směr
R25	(+/-)		Kolik mm před čelem v ose Z začíná PC - znaménko udává směr
R26			Stoupání závitu
R27			Úhlové posunutí začátku závitování
R28			Kolikrát je následující tříska menší než předešlá (kvocient < 1), doporučená hodnota kvocientu 0.8
R29			Počet třísek, kterými má být závit vyříznut Pozn.: při střídavém závitování se jednou třískou myslí dva průchody na stejné hloubce závitu !!!
R30	(+/-)	(Ø)	Hloubka závitu v ose X - znaménko udává směr
R31	(+/-)		Délka závitu v ose Z - znaménko udává směr

Parametr R28 – viz popis u G84

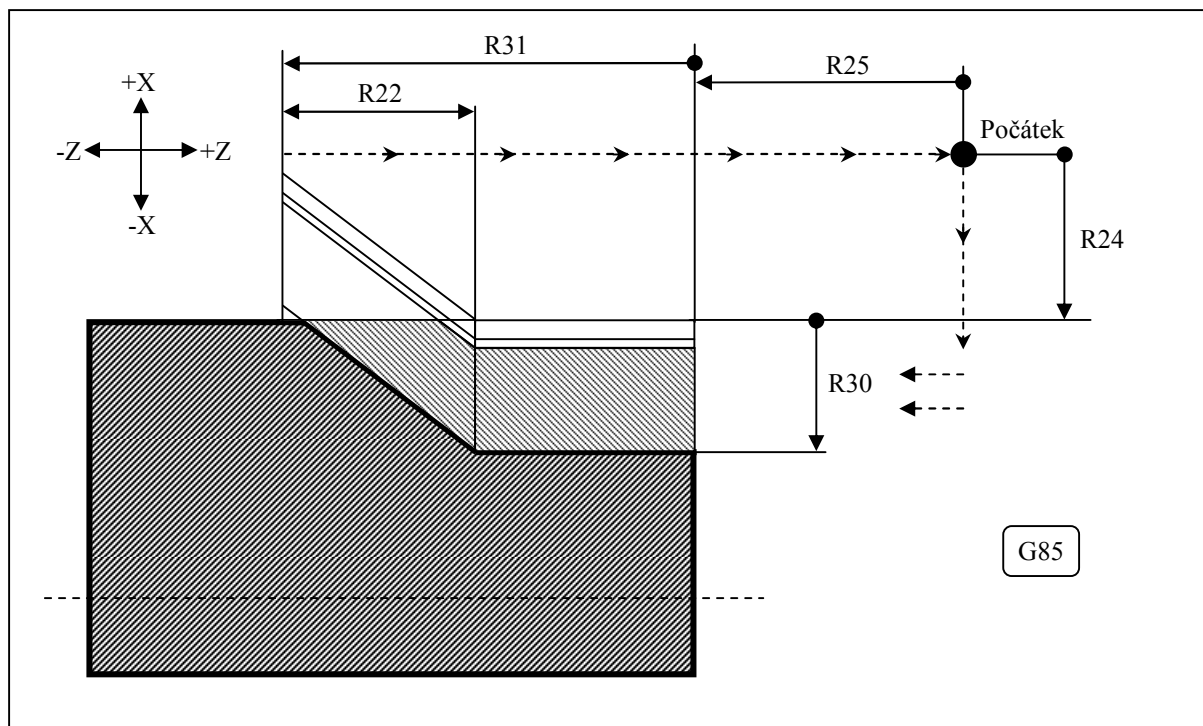
Parametr R22 udává délku výjezdu (kladné číslo). Celková délka závitování je dána parametrem R31 včetně délky výjezdu.

Úhel výjezdu je dán buď strojní konstantou číslo 8 (soubor TAB0.REK) nebo parametrem, jehož číslo je v této strojní konstantě zadáno. Podrobnosti viz kapitola ZÁVITOVÁNÍ. Délka výjezdu musí být určena tak, aby při posledním závitovacím chodu vyjel nůž nad materiál.

Závitování je možno použít ve všech čtyřech zrcadlových rovinách. Roviny závitování jsou určeny znaménky parametrů R30 a R31. Znaménka parametrů R24, R25 jsou také závislá na použité rovině závitování. Možno říci, že všechny tyto parametry (R24, R25, R30, R31) jsou programovány přírůstkově vzhledem k počátečnímu bodu, podobně jak je tomu u adres I a J při programování kružnice.



Na obrázku je příklad závitování s výjezdem, ve kterém se předpokládá, že parametry R24, R25, R30, R31 mají záporné hodnoty.



Řezání závitu na válcové ploše s výjezdem

```
" REZANI ZAVITU NA VALCOVE PLOSE S VYJEZDEM - G85
"
N1  G79  L85
N2  G28  R05=+19210002  " IF R21=0 THEN R02=73 else R02=78
      R06=00190047      " R19 -> R47 (pridavek potreba jen pro stridave)
      R07=00260048      " do R48 nenulovou hodnotu jako priznak stridani
      R00=0
N3  GR02 L5              " je-li rezani v ose, tak skok na N5
" -----
" REZANI STRIDAVE
" -----
N28  G90  G24  FR26      " Absolutne
      R00=+0              " R00=0
      R01=+1              " R01=1
      R02=-2              " R02=-2
      R03=+2.000          " Pomocna pro deleni 2

      G26  R05=+33060020  " Program 06-20
      R06=+00300044      " Uschova hloubku zavitu do R44
      R07=+23290039      " R39=real(R29)
      R08=+28283940      " R40=soucet geometricke rady, kvocient R28
      R09=+07300039      " R39=abs(R30)
      R10=+04394040      " R40=R39/R40 = 1. spona
      R11=+31300000      " IF R30<R00 THEN
      R12=+08400040      " R40=-R40
      R13=+22000000      " ENDIF
      R14=+26010283      " Naplni X prog. do R33 a Z prog. do R34
      R15=+01333333      " Pro prumerove progr: R33=R33+R33 (01333333) !
```

```

        R16=+00800085    " Uschova R35,36=R30,31
        R17=+01332411    " R11=R33+R24
        R18=+01113030    " R30=R11+R30 koncova mira
        R19=+01114011    " R11=R11+R40 (prvni triska)
        R20=+01342538    " R38=R34+R25
N30   G29   R05=+01383131    " R31=R38+R31
        R06=+01022910    " R10=R02+R29 (pocet opakovani)
        R07=+04260337    " R37=R26/R03: stoupání/2
        R08=+00400046    " R40 -> R46 trisku bude scitat do R46

N40   R18=34.000        " Konstanta pro vzorec metricky zavít.
        R20=69999.999    " Jen pro porovnání, aby poprve spocital posun
        G29   R05=+02102121    " R21=R10-R21, R21 je citac stridavych pruchodu
        R06=+01210121    " R21=R21+R01 zvetsi o 1
        R07=+01210121    " R21=R21+R01 zvetsi jeste o 1
        R08=+03182641    " (34 x S) do parametru R41 (pro vzorec)
"
"-----
"----- Prvni pruchod zavitu -----
"-----
" Vypocte posunu P v ose Z pro metricky zavít podle vorce:
" P = posun, S = stoupání, T = hloubka trisky
" Pozn.: 34 x S se nemeni a je predpocitano predem do R41
" P=((34 x S) - (55.426 x T))/96

N55   R17=96.000        " Konstanta pro vzorec metricky zavít.
        R18=34.000        " Konstanta pro vzorec metricky zavít.
        R19=55.426        " Konstanta pro vzorec metricky zavít.
        R32=-1           " Polomerove progr. R32= 0 !!!
                        " Prumerove progr. R32=-1 (binarni posun vpravo)

        G29   R05=09463242    " Pri prumerovem progr. deli trisku 2 (bin. posun)
        R06=03194245    " (55.426 x T) do parametru R45
        R07=07450045    " Absolutni hodnota R45 (vzdy kladny)
        R08=02414543    " ((34 x S) - (55.426 x T)) do R43

N56   G29   R05=21002018    " Je R20 ruzne od 0? (posun neni 0 ?)
        R06=04431720    " ANO: ((34 x S) - (55.426 x T))/96 do R20
        R07=22000000    " Konec podminky

" V R20 je vypocteny posun, od posunu se odecte jeste pripadny pridavek
" Nebo je v R20 jiz 0.

N57   G28   R05=+02204703    " Odecte jeste hrubovaci pridavek a da do R03
        R06=+00030044    " R03 uschovano jeste do R44
        R07=+02340303    " od programovane hodnoty odecte posun

N60   G00   G98 XR11  ZR03    " Jede v X a Z na miru
        G27   R05=+00330009    " R33 -> R09
        R06=+00230039    " R23 -> R39 pocet po stejne draze
"N61/G73 L70 Q9998
"N62 M0
N70   G33   IR27 JR22 ZR31    " G33 zavítovani
N75                                " Prazdny blok musi byt
N80   G00   G24 XR09 ZR31    " Vyjezd v X
        G26   R05=+00340009    " R34 -> R09, Pocatecni mira v Z
N90   G00   ZR09            " Navrat v Z
        G28   R05=+00440003    " Vybere uschovane R44 (posun) a da do R03
        R06=+01340303    " R03 = R34 + R03 ... prog. hodnota + posun

```

```

R07=+19480008      " if R48 = 0 then G73 else G78 do R08

N110 GR08 L160 Q9998      " (je-li priznak = 0, uz nedela druhy pruchod)
"-----
" Druhy pruchod na stejne hloubce X, posunutý na druhou stranu zavitu
"-----
N120 G00 G98 XR11 ZR03      " Jede v X a Z na miru
N130 G33 IR27 JR22 ZR31      " G33 - zavitovani
      G26 R05=00330009      " R09=R33
N135                        " Prazdny blok musi byt
N150 G00 G24 XR09 ZR31      " Vyjede v ose X
N160 G26 R05=+33120017      " PROGRAM OD R12 DO R17
      R12=+03402840      " R40=R40xR28 - nasobeni kvocientem
      R13=+01404646      " triska se nascitava do R46
      R14=+01114011      " R11=R11+R40 (dalsi triska)
      R15=+00340009      " R09=R34
      R16=+02210121      " R21=R21-R01, zmensi citac pruchodu o 1
      R17=+15210008      " if R21 > 0 then G73 else G78 do R08
N180 GR08 L200 Q9998      " Citac > 0 tak skok na N200
N185 G26 R05=19480008      " priznak stridani v R48 je jiz 0 ?
N186 GR08 L200 Q9998      " ANO, tak skok na N200

N190 G28 R05=+00000048      " Vynuluje priznak stridani v R48
      R06=+00000020      " Vynuluje R20 (vypocteny posun)
      R07=+00000047      " Vynuluje R47 (pridavek hrubovaci)
N200 G00 ZR09      " Navrat v Z
      G73 L55 QR10      " Podmineny skok na blok N55
N210 G00 G98 FR26 XR30      " Jede v X na posledni miru
      G27 R05=+00330009      " R09=R33
      R06=+02390139      " R39=R39-R01
N220 G33 IR27 JR22 ZR31      " G33 Zavitovani na miru
N225                        " Prazdny blok musi byt
N230 G00 G24 XR09 ZR31      " Navrat v X
      G27 R05=+00340009      " R09=R34
      R06=+16390010      " IF R39 >= R00 THEN R10=73 ELSE R10=78
N240 G00 ZR09      " Navrat v Z
      GR10 L210 Q9998      " Podmineny skok
N250 G70      " Konec
"-----
" REZANI V OSE
"-----
N5 G90 G24 FR26      " Absolutne
      R00=+0      " R00=0
      R01=+1      " R01=1
      R02=-2      " R02=-2

      G26 R05=+33060021      " Program 06-21
      R06=+23290039      " R39=real(R29)
      R07=+28283940      " R40=soucet geometricke rady, kvocient R28
      R08=+07300039      " R39=abs(R30)
      R09=+04394040      " R40=R39/R40 = 1. spona
      R10=+31300000      " IF R30<R00 THEN
      R11=+08400040      " R40=-R40
      R12=+22000000      " ENDIF
      R13=+26010283      " Naplni X prog. do R33 a Z prog. do R34
      R14=+01333333      " Pro prumerove progr: R33=R33+R33 (01333333)
!
      R15=+00800085      " Uschova R35,36=R30,31
      R16=+01332411      " R11=R33+R24
      R17=+01113030      " R30=R11+R30 koncova mira
      R18=+01114011      " R11=R11+R40 (prvni triska)

```

```

R19=+01342538      " R38=R34+R25
R20=+01383131      " R31=R38+R31
R21=+01022910      " R10=R02+R29 (pocet opakovani)

N50 G00 G98 XR11      " Jede v X na miru
      G27 R05=+00330009 " R09=R33
      R06=+00230039    " R39=R23 pocet po stejne draze
"N51/G73 L6 Q9998
"N52 M0
N6 G33 IR27 JR22      ZR31 " G33 Zavitovani
N65 " Prazdny blok musi byt
N7 G00 G24 XR09 ZR31 " Vyjezd v X
      G28 R05=+03402840 " R40=R40.R28 - nasobeni kvocientem
      R06=+01114011    " R11=R11+R40 (dalsi triska)
      R07=+00340009    " R09=R34
N8 G00 ZR09 " Navrat v Z
      G73 L50 QR10      " Podmineny skok na blok N50
N9 G00 G98 FR26 XR30 " Jede v X na posledni miru
      G27 R05=+00330009 " R09=R33
      R06=+02390139    " R39=R39-R01
N10 G33 IR27 JR22 ZR31 " G33 Zavitovani na miru
N14 " Prazdny blok musi byt
N11 G00 XR09 ZR31 " Navrat v X
      G27 R05=+00340009 " R09=R34
      R06=+16390010    " IF R39>=R00 THEN R10=73 ELSE R10=78
N12 G00 G24 ZR09 " Navrat v Z
      GR10 L9 Q9998    " Podmineny skok
N13 G70 " Konec

```

12.5.7 Srážení hrany pod úhlem - G86

```

"-----
" HRANA POD UHLEM
"
" Pouzite parametry:"
" R24 = Kolik mm pred dilcem v ose X zacina PC - znam. udava smer (Xp)"
" R25 = Kolik mm pred celem v ose Z zacina PC - znam. udava smer (Zp)"
" R26 = Rychlost posuvu"
" R27 = nepouzito"
" R28 = Sila (tloustka) trisky v mm (kolmo k ose Z) (Tx)"
" R29 = nepouzito pracovne pro (Tz)"
" R30 = Delka srazene hrany v ose X (znamenko udava smer pohybu) (Dx)"
" R31 = Delka srazene hrany v ose Z (znamenko udava smer pohybu) (Dz)"
""
" Znamenko je urcujici pouze podle parametru R24 a R25 !!!"

N1 G79 L86
""
" Znamenko si uchova do parametru R16 pro X a R17 pro Z"
""
N12 G26 R05=00280012 " Uschova tloustku trisky v X do R12"
      R10=1 " Az bude R10=0, tak bude konec"
      R16=0 R17=0
      G24 " urychleni komunikace"
N30 G26 R05=16241615 " if R24<R16(=0) then R16=-1 (R15=73 nebo 78)"
N40 GR15 L60 R16=1.0
N50 R16=-1.0
N60 G26 R05=16251715 " if R25<R17(=0) then R17=-1 (R15=73 nebo 78)"

```

```
N70 GR15 L100 R17=1.0
N80 R17=-1.0
""
" Prevede vsechny miry pro vypocty na kladne hodnoty"
""
  N100 G29 R05=07240024
        R06=07250025
        R07=07300030
        R08=07310031
N110 G26 R05=07280028
" "
" Predpocte tloustku trisky v ose:"
" Tz = (Tx . Dz)/Dx ... R29=(R28.R31)/R30"
N120 G29 R05=03283127      "R27=R28.R31  "
        R06=04273029      "R29=Tz(tloustka v ose Z)  "

" Predpocte a = (Zp . Dx)/Dz .... R20 = (R25 . R30)/R31"
" a je v R20"
        R07=03253020      "R20=R25.R30"
        R08=04203120      "R20=R20/R31"

" Predpocte b = (Xp . Dz)/Dx .... R21 = (R24 . R31)/R30"
" b je v R21"
N160 G29 R05=03243121      "R21=R24.R31"
        R06=04213021      "R21=R21/R30"

" Predpocte A = a + Xp      R22 = R20 + R24"
        R07=01202422

" Predpocte B = b + Zp      R23 = R21 + R25"
        R08=01212523

" Vypocet pojezdu v X a Z (absloutne)"
" X =  A + Tx      R18 = R22 + R28"
" Z =  B + Tz      R19 = R23 + R29"

N200 G29 R05=01222818
        R06=01232919
""
" Prvni prirustkove miry s ohledem na znamenko "
" Ziskam je vynasobenim znamenkem uschovany v R16 (X) a R17 (Z)"
        R07=03181618
        R08=03191719
        R15=-1.0      "PRO NASOBENI -1"
        R09=2.0      "Pro nasobeni pri prumer. prog. 2x"

" Pojezdy prirustkove: "
""
N240 G91 G00 XR18 G26      R05=03181518  "predem obrati znam. X.-1"
N250 G01 G26 XR18 ZR19 FR26 R05=03191519  "predem obrati znam. Z.-1"
N260 G00      ZR19
        R0=0
" A jsme zase ve vychozim bode  "
" Pricte trisku Tx = Tx + Tx ... R28 = R28 + R12"
" "
" Testuje R10"
" "
N264 G26 R5=19001001
N266 GR1 L1000      " SKOK NA KONEC"
N270 G26 R05=01281228
" Je triska Tx R28 vetsi nez delka srazene hrany Dx "
```

```

N280 G26 R05=15283011      " if R28 > R30 then R11=73 else R11=78"
N290 GR11 L310
N300 G73 L120 Q9998        " Bezny prubeh"
N310 G26 R05=00300028      " Presune Dx do Tx t.j. R30 do R28 "
N315 R10=0                 " Priznak posledni pruchod      "
N320 G73 L120 Q9998
N1000 G70
"

```

12.5.8 Řezání závitu na kuželové ploše - G87

```

"
" -----
" REZANI ZAVITU NA KUZELOVE PLOSE - G87
"
" Pouzite parametry:
" R22 = Stoupani kuzelu
" R23 = Pocet opakovanych prejezdu po konecne draze
" R24 = Kolik mm pred dilcem v ose X zacina PC - znamenko udava smer
" R25 = Kolik mm pred celem v ose Z zacina PC - znamenko udava smer
" R26 = Stoupani zavitu
" R27 = Uhlove posunuti zacatku zavitovani
" R28 = Kolikrat je nasledujici triska mensi nez predesla (kvocient < 1)
" R29 = Pocet trisek, kterymi ma byt zavit vyriznut
" R30 = Hloubka zavitu v ose X - znamenko udava smer
" R31 = Delka zavitu v ose Z - znamenko udava smer
"
"
N1  G79  L87
N2  G90  G24  FR26      " Absolutne
                        R00=+0      " R00=0
                        R01=+1      " R01=1
                        R02=-2      " R02=-2

      G26  R05=+33060021      " Program 06-21

                        R06=+23290039      " R39=real(R29)
                        R07=+28283940      " R40=soucet geometricke rady, kvocient R28
                        R08=+07300039      " R39=abs(R30)
                        R09=+04394040      " R40=R39/R40 = 1. spona
                        R10=+31300000      " IF R30<R00 THEN
                        R11=+08400040      "     R40=-R40
                        R12=+22000000      " ENDIF
                        R13=+26010283      " Naplni X prog. do R33 a Z prog. do R34
                        R14=+01333333      " Pro prumerove progr: R33=R33+R33 (01333333)
!
                        R15=+01332411      " R11=R33+R24
                        R16=+01113030      " R30=R11+R30 koncova mira
                        R17=+01114011      " R11=R11+R40 (prvni triska)
                        R18=+01342538      " R38=R34+R25
                        R19=+01383131      " R31=R38+R31
                        R20=+01022910      " R10=R02+R29 (pocet opakovani)
                        R21=+01112212      " R12=R11+R22

N50 G00      XR11      " Jede v X na miru
      G27  R05=+00330009      " R09=R33
      R06=+00230039      " R39=R23 pocet po stejne draze
N6  G33  IR27  XR12  ZR31      " Zavitovani
N7  G00      XR09      " Vyjezd v X
      G29  R05=+03402840      " R40=R40.R28 - nasobeni kvocientem

```

```

        R06=+01114011      " R11=R11+R40 (dalsi triska)
        R07=+00340009      " R09=R34
        R08=+01112212      " R12=R11+R22
N8   G00                    ZR09  " Navrat v Z
      G73  L50  QR10          " Podmineny skok na blok N50
N9   G00  FR26                XR30  " Jede v X na posledni miru
      G28  R05=+00330009      " R09=R33
        R06=+02390139      " R39=R39-R01
        R07=+01302212      " R12=R30+R22
N10  G33  IR27  XR12  ZR31    " Zavitovani na miru
N11  G00                    XR09    " Navrat v X
      G27  R05=+00340009      " R09=R34
        R06=+15390010      " IF R39>R00 THEN R10=73 ELSE R10=78
N12  G00                    ZR09    " Navrat v Z
      GR10 L9   Q9998        " Podmineny skok
N13  G70                    " Konec
"
```

13

13. PŘEPOČET SOUŘADNIC

Pozn: Přepočet souřadnic je implementován od softwarové verze 30.35 a 40.13. (26.3.2002).

13.1 Společné zásady

Pro řazení různých přepočtů souřadnic slouží 11. skupina G funkcí. Za pomoci přepočtů souřadnic je možno část programu různě natočit, zmenšit nebo posunout vůči originálu.

Každý přepočet je účinný ihned po naprogramování a platí do odvolání nebo do jeho vlastní změny. Všechny přepočty jsou vztaženy k aktuálnímu posunutí počátku G53-G59 a délkové a poloměrové korekce nejsou přepočtem ovlivněny.

V každém bloku je možno naprogramovat jen jednu z funkcí 11. skupiny G, proto když je potřeba použít kombinaci různých druhů přepočtů, je nutno je naprogramovat do několika různých bloků. Všechny funkce z 11. skupiny G mají kontinuální účinek a navzájem se neovlivňují, proto je možno kombinovat.

Každá změna přepočtů souřadnic by se měla provádět v samostatném bloku programu a při vyřazené poloměrové korekci. Poloměrovou korekci je možno použít pro aktivní přepočet.

Každá z funkcí složí na nastavování a také na vyřazení vlastního přepočtu. Jenom funkce G34 ruší všechny dostupné přepočty najednou, proto je vhodné funkci G34 zařadit do prioritního bloku.

Všechny přepočty souřadnic se automaticky ruší centrální anulací systému (bez ohledu na to, zda funkce G34 je zařazena v prioritním bloku, nebo ne).

Přepočty souřadnic popsané v této kapitole není vhodné použít pro obecnou transformaci souřadného systému, protože přepočty probíhají na úrovni přípravy bloku a přepočet se nepromítne například do ručních posuvů apod. Pro obecnou transformaci budou v systému určeny jiné prostředky.

Přehled funkcí 11. skupiny G (stav pro verzi 40.13) :

G34	Základní stav, všechny přepočty souřadnic jsou vyřazeny
G35	Zrcadlení
G36	Změna měřítka
G37	Nezávislý posun

13.2 Zrcadlení

Zrcadlením se získá tvar zrcadlového obrazu a programuje se pomocí funkce **G35**. Pomocí zrcadlení se převrací směr v ose nebo v několika osách. Pro zrcadlicí osu systém mění znaménko, podle interpolační roviny případně mění G2 za G3 a znaménko funkcí I a J, a podle korekční roviny mění případně G41 za G42.

Zrcadlení se provede vzhledem k platnému nulovému bodu a k aktuálnímu posunutí počátku. Zrcadlením také není ovlivněna délková a poloměrová korekce nástroje (ani posunutí hlav u dvouhlavého soustruhu).

Pokud se neprovedou další opatření, zrcadlení jedné osy změní sousledné obrábění za nesousledné a obráceně.

Systém umožňuje zadat zrcadlení podle libovolných rovin souřadného systému tím, že se zadávají osy, které mají převádět směr.

Programování:

V samostatném nepohybovém bloku se naprogramuje **G35** a zadají se osy, které mají převádět směr. Osy se programují s libovolnou hodnotou, například 0 (X0, Y0, Z0, U0, V0, W0, A0, B0, C0).

Když je v samostatném bloku programována samotná funkce **G35** bez os, zrcadlení se zruší.

Platné zrcadlení platí do změny pomocí funkce G35 nebo naprogramováním G34 (zrušení všech druhů přepočtů souřadnic) nebo do centrální anulaci systému.

Příklady programování zrcadlení:

G35	X0	"Převádí směr X	(zrcadlení podle osy Y)
G35	Y0	"Převádí směr Y	(zrcadlení podle osy X)
G35	X0 Y0	"Převádí směr X a Y	(zrcadlení podle středu X,Y)
G35	Y0 U0 C0	"Převádí směr Y,U a C	(zrcadlení podle středu Y,U a C)
G35		"Odvolání zrcadlení	

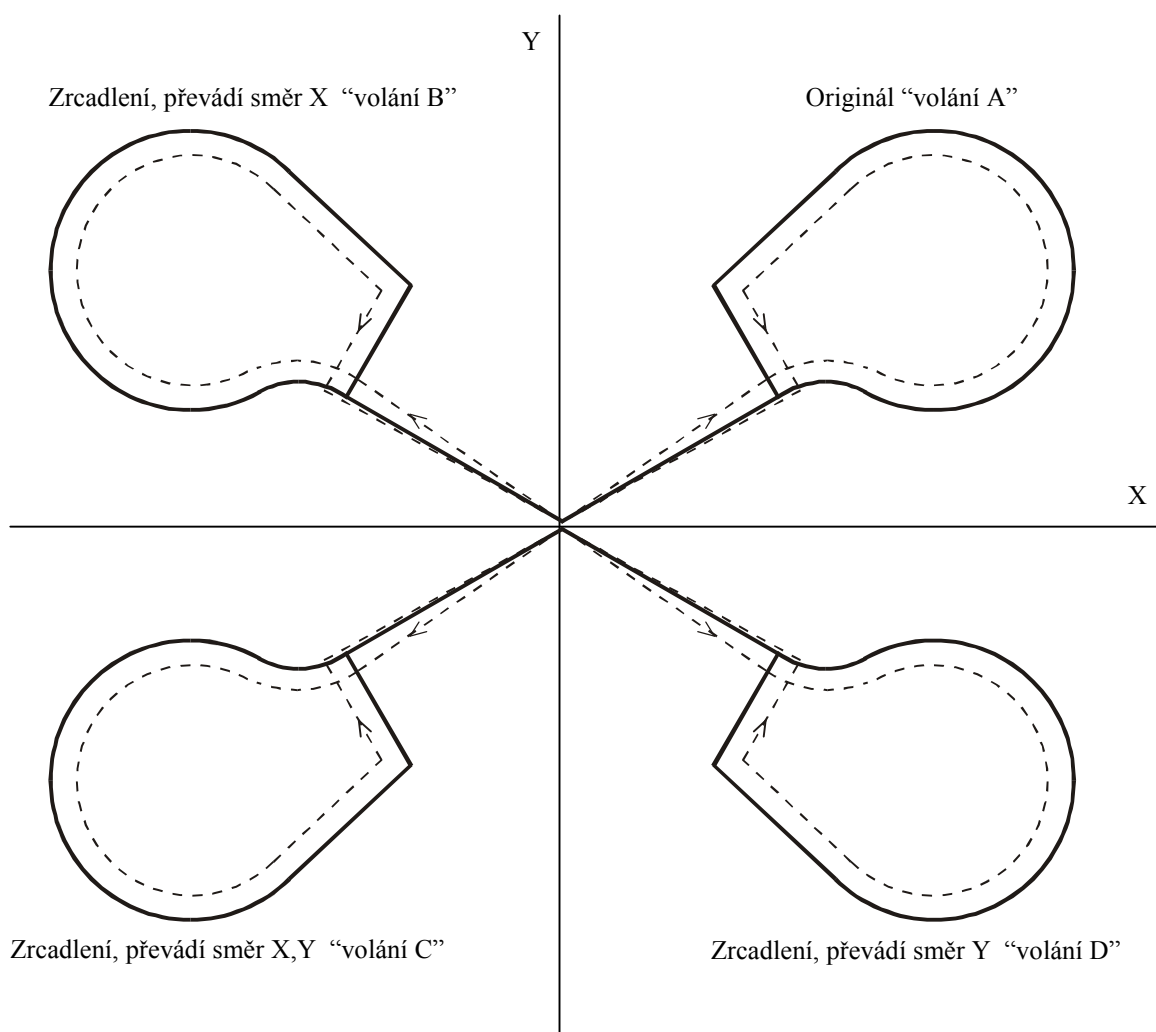
Příklad pro zrcadlení:

"V příkladu se opakuje 4x volání stejného podprogramu s různě zadaným "zrcadlením:

```
%1
N10 X0 Y0 F5000 G23
N20 G71 L100      "Volání podprogramu - A
N30 G35 X0        "Zrcadlení (převádí směr X)
N40 G71 L100      "Volání podprogramu - B
N50 G35 X0 Y0     "Zrcadlení (převádí směr X, Y)
N60 G71 L100      "Volání podprogramu - C
N70 G35 Y0        "Zrcadlení (převádí směr Y)
N80 G71 L100      "Volání podprogramu - D
N90 G35           "Zrušení zrcadlení
N99 M30
```

"Podprogram s poloměrovou korekcí

```
N1000 G79 L100      "Podprogram L100
N1010 G01 X0 Y0
N1020 G01 X30 Y20 G41 D3
N1030 G02 X50 Y24.849 I16.641 J-24.962
N1040 G03 X30 Y73.484 I3.359 J29.811
N1050 G01 X20 Y61.075
N1060 G01 X30 Y20
N1070 G01 X0 Y0 G40
N1080 G70
```



13.3 Změna měřítka

Změnou měřítka se získá geometricky zvětšený nebo zmenšený tvar a programuje se pomocí funkce **G36**. Změnou měřítka je možné zhotovit od jednoho originálu zvětšený nebo zmenšený obraz.

Změna měřítka se provede vzhledem k platnému nulovému bodu a k aktuálnímu posunutí počátku. Změnou měřítka také není ovlivněna délková a poloměrová korekce nástroje (ani posunutí hlav u dvouhlavého soustruhu).

Měřítka pro zvětšení a zmenšení se může zadat buď přímo hodnotou jak je tomu u zadávání souřadnic s přesností na 3 desetinná čísla nebo pomocí zlomku, kde číselník i jmenovatel jsou hodnoty s přesností na 3 desetinná čísla.

Programování:

V samostatném nepohybovém bloku se naprogramuje **G36** a zadají se osy, které určují poměr zvětšení nebo zmenšení.

- Pokud se naprogramuje jen první souřadnice (X), je měřítko určeno přímo její hodnotou.
- Pokud se naprogramuje první souřadnice (X) a současně druhá souřadnice (Y), je měřítko určeno hodnotou zlomku, kde v číselníku je hodnota první souřadnice a ve jmenovateli je hodnota druhé souřadnice.

Když je v samostatném bloku programována samotná funkce **G36** bez os, změna měřítka se zruší.

Platná změna měřítka platí do změny pomocí funkce G36 nebo naprogramováním G34 (zrušení všech druhů přepočtů souřadnic) nebo do centrální anulaci systému.

Příklady programování změny měřítka:

```
G36 X1.27      "Změna měřítka, zvětšení 1.27
G36 Y0.38      "Změna měřítka, zmenšení 0.38
G36 X1.54 Y5.28 "Změna měřítka, zmenšení 1.54/5.28 = 0.2916
G36            "Odvolání změny měřítka
```

Příklad pro změnu měřítka:

"V příkladu se opakuje 3x volání stejného podprogramu. Při druhém volání se "zadá změna měřítka, při třetím volání se kromě změny měřítka zadá také "zrcadlení (Podprogram se volá s různou poloměrovou korekcí)

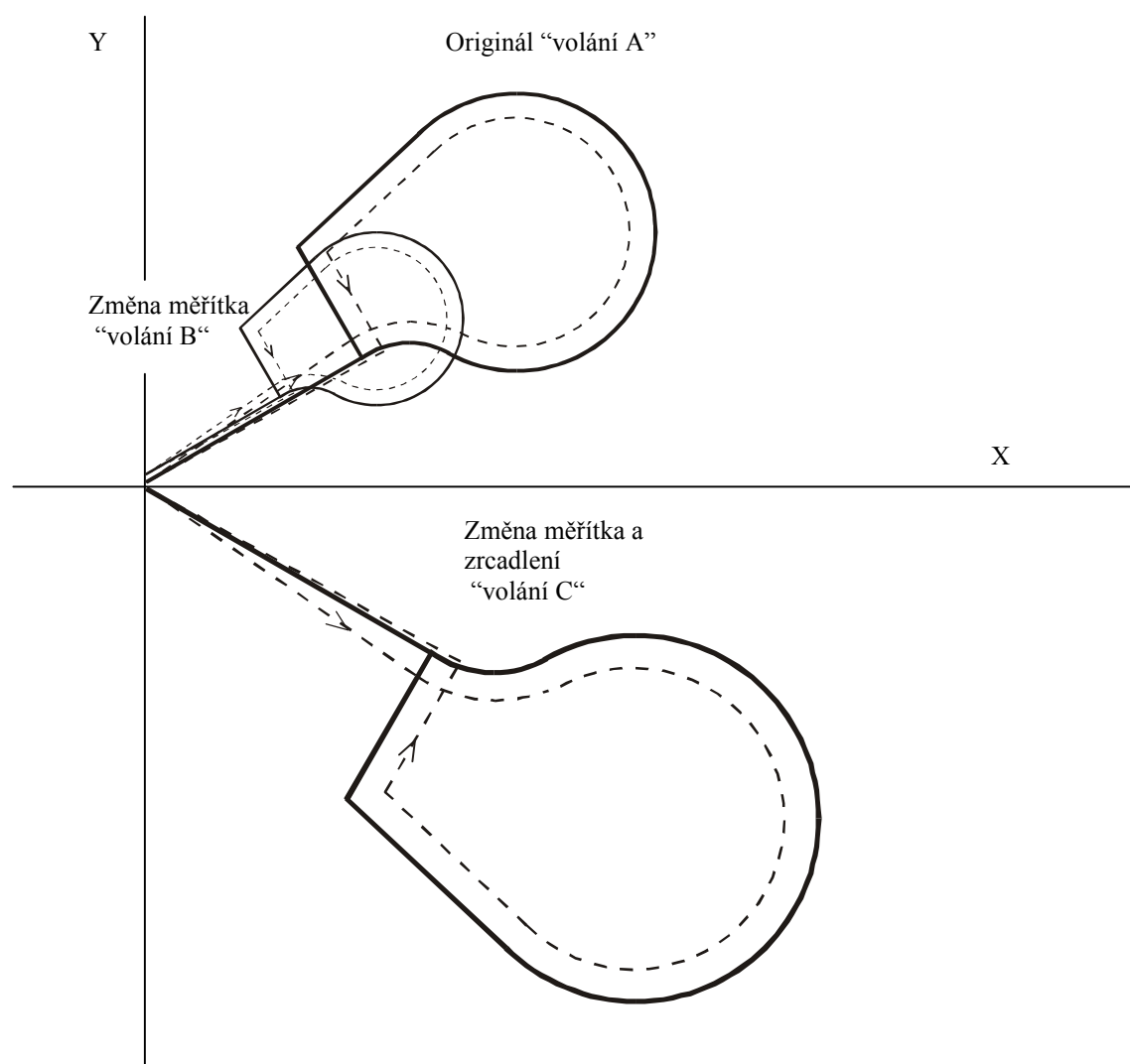
```
%1
N10 X0 Y0 F5000 G23
N20 G71 L100 D3      "Volání podprogramu - A
N30 G36 X1.54 Y3.28  "Změna měřítka zmenšení 1.54/3.28 = 0.46951
N40 G71 L100 D4      "Volání podprogramu - B
N50 G35 Y0           "Zrcadlení (převádí směr Y)
N50 G36 X1.27        "Změna měřítka zvětšení 1.27
N60 G71 L100 D5      "Volání podprogramu - C
N90 G35              "Zrušení zrcadlení
N95 G36              "Zrušení změny měřítka
N99 M30
```

"Podprogram s poloměrovou korekcí

```

N1000 G79 L100    "Podprogram L100
N1010 G01 X0 Y0
N1020 G01 X30 Y20 G41
N1030 G02 X50 Y24.849 I16.641 J-24.962
N1040 G03 X30 Y73.484 I3.359 J29.811
N1050 G01 X20 Y61.075
N1060 G01 X30 Y20
N1070 G01 X0 Y0 G40
N1080 G70

```



13.4 Nezávislý posun

Nezávislým posunem se získá posunutý tvar a programuje se pomocí funkce **G37**. Nezávislý posun je nezávislý na aktuálním posunutí počátku G53 – G59 a projeví se jako další aditivní složka posunu dráhy.

Účinek nezávislého posunu je stejný jako posunutí dráhy pomocí G53 až G59, jen posun se nezadáva pomocí tabulky posunutí, ale přímo v programu pomocí funkce G37.

Změnou posunu také není ovlivněna délková a poloměrová korekce nástroje (ani posunutí hlav u dvouhlavého soustruhu).

Hodnota nezávislého posunu se zadává přímo při programování funkce G37 hodnotou v jednotlivých souřadnicích, které představují vektor posunu vzhledem k nulovému bodu stroje.

Programování:

V samostatném nepohybovém bloku se naprogramuje **G37** a zadají se osy, které určují velikost nezávislého posunu.

Když je v samostatném bloku programována samotná funkce **G37** bez os, nezávislý posun se zruší.

Platný posun platí do změny pomocí funkce G37 nebo naprogramováním G34 (zrušení všech druhů přepočtů souřadnic) nebo do centrální anulaci systému.

Příklady programování nezávislého posunu:

```
G37 X10.127      "Posunutí osy X o 10.127
G37 X-0.138 Y12  "Posunutí osy X o -0.138 a Y o 12.0
G37              "Odvolání nezávislého posunu
```

Příklad pro nezávislý posun:

"V příkladu se opakuje 2x volání stejného podprogramu. Při druhém volání se "zadá posun

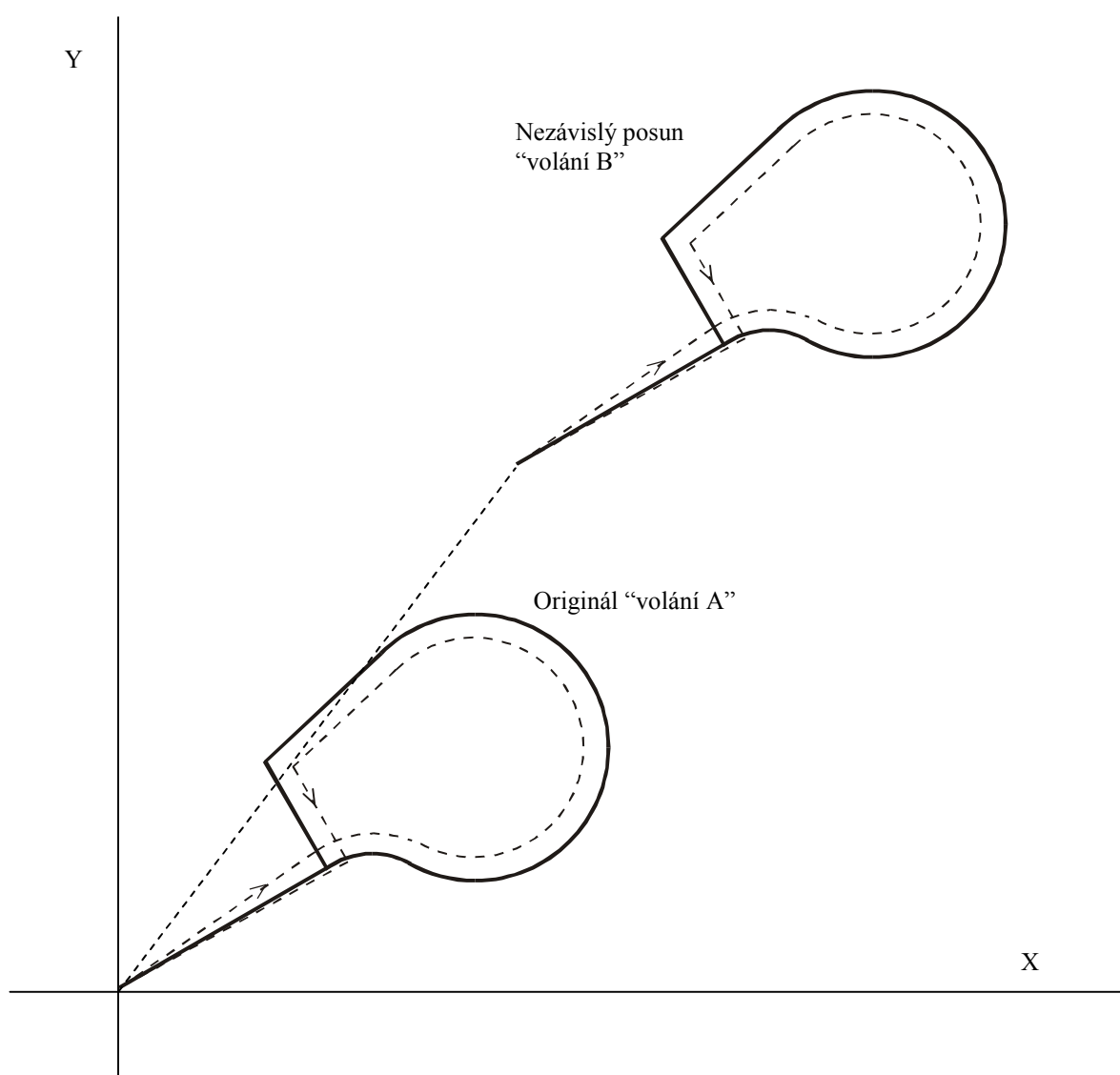
```
%1
N10 X0 Y0 F5000 G23
N20 G71 L100 D3      "Volání podprogramu - A
N30 G37 X80 Y120     "Nezávislý posun o (80,120)
N40 G71 L100 D4      "Volání podprogramu - B
N90 G37              "Odvolání posunu
N99 M30
```

"Podprogram s poloměrovou korekcí

```

N1000 G79 L100    "Podprogram L100
N1010 G01 X0 Y0
N1020 G01 X30 Y20 G41 D3
N1030 G02 X50 Y24.849 I16.641 J-24.962
N1040 G03 X30 Y73.484 I3.359 J29.811
N1050 G01 X20 Y61.075
N1060 G01 X30 Y20
N1070 G01 X0 Y0 G40
N1080 G70

```



14

14. TRANSFORMACE SOUŘADNÉHO SYSTÉMU

Transformace souřadného systému je implementována od softwarové verze 40.19 primárního procesoru a 6.201 sekundárního procesoru formou příslušenství na zvláštní požadavek.

14.1 Společné zásady

Systém umožňuje použít nezávisle na sobě 2 rovinné transformace (2D) a jednu prostorovou (3D) transformaci souřadnic. Každá souřadnice může být najednou zařazena jen v jedné transformaci.

Řízení transformace souřadnic je umožněno z NC partprogramu, z PLC programu a také pomocí speciálního dialogového okna.

Pod pojmem **fiktivní prostor** (fiktivní souřadnice) rozumíme ten prostor, ve kterém se pracuje. V tomto prostoru probíhá NC program a v něm také pracuje interpolátor systému. Fiktivní prostor je před transformací totožný s původním prostorem, daným referencí a nulovými body stroje. Systém při transformaci zobrazuje souřadnice podle fiktivních souřadnic.

Pod pojmem **reálný prostor** (reálné souřadnice) rozumíme skutečný prostor, ve kterém se obrábí. Výstup z interpolátoru systému je transformován z fiktivních do reálných přírůstků dráhy. Reálné souřadnice systému je možno vidět jen ve speciální obrazovce pro diagnostiku transformace. Softwarové limitní spínače a tabulkové nelineární korekce pracují podle reálných souřadnic systému.

Protože transformace souřadného systému probíhá až na výstupu z interpolátoru, není transformací ovlivněn průběh NC programu včetně posunutí, délkových a poloměrových korekcí a ani dalších přepočtů souřadnic.

Transformace souřadnic se týká všech druhů pohybu v systému, takže platí pro interpolátor, všechny ruční pohyby (AUTMAN), vlečení os, kopírování a také pro polohovací jednotky řízené z PLC programu.

Transformační parametry pro rovinnou transformaci jsou vypočteny ze zadaného posunutí a úhlu natočení. Pro prostorovou transformaci je nutno zadat tři úhly natočení.

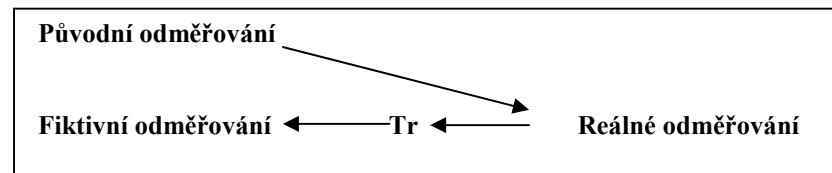
Řízení transformace se skládá ze třech hlavních úkonů:

- ⇒ Nastavení transformačních parametrů
- ⇒ Aktivace transformace
- ⇒ Dezaktivace transformace

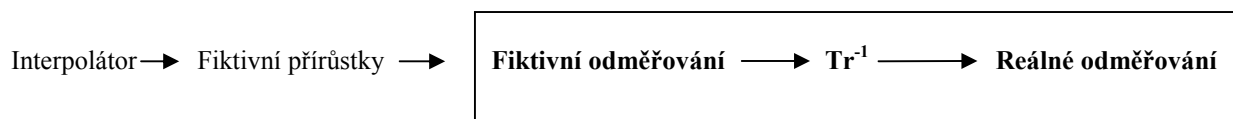
14.2 Principy rovinné transformace souřadnic

Při aktivaci transformace souřadnic se bez pohybu okamžitě přepočtou souřadnice systému na fiktivní souřadnice. Nejdříve se původní odměřování souřadnic vloží do reálného odměřování a potom se dopřední transformací vypočte fiktivní odměřování systému. Od toho okamžiku se fiktivní odměřování, které je tvořeno sumací fiktivních přírůstků, zpětnou transformací převádí na reálné odměřování systému průběžně až do odvolání transformace.

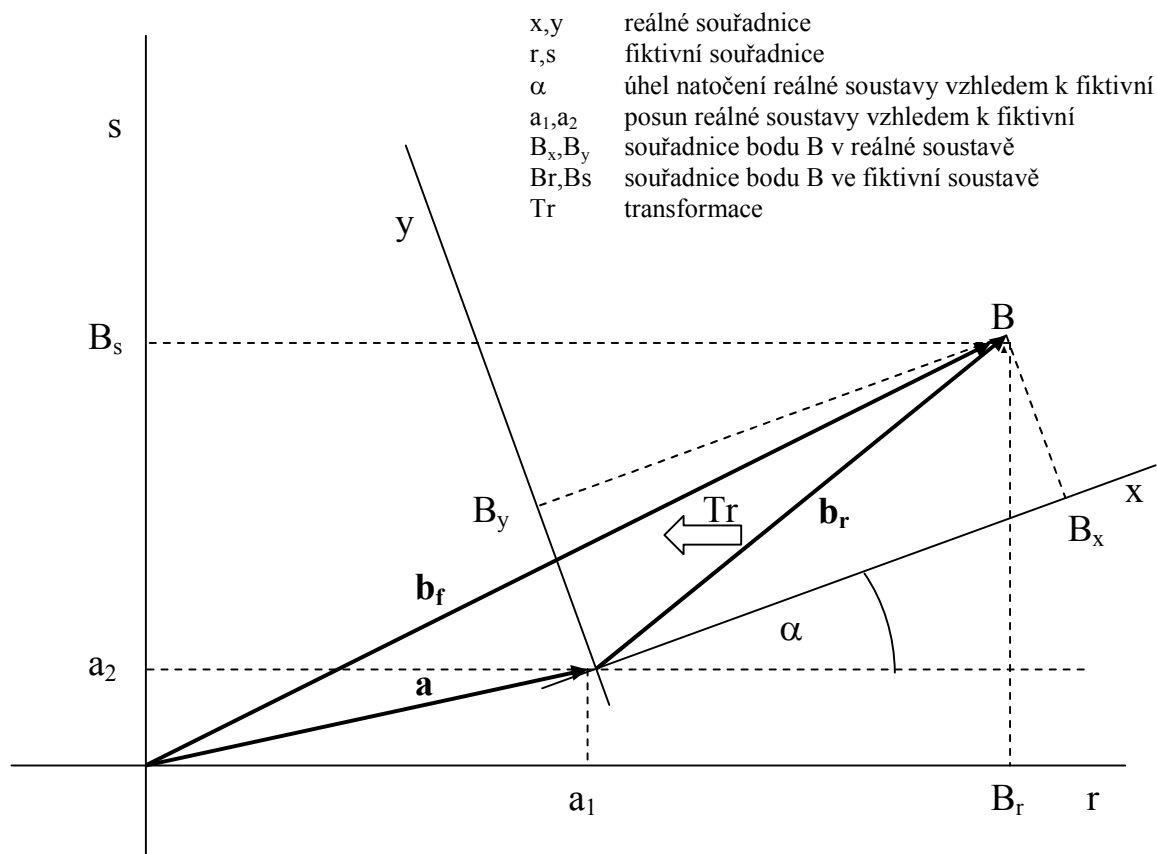
a) Jednorázový přepočet při aktivaci transformace:



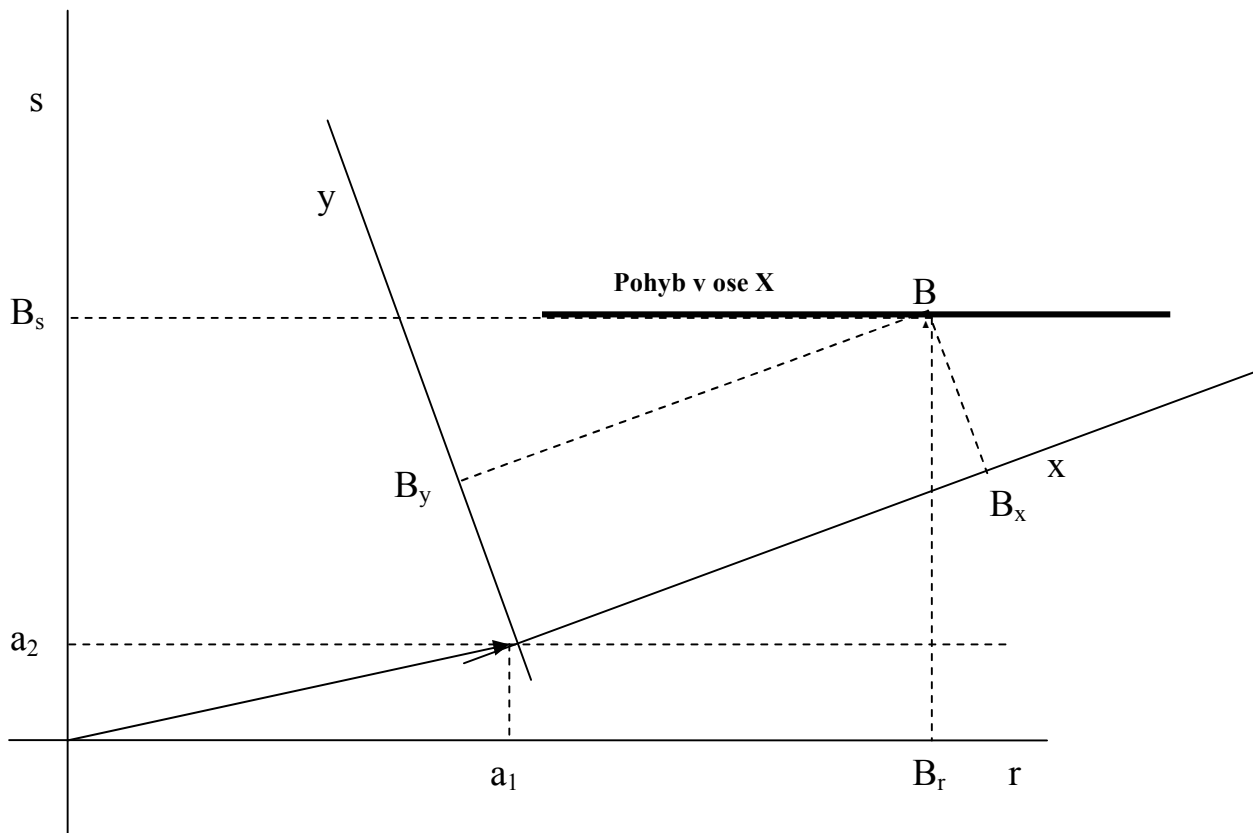
b) Průběžné zpracování při aktivní transformace:



Na příkladu rovinné 2D transformace si ukážeme způsob aktivace a deaktivace transformace. Označení os pro reálnou soustavu je x, y, z a označení os pro fiktivní soustavu je r, s, t . Vektory jsou označeny tlustým písmem.

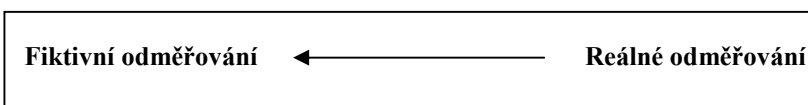


Ve fiktivním prostoru se vykonává NC program a v něm také pracuje interpolátor systému. V reálném prostoru se pohyby jeví v transformovaném stavu. Například pohyb jen v ose x způsobí v reálné soustavě pohyb dvou souřadnic:



c) Zrušení transformace

Obnovení původního stavu souřadného systému při deaktivaci transformace se vykoná přepisem reálného odměřování do fiktivního odměřování systému.



Rovnice transformace souřadnic v rovině se dá vyjádřit maticovou rovnicí:

$$\mathbf{a} + \mathbf{T} \cdot \mathbf{x} = \mathbf{r}$$

kde $\mathbf{T}$ je transformační matice

$$\begin{pmatrix} a1 \\ a2 \end{pmatrix} + \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} r \\ s \end{pmatrix}$$

Rovnice pro zpětnou rovinnou transformaci jsou vyjádřeny:

$$\mathbf{T}^{-1} \cdot \mathbf{r} - \mathbf{T}^{-1} \cdot \mathbf{a} = \mathbf{x}$$

kde $\mathbf{T}^{-1}$ je inverzní transformační matice

$$\begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} r - a_1 \\ s - a_2 \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix}$$

Označení:

x,y souřadnice reálné soustavy
 r,s fiktivní souřadnice
 α úhel natočení reálné soustavy vzhledem k fiktivní soustavě
 a₁,a₂ posun reálné soustavy vzhledem k fiktivní soustavě

14.3 Principy prostorové transformace souřadnic

Prostorová transformace je založena na podobných principech jako je rovinná transformace, ale je určena třemi souřadnicemi posunu a třemi úhly natočení. Natočení je přitom vždy kolem os. U prostorové transformace závisí na pořadí jednotlivých úkonů.

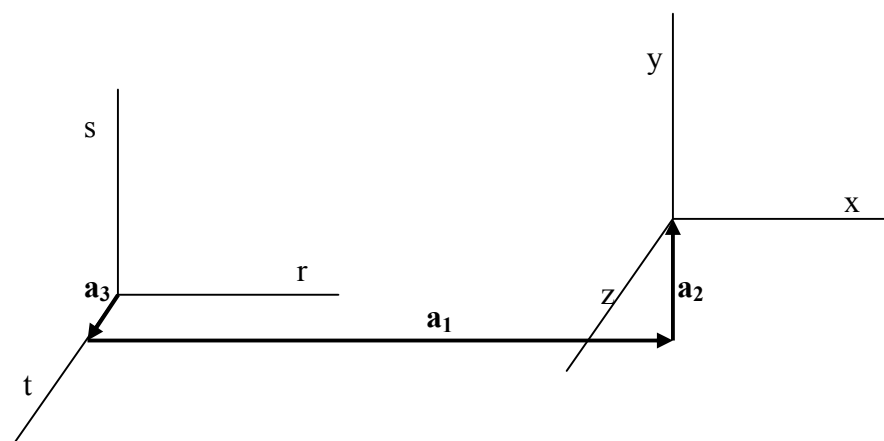
Označení:

x,y,z souřadnice reálné soustavy
 r,s,t fiktivní souřadnice
 α, β, χ úhly natočení reálné soustavy kolem svých os
 a₁,a₂,a₃ posun reálné soustavy vzhledem k fiktivní soustavě

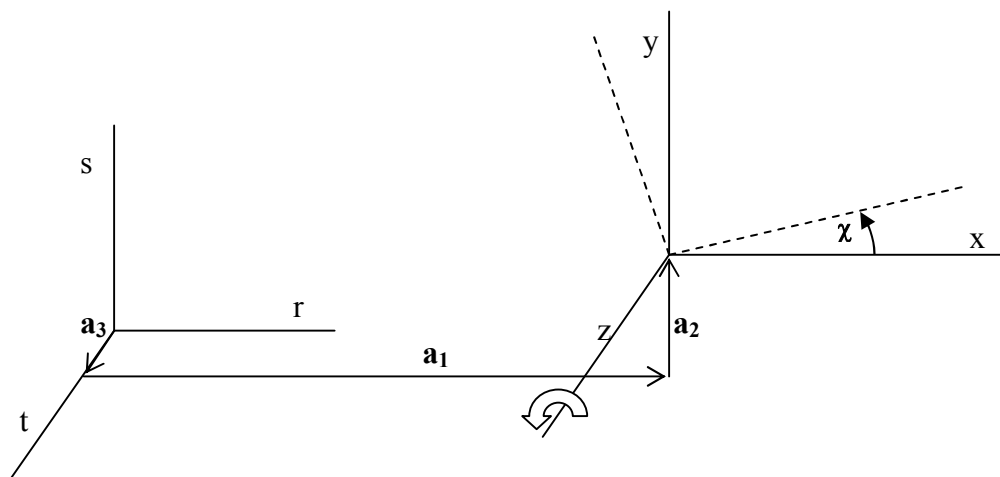
Parametry transformace α, β, χ a a₁,a₂,a₃ jsou:

a₁ posunutí reálného systému vzhledem k fiktivní soustavě ve směru **r**
a₂ posunutí reálného systému vzhledem k fiktivní soustavě ve směru **s**
a₃ posunutí reálného systému vzhledem k fiktivní soustavě ve směru **t**
 α úhel natočení reálného systému okolo osy **x**
 β úhel natočení reálného systému okolo osy **y**
 χ úhel natočení reálného systému okolo osy **z**

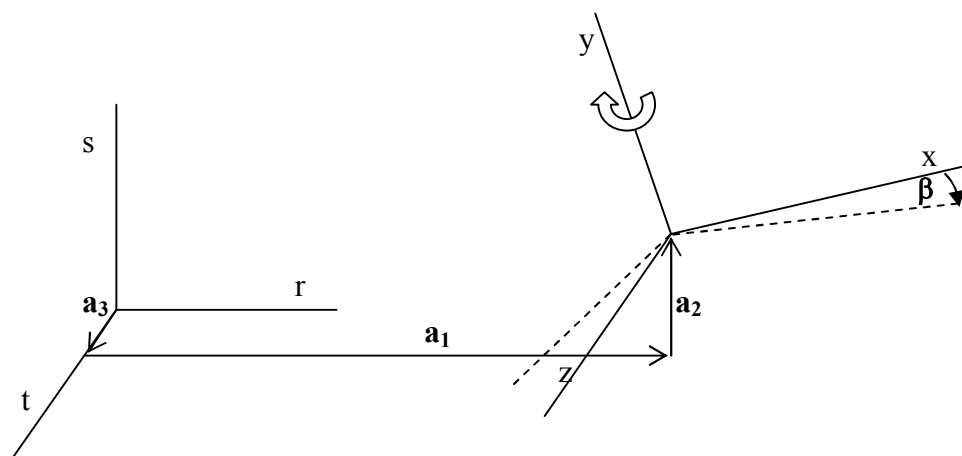
Posunutí reálného souřadnicového systému vůči fiktivnímu systému:



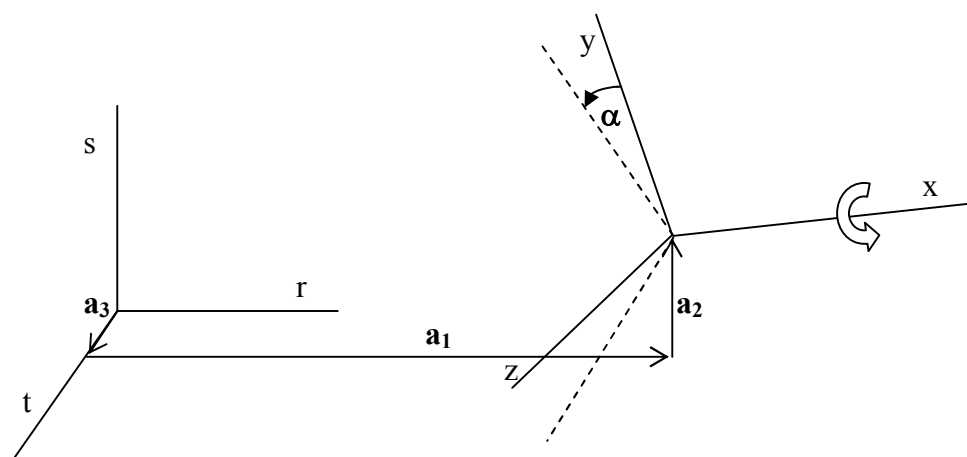
Natočení reálného souřadného systému kolem osy z o úhel χ :



Natočení reálného souřadného systému kolem osy y o úhel β :



Natočení reálného souřadného systému kolem osy x o úhel α :



Rovnice transformace souřadnic v prostoru se dá vyjádřit maticovou rovnicí:

$$\mathbf{a} + \mathbf{T} \cdot \mathbf{x} = \mathbf{r} \quad \text{kde } \mathbf{T} \text{ je prostorová transformační matice}$$

Transformační matice $\mathbf{T}$ je složena z jednotlivých rovinných transformačních matic, přitom záleží na pořadí jednotlivých rotací:

$$\mathbf{T} = \mathbf{T}_{yz} \cdot \mathbf{T}_{xz} \cdot \mathbf{T}_{xy}$$

1.rotace: $\mathbf{T}_{xy}$ je rovinná transformační matice při natočení kolem osy $\mathbf{Z}$ o úhel χ .

$$T_{xy} = \begin{pmatrix} \cos(\chi) & -\sin(\chi) & 0 \\ \sin(\chi) & \cos(\chi) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

2.rotace: $\mathbf{T}_{xz}$ je rovinná transformační matice při natočení kolem osy $\mathbf{Y}$ o úhel β .

$$T_{xz} = \begin{pmatrix} \cos(\beta) & 0 & -\sin(\beta) \\ 0 & 1 & 0 \\ \sin(\beta) & 0 & \cos(\beta) \end{pmatrix}$$

3.rotace: $\mathbf{T}_{yz}$ je rovinná transformační matice při natočení kolem osy $\mathbf{X}$ o úhel α .

$$T_{yz} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

Výsledná rovnice transformace souřadnic v prostoru je:

$$\begin{pmatrix} a1 \\ a2 \\ a3 \end{pmatrix} + \begin{pmatrix} \cos(\beta)\cos(\chi) & -\cos(\beta)\sin(\chi) & -\sin(\chi) \\ \cos(\alpha)\sin(\chi) - & \cos(\alpha)\cos(\chi) + & -\sin(\alpha)\cos(\beta) \\ -\sin(\alpha)\sin(\beta)\cos(\chi) & +\sin(\alpha)\sin(\beta)\sin(\chi) & \\ \cos(\alpha)\sin(\beta)\cos(\chi) + & \sin(\alpha)\cos(\chi) - & \cos(\alpha)\cos(\beta) \\ +\sin(\alpha)\sin(\chi) & -\cos(\alpha)\sin(\beta)\sin(\chi) & \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} r \\ s \\ t \end{pmatrix}$$

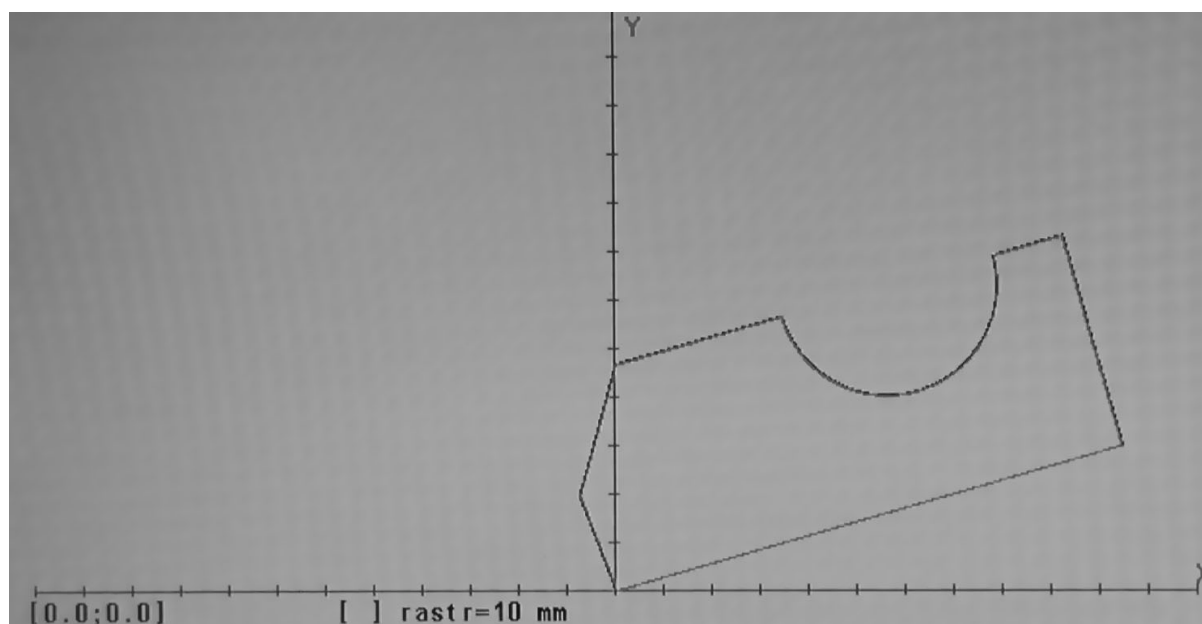
Rovnice pro zpětnou prostorovou transformaci je vyjádřena:

$$\mathbf{T}^{-1} \cdot \mathbf{r} - \mathbf{T}^{-1} \cdot \mathbf{a} = \mathbf{x}$$

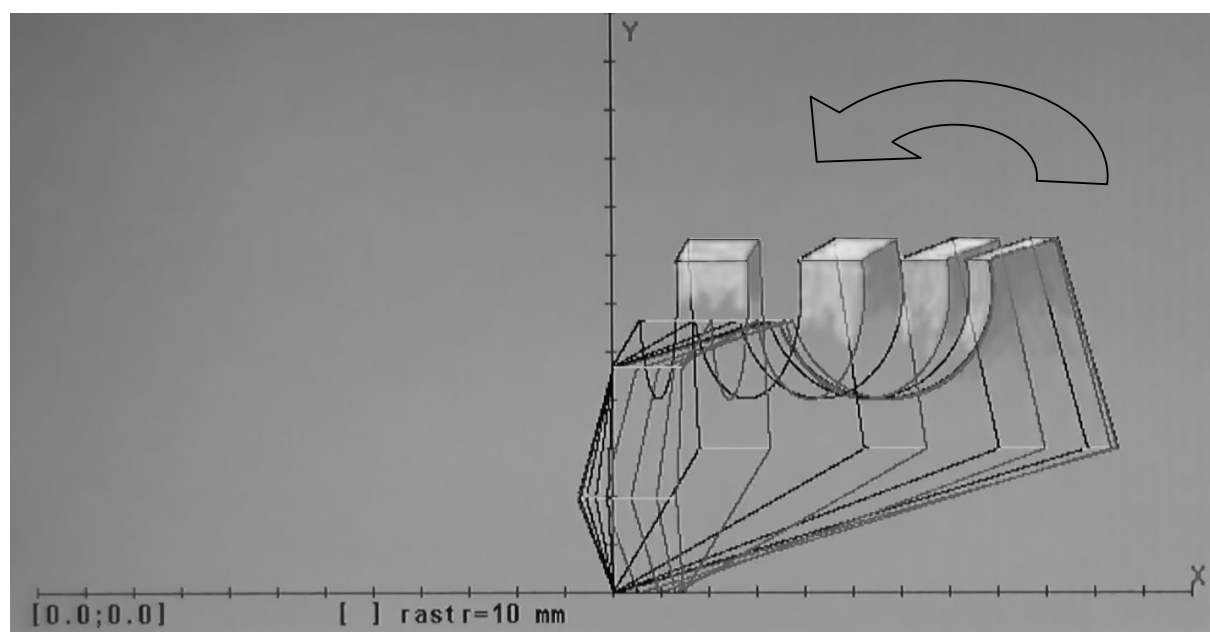
$$\begin{pmatrix} \cos(\beta)\cos(\chi) & \cos(\alpha)\sin(\chi) - \cos(\alpha)\sin(\beta)\cos(\chi) + \sin(\alpha)\sin(\chi) \\ -\cos(\beta)\sin(\chi) & \cos(\alpha)\cos(\chi) + \sin(\alpha)\sin(\beta)\sin(\chi) - \cos(\alpha)\sin(\beta)\sin(\chi) \\ -\sin(\beta) & -\sin(\alpha)\cos(\beta) & \cos(\alpha)\cos(\beta) \end{pmatrix} \begin{pmatrix} r - a1 \\ s - a2 \\ t - a3 \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Příklad natáčení v prostoru:

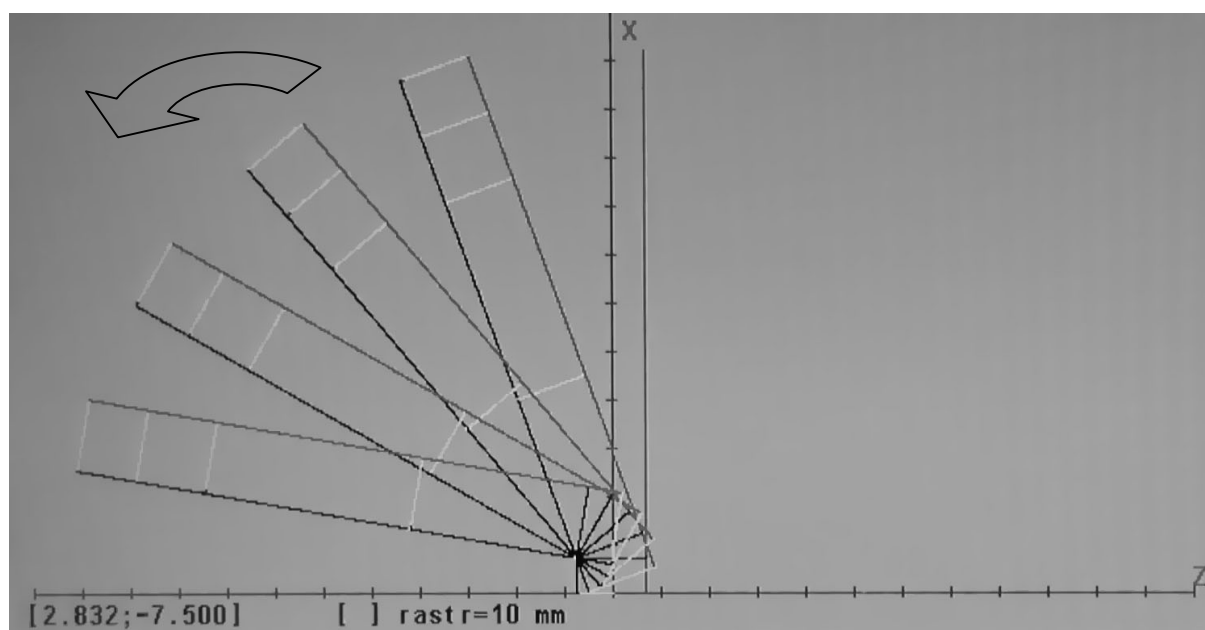
Původní tvar NC programu před natáčením. Těleso má tloušťku 10 mm.



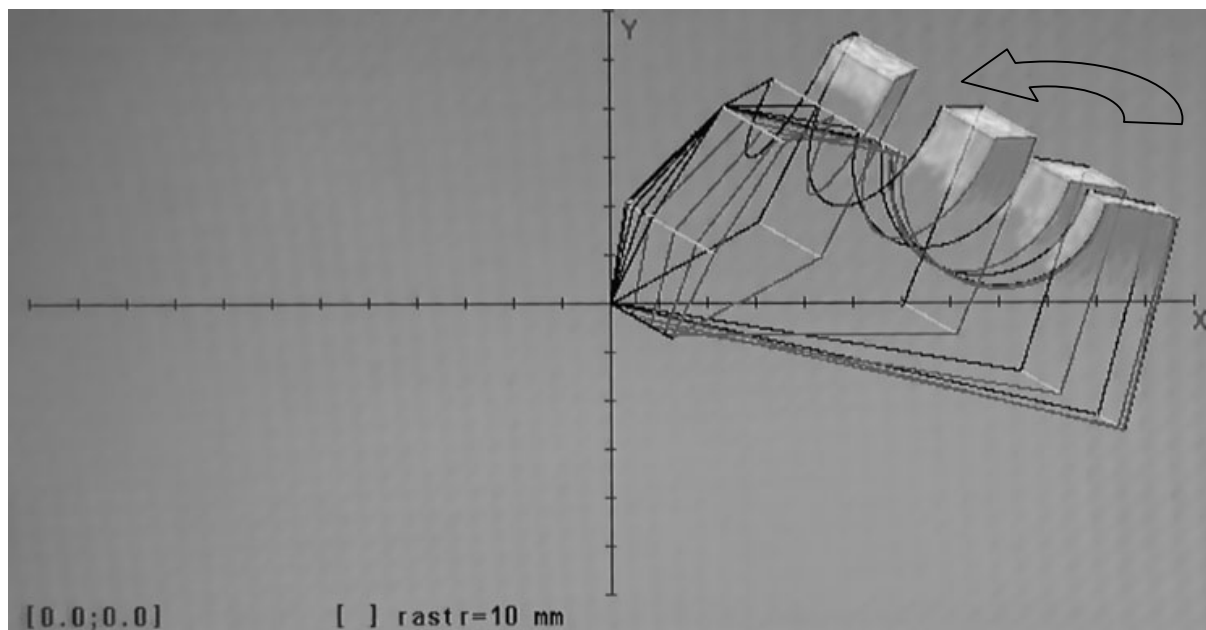
Natočení kolem osy Y ($\alpha=0^\circ$; $\beta=0, 20, 40, 60^\circ$; $\chi=0^\circ$), pohled z reálné soustavy XY:



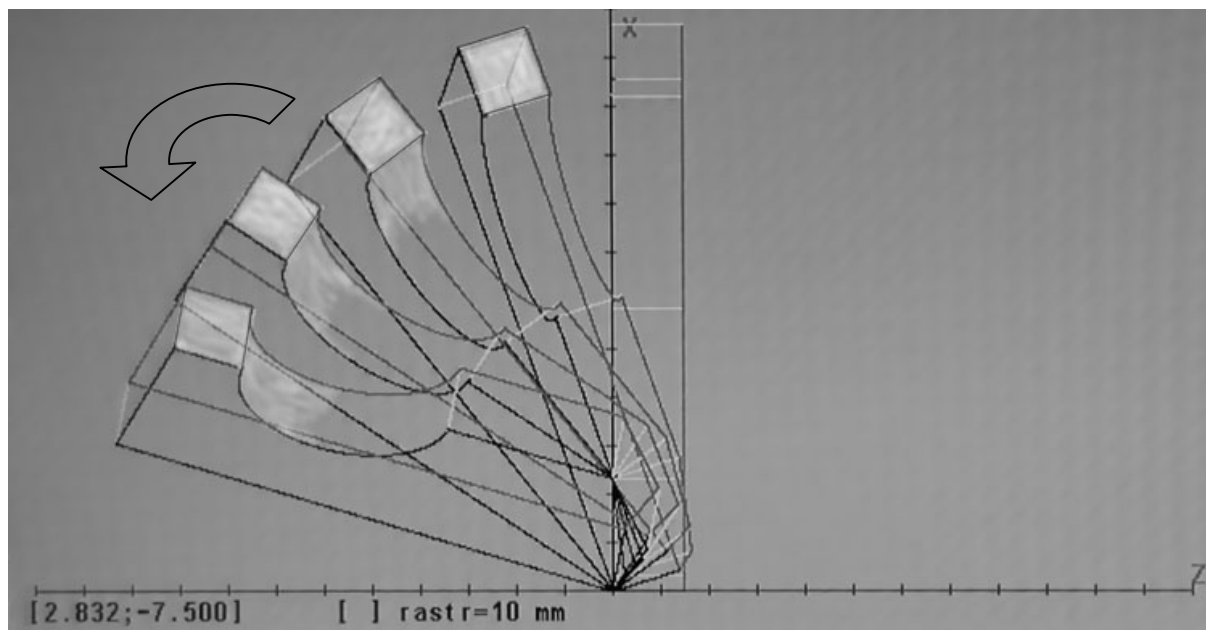
Natočení kolem osy Y ($\alpha=0^\circ$; $\beta=0, 20, 40, 60^\circ$; $\chi=0^\circ$), pohled z reálné soustavy XZ:



Natočení kolem osy Z a Y ($\alpha=0^\circ$; $\beta=0, 20, 40, 60^\circ$; $\chi=30^\circ$), pohled z reálné soustavy XY:



Natočení kolem osy Z a Y ($\alpha=0^\circ$; $\beta=0, 20, 40, 60^\circ$; $\chi=30^\circ$), pohled z reálné soustavy XZ:



14.4 Řízení transformací z NC programu

Pro nastavování, aktivaci a deaktivaci transformací slouží jedenáctá skupina G funkcí. Účinek výkonu G funkcí je možno okamžitě sledovat pomocí diagnostické obrazovky, kterou možno vyvolit pomocí volby indikace (tlačítko WIN).

Použité G funkce:

Funkce z 11. skupiny G funkcí:	Přidělení
G38	Řízení 1. rovinné transformace - 2D(1)
G39	Řízení 2. rovinné transformace - 2D(2)
G31	Řízení prostorové transformace - 3D(1)

14.4.1 Nastavení parametrů transformace z NC programu

Nastavení parametrů transformace, jako jsou posunutí a natočení reálné soustavy vzhledem k fiktivní, možno programovat pomocí funkcí G38, G39 a G31 a hodnoty se zadávají pomocí adres X, Y, Z, ..., I, J, K, E. Nastavení parametrů transformace se musí zadávat v nepohybovém bloku NC programu a nesmí se provádět, když je transformace aktivována.

Adresy souřadnic **X, Y, Z,...** definují posunutí a současně definují osy, které vstupují do transformace. Pro rovinnou transformaci musí být použity 2 souřadnice a pro prostorovou transformaci musí být použity 3 souřadnice.

Adresy **I, J, K** slouží pro zadávání úhlů natočení. Pro rovinnou transformaci musí být zadán jeden úhel pomocí adresy I a pro prostorovou transformaci musí být zadány tři úhly pomocí adres I, J, a K.

Adresa **E** může sloužit pro potřebu zpřesnění zadání. V případě, že je nutno zadávat úhly přesněji než na tisícinu stupně, může se pomocí adresy E zadat násobitel (mírka). Například pro E=100 se hodnoty úhlů zadávají v stotisícinách stupně.

G funkce pro transformaci	Adresy zadání	Význam	Příklad
G38, G39, G31	X, Y, Z, U, V, W, A, B, C	Zadává se posunutí reálné soustavy vzhledem k fiktivní soustavě. Souřadnice současně definují osy, které vstupují do transformace. Pro rovinnou transformaci musí být použity 2 souřadnice a pro prostorovou souřadnicí musí být použity 3 souřadnice.	Pro G38, G39: X12.5 Z-22.5 Pro G31: X0 Y10.1 Z1.234
	I, J, K	Zadávají se úhly natočení reálné soustavy vzhledem k fiktivní soustavě: Pro rovinnou transformaci se zadává úhel natočení pomocí adresy I . Pro prostorovou transformaci se zadává : Úhel α natočení kolem 1.osy (X) pomocí adresy I . Úhel β natočení kolem 2.osy (Y) pomocí adresy J . Úhel χ natočení kolem 3.osy (Z) pomocí adresy K .	Pro G38, G39: I=30.5 (natočení 30.5°) Pro G31: I=0, J=-30.4, K=20 ($\alpha=0^\circ$, $\beta=30.4^\circ$ $\chi=20^\circ$)
	E	Nepovinná adresa pro možnost zpřesnění zadání úhlů natočení. Hodnoty natočení zadané pomocí parametrů I, J, K se touto adresou vydělí. Pokud se například naprogramuje na hodnotu 10, budou hodnoty úhlu zadány v desetitisícinách úhlu.	E=10 I=12.345 Skutečná hodnota natočení: 1.2345°

Příklady:

Pro 1. rovinovou transformaci v osách X, Z, úhlu natočení 22.5° a posunutí (10.5 , 15.1).

```
N10 G38 X10.5 Z15.1 I22
```

Pro prostorovou transformaci v osách X,Y, Z úhlu natočení kolem osy Z= 22.5° a kolem osy X=12° a posunutí (1.5 , 0, -15.1).

```
N10 G31 X1.5 Y0 Z-15.1 I12 J0 K22.5
```

14.4.2 Aktivace a deaktivace transformace z NC programu

Aktivace transformace se provede naprogramováním jedné z funkcí **G31**, **G38** a **G39** bez dalších parametrů. Předpokládá se, že hodnoty parametrů transformace už byly nastaveny nebo obnoveny ze zálohy. Při aktivaci transformace se neprovede žádný pohyb, jen se změní hodnoty odměřování přepočtem transformace z reálné soustavy. Po aktivaci transformace by neměl následovat bezprostředně pohybový blok, protože systém musí načerpat nové transformované souřadnice. Je vhodné po aktivaci transformace zařadit jeden prázdný blok.

Příklady:

Aktivace 2.rovinné transformace:

```
N10 G39          "aktivace transformace
N20              "prázdný blok
```

Aktivace prostorové transformace:

```
N10 G31          "aktivace transformace
N20              "prázdný blok
```

Dezaktivace transformace se provede naprogramováním jedné z funkcí **G31**, **G38** a **G39** s parametrem **I0**. Při deaktivaci transformace se neprovede žádný pohyb, jen se změní hodnoty odměřování přepočtem zpětné transformace z fiktivní soustavy. Po deaktivaci transformace by neměl následovat bezprostředně pohybový blok, protože systém musí načerpat nové souřadnice. Je vhodné po deaktivaci transformace zařadit jeden prázdný blok.

Příklady:

Dezaktivace 1.rovinné transformace:

```
N10 G38 I0       "dezaktivace transformace
N20              "prázdný blok
```

Dezaktivace prostorové transformace:

```
N10 G31 I0       "dezaktivace transformace
N20              "prázdný blok
```

14.5 Řízení transformací pomocí dialogového okna

Volba příslušného dialogového okna se provádí pomocí tlačítka panelu systému, které aktivují makra:

6000H dialogové zadávání pro rovinnou transformaci 2D(1)
 6100H dialogové zadávání pro rovinnou transformaci 2D(2)
 6200H dialogové zadávání pro prostorovou transformaci 3D(1)

Makra se přiřadí příslušnému tlačítku v souboru **KLAV.KNF** (viz příloha L – tlačítka panelu systému).

Stiskem příslušného tlačítka se v pravé části obrazovky objeví obrazovka pro dialogové zadávání hodnot transformace a v levé části obrazovky se objeví obrazovka diagnostiky transformace. Pomocí obrazovky diagnostiky transformace možno průběžně sledovat aktuální stav transformací. Opětovným stiskem téhož tlačítka se stav obrazovek vrátí do původního stavu.

Obrazovka pro dialogové zadávání rovinné transformace:

The screenshot shows a dialog box titled "Transformace os". It has a dark green header bar with the title in white. Below the header, the text "Typ transformace:" is followed by three options: "2D(1)" (highlighted in blue), "2D(2)", and "3D(1)". Below this are three buttons: "AKTIVOVAT" (blue), "ZRUSIT" (blue), and "ZADAT" (blue). Underneath these buttons are two labels: "Rovina" followed by a blue square, and "Prostor" followed by a grey square. Below these is the section "Parametry transformace:". It contains two columns of input fields. The left column has labels "a1", "a2", and "a3" next to three input fields (the top two are blue, the bottom one is grey). The right column has labels " α ", " β ", and " γ " next to three input fields (the top one is blue, the bottom two are grey). Below the input fields are two buttons: "ULOZIT" (blue) and "UKONCIT" (blue). At the bottom of the dialog is a dark green bar with white text: "Aktivovat / Zrusit / Zadat transformaci" and "→ ↑ ↓ ← posun kursoru, ↵ potvrzení".

Obrazovka pro dialogové zadávání prostorové transformace:

Způsob zadávání parametrů transformace a její řízení pomocí dialogového okna je intuitivní.

V horní části se pro informaci zobrazí typ transformace:

- 2D(1)** 1.rovinná transformace
- 2D(2)** 2.rovinná transformace
- 3D(1)** prostorová transformace

V druhém řádku se nejdříve zvolí jeden z režimů:

AKTIVOVAT – ZRUŠIT - ZADAT

Volba se vykonává pomocí kurzorových tlačítek a potvrzením tlačítkem **ENTER**. Vyvolený režim je na obrazovce zvýrazněn. Když se zvolí režim **AKTIVOVAT** a **ZRUŠIT**, nastaví se aktivní oblast na políčko **UKONČIT**. Pokud teď stiskneme tlačítko **ENTER**, opustíme dialogové zadávání a obnoví se původní stav obrazovky.

Když se zvolí režim **ZADAT**, zaktivní se políčko pro zadání os, které vstupují do transformace. Pro rovinné transformace se musí zadat 2 a pro prostor 3 různé existující souřadnice. Názvy os musí být stejné, jako jsou nakonfigurovány ve strojních konstantách.

Po vyplnění názvů os se zvýrazní první políčko pro zadávání parametrů transformace.

Zadávají se posunutí reálné soustavy vzhledem k fiktivní, desetinná tečka má význam milimetrů:

- pro rovinnou transformaci: **a1**, **a2**.
- pro prostorovou transformaci: **a1**, **a2**, **a3**.

Zadávat se úhly natočení reálné soustavy vzhledem k fiktivní, desetinná tečka má význam stupňů:

- pro rovinnou transformaci: α .
- pro prostorovou transformaci: α, β, χ .

Desetinná tečka se uvede podle požadované přesnosti. Nevyplněné hodnoty se nahradí nulou. Každé zadání se ukončí tlačítkem **ENTER** nebo kurzorovým tlačítkem.

Po vyplnění se pomocí kurzorových tlačítek nakráčí na políčko **ULOŽIT**. Po potvrzení tlačítkem **ENTER** se vypočtou všechny parametry potřebné pro transformaci, ale samotná transformace se ještě neprovede.

Po potvrzení tlačítkem **ENTER** se zvýrazní políčko **UKONČIT**. Pokud teď stiskneme tlačítko **ENTER**, opustíme dialogové zadávání a obnoví se původní stav obrazovek. Můžeme ale nakráčet pomocí kurzorových tlačítek například na režim **AKTIVOVAT** a hned zadanou transformaci aktivovat. Po každé akci nemusíme opustit dialogové okno, ale můžeme pokračovat dál v zadávání transformace.

Po aktivaci transformace je zakázáno zadávat nové parametry transformace, pokud se transformace neodvolá pomocí režimu **ZRUŠIT**.

14.6 Řízení transformace z PLC programu

V PLC programu je umožněno aktivovat a deaktivovat jednotlivé transformace a řídit zálohování parametrů transformace při vypnutí systému. Dále se problematika týká některých aspektů povolení pohybu transformovaných os, čtení odměřování fiktivních a reálných souřadnic. V PLC programu se normálně nepředpokládá možnost nastavování parametrů transformace, ale jen řízení její aktivace.

Celá problematika řízení transformace z PLC programu není součástí návodu na programování a podrobně je rozebrána v „Návodu k programování PLC“ v kapitole: „Doplňky“.

14.7 Diagnostika transformace souřadnic

Pro diagnostiku transformace souřadnic slouží speciální obrazovka, která se dá vyvolit pomocí volby indikace nebo stiskem tlačítek, která aktivují makra pro aktivaci dialogového okna řízení transformace. Na obrazovce diagnostiky transformace je možno sledovat reálnou a fiktivní polohu souřadnic, aktuální parametry transformace a stav aktivace jednotlivých transformací.

Na diagnostické obrazovce v horní části se zobrazují **fiktivní** a **reálné** souřadnice. Ve fiktivním prostoru probíhá NC program a v něm také pracuje interpolátor systému. Fiktivní prostor je před transformací totožný s původním prostorem, daným referencí a nulovými body stroje. Systém při transformaci zobrazuje souřadnice podle fiktivních souřadnic. Reálné souřadnice jsou skutečné souřadnice prostoru, ve kterém se obrábí. Výstup z interpolátoru systému je transformován z fiktivních do reálných přírůstků dráhy. Před aktivací transformace jsou fiktivní i reálné souřadnice totožné.

Ve střední části diagnostické obrazovky se zobrazují aktuální transformační parametry dvou rovinných transformací **2D(1)** a **2D(2)**. Ve spodní části se zobrazují transformační parametry pro prostorovou transformaci.

V řádku označeném **Tr** se indikují červenou barvou souřadnice, které vstupují do transformace. Pokud jsou místo souřadnic zobrazeny pomlčky, transformace není aktivována.

V řádku označeném **status** se zobrazuje v bitové formě aktuální stav transformace. Význam jednotlivých bitů je popsán v „Návodu na programování PLC, v kapitole Doplnky“ a mají význam hlavně pro návrháře PLC.

bit	název v PLC	Popis bitů stavu transformace
bit 0	TR_ENABLE	Použití transformace je povoleno
bit 1	TR_SET	Transformace je aktivní
bit 2	TR_RESTORE	Transformace není aktivní a bylo provedeno obnovení stavu

V řádcích označených **a1**, **a2**, **a3** se zobrazuje aktuální posunutí reálné soustavy vzhledem k fiktivní soustavě.

V řádcích označených **α** , **β** , **γ** se zobrazují aktuální úhly natočení kolem jednotlivých reálných os. Popis byl uveden v kapitolách: „Principy rovinné a prostorové transformace“.

Aktuální stav aktivace transformace je indikován na systému v řádku režimů speciálním znakem pro rovinnou a prostorovou transformaci:

Rovinná transformace:



Prostorová transformace:



Diagnostika transformace souřadnic			
osy:	fiktivní	reálné	
X	+100,120	+0,000	
Y	+200,330	+0,000	
Z	+0,000	+0,000	
B	+0,000	+0,000	
C	+0,000	+0,000	
-	+0,000	+0,000	

2D(1)	Tr: X Y	2D(2)	Tr: - -
status:	00000011	status:	00000001
a1:	+100,120	a1:	+0,000
a2:	+200,330	a2:	+0,000
α :	+30,400	α :	+0,000

3D(1)	Tr - - -	status:	00000001
a1:	+0,000	α :	+0,000
a2:	+0,000	β :	+0,000
a3:	+0,000	γ :	+0,000

Příklad diagnostické obrazovky. Právě je aktivní 1. rovinná transformace:

Diagnostika transformace souřadnic			
osy:	fiktivní	reálné	
X	+118,594	+19,456	
Y	+200,345	+2,178	
Z	+30,839	+17,341	
A	+0,000	+0,000	
V	+0,000	+0,000	
W	+0,000	+0,000	
2D(1) Tr: - -		2D(2) Tr: - -	
status: 00000001		status: 00000001	
a1: +0,000		a1: +0,000	
a2: +0,000		a2: +0,000	
α : +0,000		α : +0,000	
3D(1) Tr X Y Z		status: 00000011	
a1: +100,100		α : +20,330	
a2: +200,345		β : +0,000	
a3: +12,345		γ : +12,770	

Příklad diagnostické obrazovky. Právě je aktivní prostorová transformace:



Příloha A - plynulá návaznost bloků

Plynulost

Systémy CNC8x6 s procesorem v kazetě CPU04 (80486) a systémy řady CNC8x9 (DUAL) umožňují plynulé obrábění složitých tvarů, které jsou naprogramovány například pomocí velkého množství malých bloků. Pod pojmem **plynulost** rozumíme takovou vlastnost systému, když systém přejede bez poklesu rychlosti z jednoho bloku do následujícího pohybového bloku. Pro plynulý přejezd mezi bloky není nutnou podmínkou, aby bloky navazovaly na sebe tečně (směrnice pohybu obou bloků nemusí být shodné).

Obálková rychlost

Systémy řady CNC8x9 (DUAL) navíc umožňují pro řízení plynulosti použít tzv. „**obálkovou rychlost**“. Pod pojmem „obálková rychlost“ rozumíme takovou vlastnost systému, kdy systém jede **plynule** a přitom se snaží dosáhnout programovanou rychlost i v průběhu více bloků. Tím je umožněno, že i bloky o malé délce se jedou mnohonásobně větší rychlostí, než by se jelo bez řízení obálkové rychlosti. Z toho vyplývá, že systém musí předvídat a korigovat rychlost na základě určitých kritérií, které vyplynou v příštích blocích. Analýza kritérií z příštích bloků probíhá max. 500 bloků dopředu. Problematika obálkové rychlosti je rozebrána v této kapitole později.

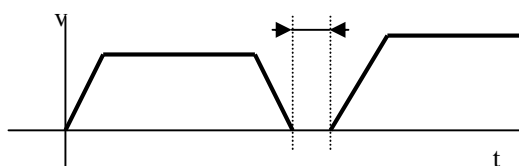
V dalším textu budou rozebrány možnosti problematiky plynulosti.

Příloha A1 - napojování bloků bez plynulé návaznosti

Standardní režim v napojování bloků je volen funkcí **G98**. Jedná se o implicitní nastavení po zapnutí systému. V tomto případě se pohyb v každém bloku začíná a končí rampou. Strmost rozjezdové a dojezdové rampy je dána strojní konstantou R52.

V systému neprobíhá příprava dalšího bloku programu napřed, a proto vzniká mezi bloky určitá časová prodleva daná výkonností hardwarové sestavy systému.

Standardní režim napojování bloků není vhodné používat v obrábění s vyššími nároky na kvalitu povrchu.



G98

Příloha A2 - těsné napojování bloků

Těsné napojování bloků se zařadí pomocí kontinuální funkce **G24**. V tomto případě se z panelu systému vysílá následující blok do kazety (nebo do sekundárního procesoru v systémech CNC8x9-DUAL) už v průběhu předchozího bloku. Pohyb v každém bloku začíná a končí rampou a po ukončení dojezdové rampy systém čeká na dojetí minimální odchylky podle strojní konstanty R06 – R07. Když systém vyhodnotí dojetí minimální odchylky, okamžitě rozjede následující blok.

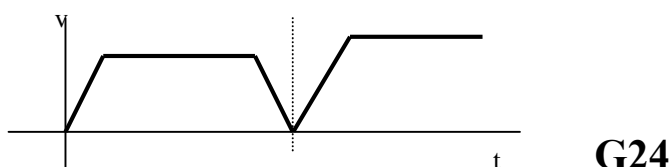
Podmínkou těsného napojování bloků je minimální doba trvání jednoho bloku. Pokud je dodržena minimální doba trvání bloku a je programována funkce G24, systém těsně napojí následující blok.

Minimální doby trvání pohybového bloku jsou pro těsné napojování bloků:

30 ms pro systémy řady CNC8x6

2 ms pro systémy řady CNC8x9

Plynulost pohybu v tomto případě ve velké míře závisí od dynamiky servopohonů a od strmosti rozjezdových a dojezdových ramp.



Příloha A3 - ruční řízení plynulé návaznosti

Nutnou podmínkou ručního řízení plynulé návaznosti je nastavení 1.dekády strojní konstanty R97 na hodnotu **1**.
1.dekáda strojní konstanty R97:

- 0** přechod mezi bloky bez počítání zbytků (nepoužívat)
- 1** ruční řízení plynulé návaznosti funkcemi: G23 – G24
- 2** automatické rozpoznávání plynulé návaznosti

Plynulá návaznost bloků v ručním řízení se ovládá funkcemi **G23** a **G24**. Pohybové bloky s pracovním posuvem, ve kterých je programována funkce G23, navážou na sebe plynule. Znamená to, že systém v tomto případě nezařadí žádnou rampu a nečeká na dojezd minimální odchylky. Navíc systém rozpočítá zbytky poslední dráhy v taktu bloku rovnoměrně v několika přejezdových taktech, takže pokud není velký rozdíl v tečnách (směrnice pohybu obou bloků), bude přejezd bez změny rychlosti. Větší rozdíl v tečnách navazujících bloků se projeví určitou změnou rychlosti přejezdu, která však je malá a v tomto případě není vůbec důležitá. Obecně možno říci, že čím bude menší rozdíl v tečnách obou navazujících bloků, tím bude přechod plynulejší.

Nutnou podmínkou pro plynulou návaznost je dodržení minimální délky trvání bloku pro plynulou jízdu.

Minimální doby trvání bloku pro plynulou jízdu jsou:

60 ms pro systémy řady CNC8x6

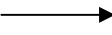
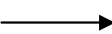
4 ms pro systémy řady CNC8x9

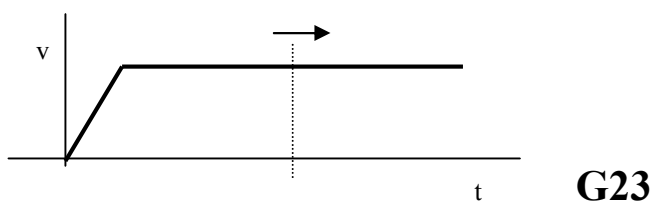
Systém při plynulé jízdě (bez aktivní obálkové rychlosti) kontroluje jen brzdovou dráhu následujícího bloku, to znamená, že v případě potřeby brzdí maximálně jeden blok předem. Proto je nutno ještě zabezpečit, aby nebylo

odvolání plynulé jízdy ve velmi krátkém bloku, který následuje po plynulých blocích s větší programovanou rychlostí.

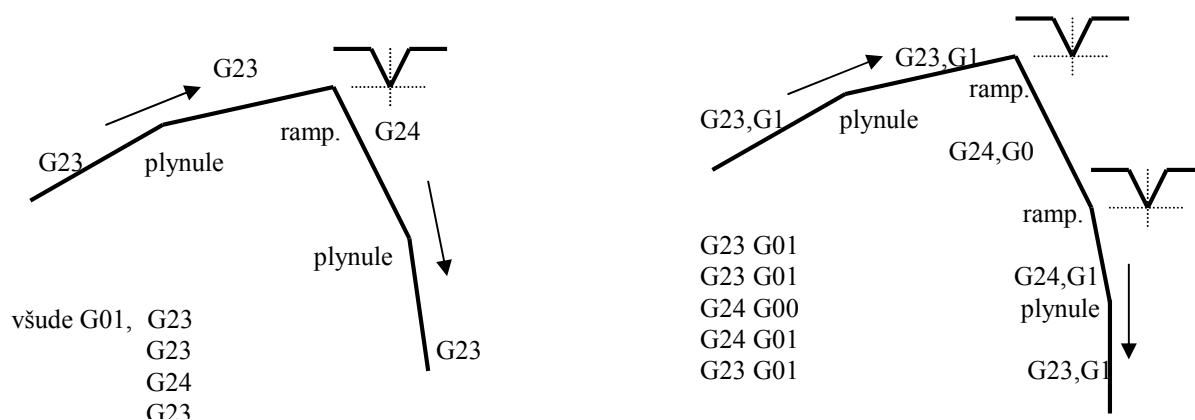
V době, když systém řídí pohyb pomocí plynulé návaznosti, určuje zrychlení (rampu) strojní konstanta **R236**. Když se systém neřídí plynulou návazností a je programován pracovní posuv, určuje zrychlení strojní konstanta R52. Pro rychloposuv se uplatní strojní konstanta R237.

Vyhodnocení plynulé návaznosti ve dvou po sobě následujících blocích:

Blok n-1	Blok n	Znak plynulosti	Popis plynulosti
G24	G24		Není plynulá návaznost
G23	G24		Není plynulá návaznost
G24	G23 (G01,G02,G03)		Je plynulá návaznost
G23	G23		Je plynulá návaznost



Příklady použití ručního řízení plynulé návaznosti bloků:



Příloha A4 - automatické rozpoznávání plynulé návaznosti

Nutnou podmínkou automatického řízení plynulé návaznosti je nastavení 1.dekády strojní konstanty R97 na hodnotu **2**. V současné verzi software je možno automatické rozpoznávání využít pro první 3 souřadnice.

Automatické rozpoznávání plynulé návaznosti bloků se zařadí pomocí funkce **G23**. Když je 1. dekáda strojní konstanty R97 nastavena na hodnotu 2, má funkce G23 jiný význam, než tomu bylo v ručním řízení plynulé návaznosti. V tomto případě **systém sám rozpoznává možnost plynulé návaznosti** a sám ji také řadí a vyřazuje. Ve většině případů postačí, když bude na začátku programu uvedena funkce G23, která je kontinuální a bude platit pro celý program. V dalším textu bude pojednáno o tom, jaké kritéria při rozpoznávání systém použije.

Automatické rozpoznávání plynulé návaznosti se ruší pomocí funkce **G24**. V tomto případě systém úplně vyřadí plynulou návaznost (i s automatickým rozpoznáváním) a bude řídit pohyb v těsném napojování bloků se zařazením rozjezdových a dojezdových ramp. Platí všechno, co bylo napsáno v části „Těsné napojování bloků.“ (Funkce G24 se při automatickém rozpoznávání používá jen výjimečně.)

- | | |
|------------------|--|
| G23 | Zařazení automatického rozpoznávání plynulé návaznosti, bude záviset na systému, zda rozpozná plynulé přejetí bloku. |
| G24 | Vyřazení plynulé návaznosti, systém řídí pohyb v těsném napojování bloků se zařazením rozjezdových a dojezdových ramp.
(Funkce se při automatickém rozpoznávání používá jen výjimečně.) |

V případě, když systém rozpozná plynulou návaznost a zařadí ji, bloky na sebe navážou plynule (podobně jako při G23 u ručního řízení plynulé návaznosti). Znamená to, že systém v tomto případě nezařadí žádnou rampu a nečeká na dojezd minimální odchylky. Navíc systém rozpočítá zbytky poslední dráhy v taktu bloku rovnoměrně v několika přejezdových taktech, takže pokud není velký rozdíl v tečnách (směrnice pohybu obou bloků), bude přejezd bez změny rychlosti. Větší rozdíl v tečnách navazujících bloků se projeví určitou změnou rychlosti přejezdu, která však je malá a v tomto případě není vůbec důležitá. Obecně možno říci, že čím bude menší rozdíl v tečnách obou navazujících bloků, tím bude přechod plynulejší.

Nutnou podmínkou pro plynulou návaznost je dodržení minimální délky trvání bloku pro plynulou jízdu.

Minimální doby trvání bloku pro plynulou jízdu jsou:

- | | |
|--------------|-------------------------|
| 60 ms | pro systémy řady CNC8x6 |
| 4 ms | pro systémy řady CNC8x9 |

Systém při plynulé jízdě (bez aktivní obálkové rychlosti) kontroluje jen brzdnou dráhu následujícího bloku, to znamená, že v případě potřeby **brzdí maximálně jeden blok předem**. Proto je nutno ještě zabezpečit, aby nebylo odvolání plynulé jízdy ve velmi krátkém bloku, který následuje po plynulých blocích s větší programovanou rychlostí. (Pokud se tento požadavek nedá zabezpečit, je potřeba aktivovat obálkovou rychlost.)

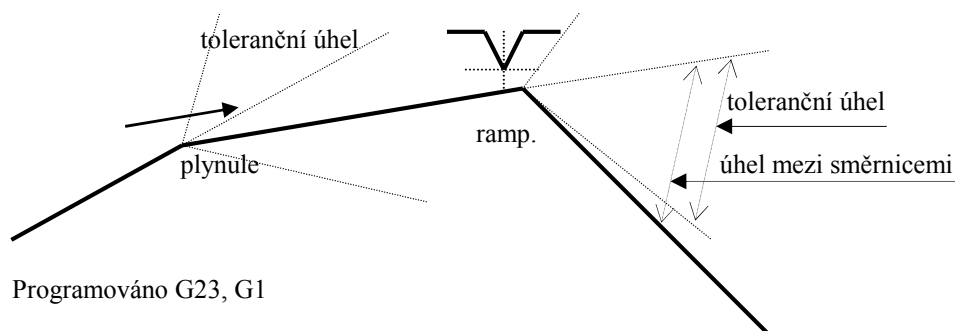
Toleranční úhel pro automatické rozpoznávání

Ve strojní konstantě **R39** je zadán toleranční úhel pro automatické rozpoznávání. Zadává se s přesností na tisíce stupně, takže desetinná čárka je platná a údaj je ve stupních. Systém zjistí prostorový úhel mezi aktuálními směrnicemi pohybu v místě napojení bloků (úhel mezi tečnami pohybu) a porovná jej s tolerančním úhlem zadaným ve strojní konstantě R39. Pokud je úhel mezi směrnicemi pohybu menší než toleranční úhel, může dojít k plynulému napojení bloků. Systém zabrání plynulým přejezdům, když je programován rychloposuv G0.

V době, když systém řídí pohyb pomocí plynulé návaznosti, určuje zrychlení (rampu) strojní konstanta **R236**. Když se systém neřídí plynulou návazností a je programován pracovní posuv, určuje zrychlení strojní konstanta R52. Pro rychloposuv se uplatní strojní konstanta R237.

Nutné podmínky rozpoznávání o zařazení plynulé návaznosti

- Je programována G23 a 1.dekáda strojní konstanty R97 má hodnotu 2.
- Úhel mezi směrnicemi pohybu obou bloků je v toleranci.
- Předchozí i aktuální blok je pohybový a je programována jedna z funkcí G1, G2, G3.
- Délka trvání bloků není menší než minimální doba trvání bloku pro plynulou jízdu.



Příloha A5 – grafická diagnostika plynulé návaznosti

Diagnostika plynulé návaznosti slouží na zjišťování sporných míst v programu z hlediska plynulé jízdy. Pro účely diagnostiky plynulosti jsou v systému implementovány dva grafické formáty. Další popis je platný pro softwarové verze panelu 5.033 a 6.030 a vyšší.

Grafický náhled programu

Aktivace grafického programu se na systému povoluje v souboru **CNC836.KNF** v parametru **\$57**, který musí být nastaven na hodnotu 2 (dva průchody).

Grafický náhled programu se aktivuje tlačítkem **GRAF** na panelu systému (nebo pomocí volby indikace WIN). Po volbě programu v režimu AUT má tlačítko GRAF funkce, které se cyklicky obměňují po 3 stiscích:

Funkce grafického náhledu (tlačítko GRAF):

1.stisk tlačítka GRAF	Systém si zapamatuje aktuální stav formátů v režimu AUT a na pravé části obrazovky vykreslí grafický náhled na vyvolený program. Vykreslení se řídí strojní konstantou R387 , kde se zadává počet bloků pro vykreslení. Pokud je konstanta R327 nulová, vykreslí se celý program (pokud má program například 1000000 bloků, může tak pokreslit celou obrazovku a tvar bude málo čitelný). Když je strojní konstanta R387 nastavena například na hodnotu 50000, vykreslí se jen prvních 50000 bloků programu. Při stisku tlačítka ENTER se obrazovka smaže a vykreslí se dalších 50000 bloků.	
2.stisk tlačítka GRAF	Zobrazí se zvláštní MENU, které patří pro grafický náhled programů. Pokud funkce které nabízí toto menu nepotřebujeme využít, pokračujeme na 3. stisk tlačítka GRAF.	
	LUPA	Po stisku se zobrazí naváděcí kříž, který pomocí kurzorových tlačítek navedeme na místo v grafu, které potřebujeme zvětšit. Tlačítkem ENTER se postupně přepíná režim zvětšování (ZOOM) a režim posouvání (SHIFT). Pro řízení zvětšování a posouvání slouží kurzorová tlačítka. Po 1.stlačení tlačítka LUPA se na grafickém průběhu zobrazí žluté a červené tečky na přechodu bloků. Žlutá tečka znamená plynulé přejetí bloků. Červená tečka symbolizuje porušení plynulé jízdy bloků.
	KOREKCE	Zobrazí dráhu s poloměrovou korekcí (ekvidistantu) modrou barvou.
	PLYNULOST	V blocích, kde je plynulost porušena, (červené tečky na přelomu bloků), se zobrazí odpovídající čísla bloků.
	ROVINA	Volba roviny zobrazení X-Y, X-Z, Y-Z (musí být odpovídajícím způsobem nastaveny strojní konstanty R340 a R341 pro korekční rovinu)
3.stisk tlačítka GRAF	Obnoví se původní stav formátů a původní MENU před 1. stiskem tlačítka GRAF.	

Pro sledování skutečného průběhu rychlosti možno využít formát **sledování průběhu rychlosti**. Pomocí volby indikace WIN se zvolí formát sledování průběhu rychlosti a zvolí se sledování bloků N (místo souřadnic X,Z,...). Po zkalibrování můžeme sledovat průběh výsledné prostorové rychlosti v reálném čase a pomocí bílých svislých čar jsou znázorněny jednotlivé bloky.

Příloha A6 - obálková rychlost s lineárním průběhem rychlosti

Systémy řady CNC8x9 (DUAL) od softwarové verze 40.10 panelu a 6.030 sekundárního procesoru, umožňují pro řízení plynulosti použít tzv. „**obálkovou rychlost**“. Pod pojmem „obálková rychlost“ rozumíme dynamické řízení rychlosti s analýzou příštích bloků, kdy systém jede **plynule** a přitom se snaží dosáhnout programovanou rychlost i v průběhu více bloků. Tím je umožněno, že i bloky o malé délce se jedou mnohonásobně větší rychlostí, než by se jelo bez řízení obálkové rychlosti. Z toho vyplývá, že systém musí předvídat a korigovat rychlost na základě určitých kritérií, které vyplynou v příštích blocích. Analýza kritérií z příštích bloků probíhá max. 500 bloků dopředu.

Obálková rychlost umožní na stroji používat **vysokorychlostní obrábění** i když je program sestaven z velkého množství malých bloků. Při vyšších rychlostech obrábění je potřeba kompenzovat regulační odchylku polohové vazby pro dosažení požadované přesnosti obrábění. Kompenzace regulační odchylky až na nulovou hodnotu a to i při dynamických stavech stroje umožní vhodné nastavení parametrů servosmyčky pro **feedforward** s derivací (viz Návod k programování PLC: Nastavování parametrů servopohonů, nebo strojní konstanty R356 - R381)

Příloha A6.1 – dynamické lineární řízení rychlosti s analýzou příštích bloků

Nutnou podmínkou je nastavení na dvouprůchodový záznam programu do paměti (v CNC836.KNF se nastaví \$57 na hodnotu 2) a 2.dekády strojní konstanty **R338** na hodnotu 1.

Když se systém neřídí plynulou návazností a je programován pracovní posuv, určuje zrychlení (rampu) strojní konstanta **R52**. Pro rychloposuv se uplatní strojní konstanta **R237**.

V době, když systém řídí pohyb pomocí plynulé návaznosti, určuje zrychlení (rampu) strojní konstanta **R236**. Pokud je v konstantě R236 nulová hodnota, převezme se zrychlení pro plynulou návaznost z konstanty R52.

Přehled kritérií pro řízení rychlosti :

- **Změna zadané rychlosti.** Omezení rychlosti s dostatečným předstihem, pokud je v příštích blocích programovaná menší rychlost.
- **Konec plynulosti.** Konec úseku plynulého jetí, kdy prostorový úhel návaznosti dvou po sobě jdoucích bloků je větší než povolený limit. Například systém s předstihem brzdí při přechodu na jinou vrstevnici při prostorovém obrábění.
- **Kritéria pro kruhovou interpolaci.** Kritérium odstředivého zrychlení a přesnosti generace dráhy pro kruhovou interpolaci. Systém s předstihem omezí rychlost tak, aby byly dodrženy zadané parametry pro kruhovou interpolaci.
- **Dynamické kritérium.** Princip je založen na předpokladu, že soustava pohonů s celou mechanikou stroje má určité maximální povolené přetížení. Na základě programované rychlosti, úhlu návaznosti bloků a délky bloků se rychlost koriguje s dostatečným předstihem tak, aby nebylo toto maximální povolené přetížení překročeno.
- **Kritérium přesnosti.** Systém na základě programované rychlosti a úhlu návaznosti bloků s dostatečným předstihem omezuje rychlost tak, aby byla odchylka od ideální dráhy menší než zadaný limit.
- **Minimální doba trvání bloku.** Systém kontroluje skutečnou dobu trvání bloku a v případě potřeby s dostatečným předstihem omezí rychlost tak, aby doba trvání bloku byla větší nebo rovna minimální době trvání bloku pro plynulou jízdu (4 ms).

Příloha A6.2 - přehled parametrů pro aktivaci a činnost obálkové rychlosti

Parametr	Příklad	Popis
CNC836.KNF: \$57 = 2	2	Dva průchody záznamu programu. Povolení dvouprůchodového záznamu programu do paměti při volbě programu. Druhý průchod záznamu umožní analýzu kritérií pro plynulou jízdu (kromě toho i grafické vykreslení dráhy s poloměrovou korekcí).
2. dekáda R338 = 1	1	Obálková rychlost s lineárním průběhem rychlosti. Předvídání pro korekci rychlosti max.500 bloků předem.
R39	60.000	Toleranční úhel pro plynulou návaznost. Konstanta udává úhel tečen v koncovém bodu předchozího bloku a aktuálního bloku. Pokud je programována funkce G23, je pracovní posuv, a úhel tečen je menší než zadaný limit, považuje se dráha za plynulou a může dojít jen ke korekcím rychlosti na základě dalších kritérií. Pokud je úhel tečen větší než zadaný limit, dojde k zpomalení na konci předchozího bloku až na nulovou rychlost (zkontrolování minimální odchylky) a k opětovnému rozjezdu v následujícím bloku. Velikost úhlu v R39 se doporučuje nastavit na větší hodnotu (například 60 stupňů), protože další kritéria korekce rychlosti mohou odpovídajícím způsobem přizpůsobit rychlost aktuálnímu úhlu návaznosti bloků, a celkový pohyb je pak plynulejší.
1. dekáda R97 = 2	2	Automatické rozpoznání plynulé návaznosti. Zařadí se pomocí funkce G23, kdy systém sám testuje prostorový úhel mezi směrnicemi v místě napojení bloků, porovnává jej s tolerančním úhlem zadaným ve strojní konstantě R39 a rozhoduje o plynulosti.
R236	400	Zrychlení pro plynulou návaznost. Konstanta R236 určuje zrychlení (rampu) v době, kdy systém řídí pohyb pomocí plynulé návaznosti. Když je v konstantě R236 nulová hodnota, zrychlení pro plynulou návaznost se převezme ze strojní konstanty R52. (Doporučuje se nastavit menší hodnotu alespoň o 100, než je nastavena v konstantě R52.)
1. až 4. dekáda R232	0000	Geometrické kritérium pro kruhovou interpolaci. Systém omezuje rychlost při kruhové interpolaci tak, aby odchylka od ideální kruhovitosti byla menší než limit. (Viz Příloha F – strojní konstanty systému CNC8x6 CNC8x9.)
5. až 8. dekáda R232	0040	Dynamické kritérium pro kruhovou interpolaci. Systém omezuje rychlost při kruhové interpolaci tak, aby nebylo překročeno maximální povolené přetížení stroje. (Viz Příloha F – strojní konstanty systému CNC8x6 CNC8x9.)
R382	0	Délka bloku pro zrušení plynulé návaznosti. Parametr R382 určuje limit délky bloku při plynulé návaznosti bloků G23. Když je délka bloku menší než zadaný limit, je na tomto místě plynulá návaznost přerušena. Znaménko mínus aktivaci limitu blokuje. Parametr se může uplatnit pro obálkovou rychlost, kdy zrušení plynulé jízdy pro velmi malé bloky může mít optimálnější časový průběh.

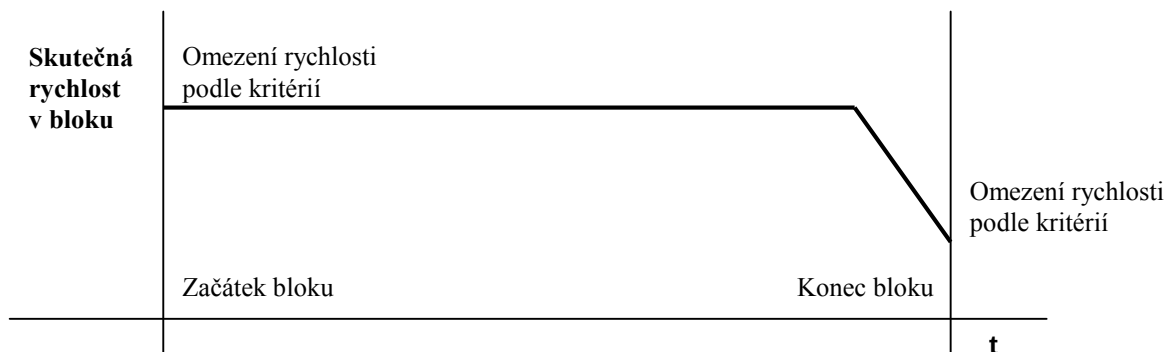
R383	0.005	Délka bloku při plynulé návaznosti pro vynechání bloku. Když je délka bloku menší než zadaný limit, je tento blok přidružen k sousednímu bloku. Znaménko mínus aktivaci limitu blokuje. Parametr se může uplatnit pro obálkovou rychlost, kdy vynechání velmi malých bloků zbytečně nezpomaluje obrábění. Velmi malé bloky (několik mikrometrů) mohou být generovány některými návrhovými programy CAD. Přidružení bloků je povoleno jen pro lineární pohyb a jen tehdy, když v bloku nejsou programovány žádné technologické funkce. Nesmí být programována změna rychlosti a také nesmí být programována funkce G91.
1. až 4. dekáda R384	0.008	Omezení rychlosti – kritérium přesnosti. Systém omezuje rychlost při plynulé návaznosti a při aktivní obálkové rychlosti tak, aby byla odchylka od ideální dráhy menší než zadaný limit v parametru R384. Hodnota se zadává v mikrometrech. Maximální rychlost z hlediska požadované přesnosti je závislá na úhlu návaznosti sousedních bloků. (Viz dále.)
5. až 6. dekáda R384	20	Procento poklesu rychlosti pro grafický náhled. Grafický náhled programu je možno využít pro diagnostiku plynulosti. Po 1.stlačení tlačítka LUPA se na grafickém průběhu zobrazí žluté a červené tečky na přechodu bloků. Žlutá tečka znamená plynulé přjetí bloků. Červená tečka symbolizuje pokles rychlosti pod procento z programované rychlosti, které se zadává v 5. až 6. dekádě R384. Například pro hodnotu 20 se na grafice objeví červené tečky při poklesu rychlosti pod 20 procent z programované rychlosti.
1. až 7. dekáda R385	10.000	Omezení rychlosti – dynamické kritérium. Systém omezuje rychlost při plynulé návaznosti a při aktivní obálkové rychlosti tak, aby nebylo překročeno povolené přetížení stroje zadáno v parametru R385. Hodnota se zadává v mm/s^2. Maximální rychlost z hlediska požadovaného přetížení je závislá na úhlu návaznosti sousedních bloků. Nastavení parametru úzce souvisí s dynamikou pohonů. Princip je založen na předpokladu, že soustava pohonů s celou mechanikou stroje má určité maximální povolené přetížení a systém omezí rychlost tak, aby nebylo překročeno. (Viz dále.)
8. dekáda R385	1	Korekce dynamického kritéria na délku bloku. V osmé dekádě parametru R385 se může aktivovat korekce kritéria přetížení na délku bloku, přitom pro správnou funkci je nutno zadat hodnotu podle použité délky bloků při nastavování dynamického kritéria (referenční délka): 0 Korekce dynamického kritéria není aktivována 1 Referenční délka bloku je 1 mm 2 Referenční délka bloku je 0.5 mm 3 Referenční délka bloku je 0.1 mm 4 Referenční délka bloku je 0.05 mm

R386	10.000	Délka bloku pro nárůst rychlosti. Pokud je délka bloku menší než zadaný limit ve strojní konstantě R386, systém zachová v rámci bloku rychlosti vypočtené z dynamického kritéria a kritéria přesnosti na začátku a na konci bloku. Systém zachová přechodové rychlosti v celém bloku. Pro novější verze software (6.211,...) se doporučuje konstantu R386 nepoužívat a místo ní použít konstantu pro kritérium času pro nárůst rychlosti R395.
R387	50.000	Počet bloků pro grafický náhled. Vykreslení grafického náhledu se řídí strojní konstantou R387, kde se zadává počet bloků pro vykreslení. Pokud je konstanta R327 nulová, vykreslí se celý program (pokud má program například 1000000 bloků, může tak pokreslit celou obrazovku a tvar bude málo čitelný). Když je strojní konstanta R387 nastavena například na hodnotu 50000, vykreslí se jen prvních 50000 bloků programu. Při stisku tlačítka ENTER se obrazovka smaže a vykreslí se dalších 50000 bloků.
R395	0.500	Kritérium času pro nárůst rychlosti Pokud je čas výkonu bloku menší než zadaný limit ve strojní konstantě R395, systém zachová v rámci bloku rychlosti vypočtené z dynamického kritéria a kritéria přesnosti na začátku a na konci bloku. Čas se zadává v milisekundách. Pokud je čas výkonu bloku větší než zadaný limit v R395, je povolen nárůst rychlosti maximálně na programovanou rychlost a opětovný sestup rychlosti v rámci jednoho bloku. Systém proto zachová přechodové rychlosti vypočtené podle dynamického kritéria a kritéria přesnosti jen na začátku a na konci bloku. (Hodnota nula v R395 je základní nastavení na čas 0,5s.) Doporučujeme konstantu R386 pro délku bloku nepoužívat a nastavit ji například na hodnotu 100.000.

Příloha A6.3 – kritérium času pro nárůst rychlosti, parametr R395.

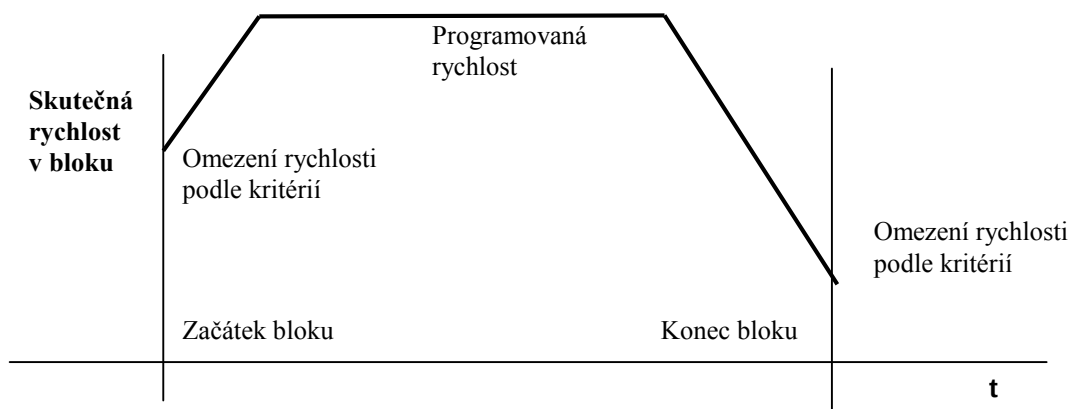
Pokud je čas výkonu bloku menší než zadaný limit ve strojní konstantě R395, systém zachová v rámci bloku rychlosti vypočtené z dynamického kritéria a kritéria přesnosti na začátku a na konci bloku. To může být nevýhodné v případě delších bloků, které jsou spojeny větším úhlem návaznosti, protože dodržení přechodových rychlostí podle nastavených kritérií bude platit pro celý blok. Delší bloky potom se projedou zbytečně pomalu.

Příklad zachování rychlosti v rámci bloku pro čas jetí bloku menší než limit v R395:



Pokud je čas výkonu bloku větší než zadaný limit v R395, je povolen nárůst rychlosti maximálně na programovanou rychlost a opětovný sestup rychlosti v rámci jednoho bloku. Systém proto zachová přechodové rychlosti vypočtené podle dynamického kritéria a kritéria přesnosti jen na začátku a na konci bloku. Vhodným nastavením konstanty R395 může dojít k podstatnému urychlení jetí programu.

Příklad pro povolený nárůst rychlosti v rámci bloku, když je čas jetí délku bloku větší než limit v R395:



Příloha A6.4 - kritérium přesnosti, parametr R384

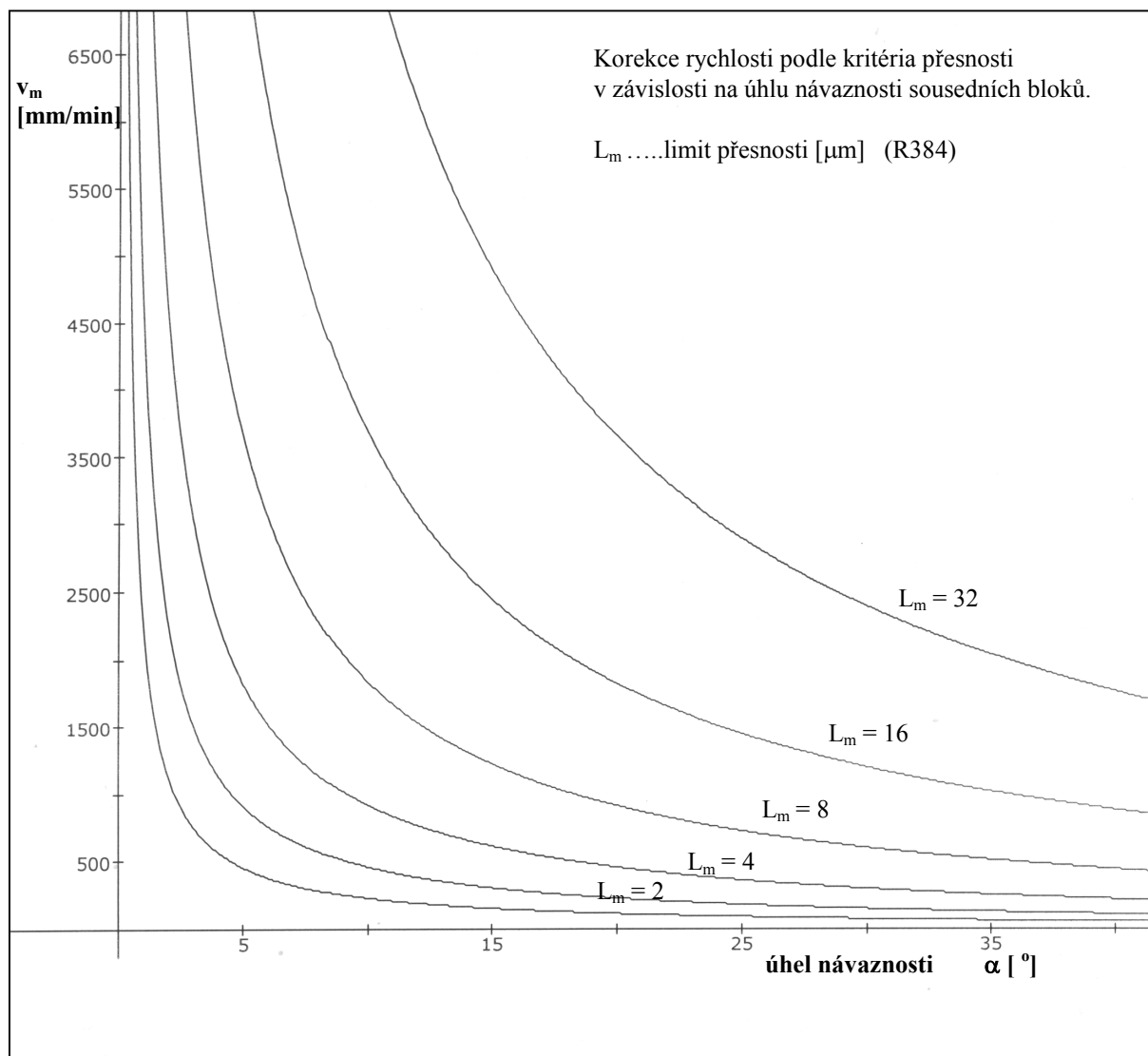
Systém omezuje s dostatečným předstihem rychlost při plynulé návaznosti a při aktivní obálkové rychlosti tak, aby byla odchylka od ideální dráhy menší než zadaný limit v parametru **R384**. Hodnota se zadává v mikrometrech.

Maximální povolená rychlost z hlediska požadované přesnosti je závislá na úhlu návaznosti sousedních bloků podle vztahu:

$$v_m = \frac{L_m}{T_s \cdot \tan(\alpha / 2)}$$

v_m	maximální povolená rychlost podle kritéria přesnosti
L_m	limit přesnosti zadaný ve strojní konstantě R384
T_s	takt interpolace
α	aktuální úhel návaznosti sousedních bloků

Parametr R384 je vhodné nastavovat podle typu obrábění (hrubování, přesné obrábění). Když se zadá hodnota nula, kritérium přesnosti není aktivováno. Pokud kritérium přesnosti není aktivováno nebo je v parametru R384 nastavena velká hodnota pro limit (50 mikronů), ve většině případů bude omezovat rychlost dynamické kritérium podle parametru R385, které je pro daný stroj pevně nastaveno a nemá se měnit.



Příloha A6.5 - dynamické kritérium, parametr R385

Systém s dostatečným předstihem omezuje rychlost při plynulé návaznosti a při aktivní obálkové rychlosti tak, aby nebylo překročeno povolené přetížení stroje zadáno v 1. až 7. dekádě parametru **R385**. Hodnota se zadává v mm/s^2 .

Nastavení parametru úzce souvisí s dynamikou pohonů a s nastavenou hodnotou zesílení (K_v) u servosmyček. Princip je založen na předpokladu, že soustava pohonů s celou mechanikou stroje má určité maximální povolené přetížení (zrychlení) a systém omezí rychlost tak, aby nebylo překročeno.

Maximální rychlost z hlediska požadovaného přetížení je závislá na délce bloků a úhlu návaznosti sousedních bloků podle vzorce:

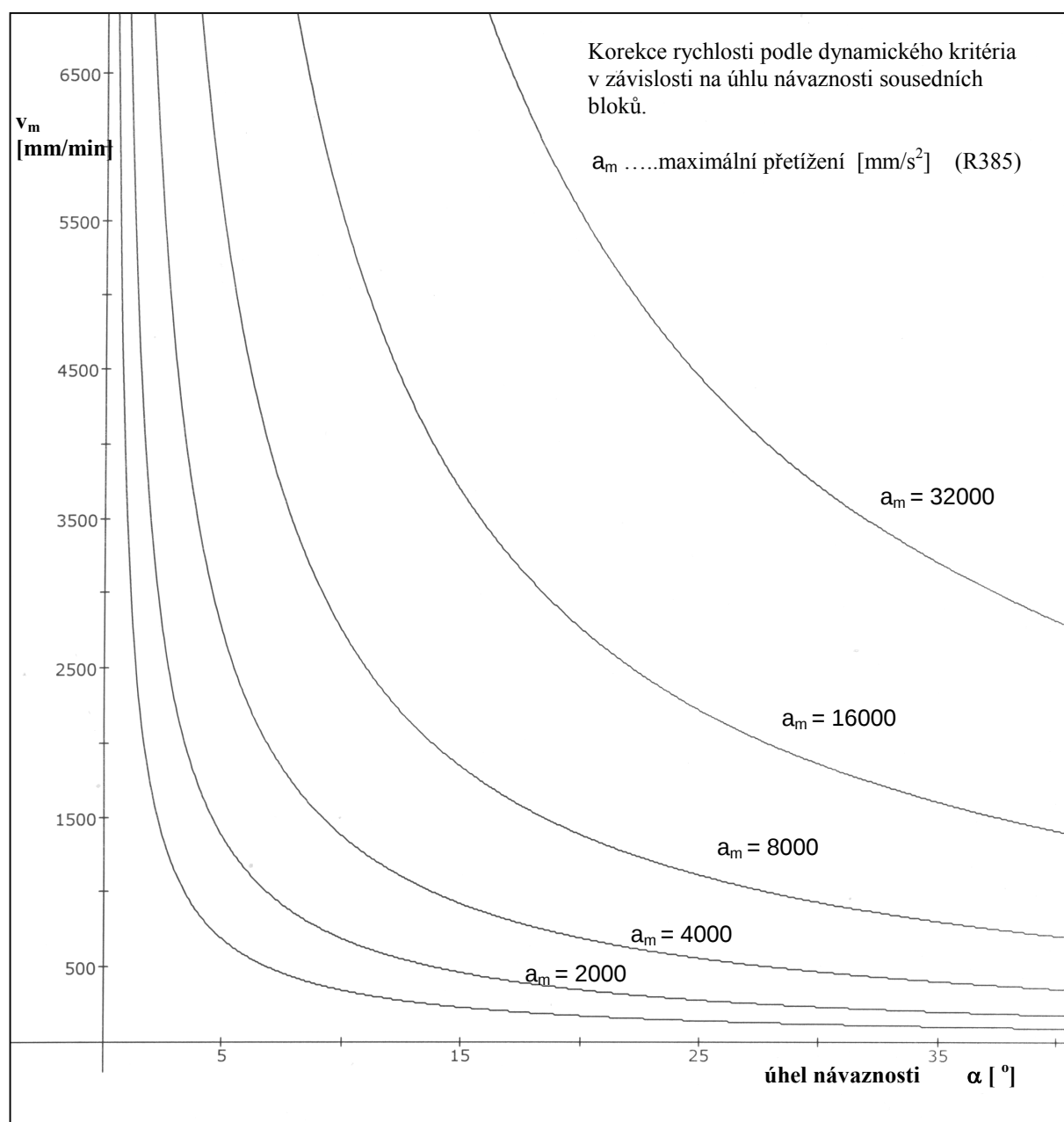
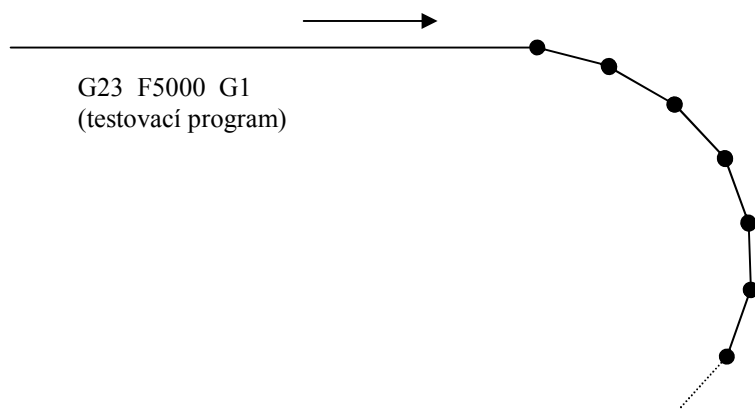
$$v_m = \frac{a_m \cdot T_s}{2 \cdot \sin(\alpha / 2)}$$

v_m	maximální povolená rychlost podle dynamického kritéria
a_m	maximální povolené zrychlení zadané ve strojní konstantě R385
T_s	takt interpolace
α	aktuální úhel návaznosti sousedních bloků

Dynamické kritérium je vlastně parametr pro konkrétní stroj, a proto by se po nastavení neměl měnit. Při nastavování je vhodné sestavit testovací program, který je složen z malých bloků s lineárním pohybem o stejné délce. Délka bloků by měla odpovídat přibližně minimální délce bloků, které generuje vyšší návrhový systém CAD. Doporučuje se použít délku bloku z řady **1mm, 0.5mm, 0.1mm** nebo **0.05mm**, protože potom bude možné aktivovat korekci dynamického kritéria na délku bloku. Bloky by měly mezi sebou svírat vždy stejný úhel, například 10 stupňů. Celý testovací program by tak způsobil prudké otočení pohybu o 180 stupňů. Program musí obsahovat funkci G23 a rychlost, která bude typická pro obrábění na konkrétním stroji. Při projíždění testovacího programu postupně nastavujeme dynamické kritérium, například pomocí sledování odchylky polohy ve volbě indikace.

V osmé dekádě parametru R385 se může aktivovat korekce kritéria přetížení na délku bloku, přitom pro správnou funkci je nutno zadat hodnotu podle použité délky bloků při nastavování dynamického kritéria (referenční délka):

8. dekáda R385	0	Korekce dynamického kritéria není aktivována.
	1	Referenční délka bloků je 1 mm.
	2	Referenční délka bloků je 0.5 mm.
	3	Referenční délka bloků je 0.1 mm.
	4	Referenční délka bloků je 0.05 mm.



Příloha A7- obálková rychlost s parabolickým průběhem rychlosti

Systémy řady CNC8x9 (DUAL) od softwarové verze 40.28 panelu a 6.312 sekundárního procesoru, umožňují pro řízení plynulosti použít tzv. „**obálkovou rychlost s parabolickým průběhem rychlosti**“. Jedná se o dynamické řízení rychlosti s analýzou příštích bloků, kdy systém jede plynule a přitom se snaží dosáhnout programovanou rychlost i v průběhu více bloků s parabolickým časovým průběhem. Tím je umožněno, že i bloky o malé délce se jedou mnohonásobně větší rychlostí, než by se jelo bez řízení obálkové rychlosti. Z toho vyplývá, že systém musí předvídat a korigovat rychlost na základě určitých kritérií, které vyplynou v příštích blocích. Analýza kritérií z příštích bloků probíhá max. 500 bloků dopředu. Všechny změny rychlosti, i v průběhu plynulého jetí, mají parabolický časový průběh.

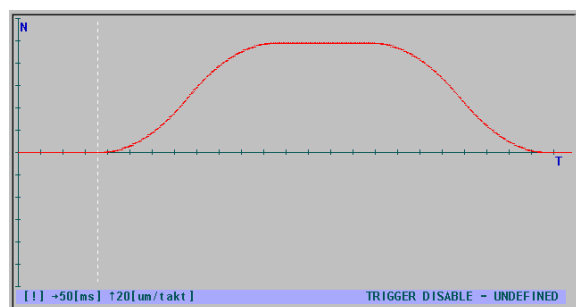
Obálková rychlost umožní na stroji používat **vysokorychlostní obrábění** i když je program sestaven z velkého množství malých bloků. Při vyšších rychlostech obrábění je potřeba kompenzovat regulační odchylku polohové vazby pro dosažení požadované přesnosti obrábění. Pro kompenzaci odchylky se používá regulační obvod s feedforwardem. Derivační složka feedforwardu má kompenzovat přejezdy i v dynamických stavech stroje. (viz Návod k programování PLC: Nastavování parametrů servopohonů nebo strojní konstanty R356 - R381)

Příloha A7.1 - parabolický průběh řízení rychlosti

Omezení nárůstu zrychlení (ryv) na lineární průběh, které se projeví parabolickým průběhem rychlosti. Impuls hnací síly v čase je mnohem menší než u lineárního průběhu rychlosti. Tím je menší i reakční síla působící přes pohon na rám stroje, což má za následek výrazné potlačení rezonančních kmitů stroje.

Pro nastavení tvaru časového průběhu rychlosti je potřeba zadat dva parametry:

- **Lineární rampa (A)** Popisuje úsek řízení, kdy se rychlost mění lineárně a zrychlení je konstantní
- **Parabolická rampa (P)** Popisuje úsek řízení, kdy se rychlost mění kvadraticky a zrychlení je lineární



Pro menší změny rychlosti se lineární průběh neuplatní, protože přímo navazuje jeden parabolický průběh na inverzní parabolický průběh. Při větších změnách rychlosti se mění parabolický průběh na lineární průběh rychlosti. Strmost lineárního průběhu je proto vlastně maximální povolenou strmostí nárůstu rychlosti. Správně nastavené parametry pro lineární a parabolickou rampu (A, P) je potřeba zkontrolovat na osciloskopu systému.

Čas potřebný pro dosažení rychlosti **F** je:

$$t = 1.478 \sqrt{\frac{F}{p}}$$

T čas pro dosažení rychlosti [s]
 F rychlost [mm/min]
 P parabolická rampa (dle stroj.konstanty (10000))

Podmínka pro nastavení parametru lineární rampy **A** pro uplatnění lineární rampy:

$$a < \sqrt{\frac{F_{\max} p}{1966}}$$

A lineární rampa [mm/s²]
 F_{max} maximální rychlost [mm/min]
 P parabolická rampa (dle stroj.konstanty (10000))

Příloha A7.2 - dynamické parabolické řízení rychlosti s analýzou příštích bloků

Nutnou podmínkou je nastavení na dvouprůchodový záznam programu do paměti (v CNC836.KNF se nastaví \$57 na hodnotu 2) a 2.dekády strojní konstanty **R338** na hodnotu 2.

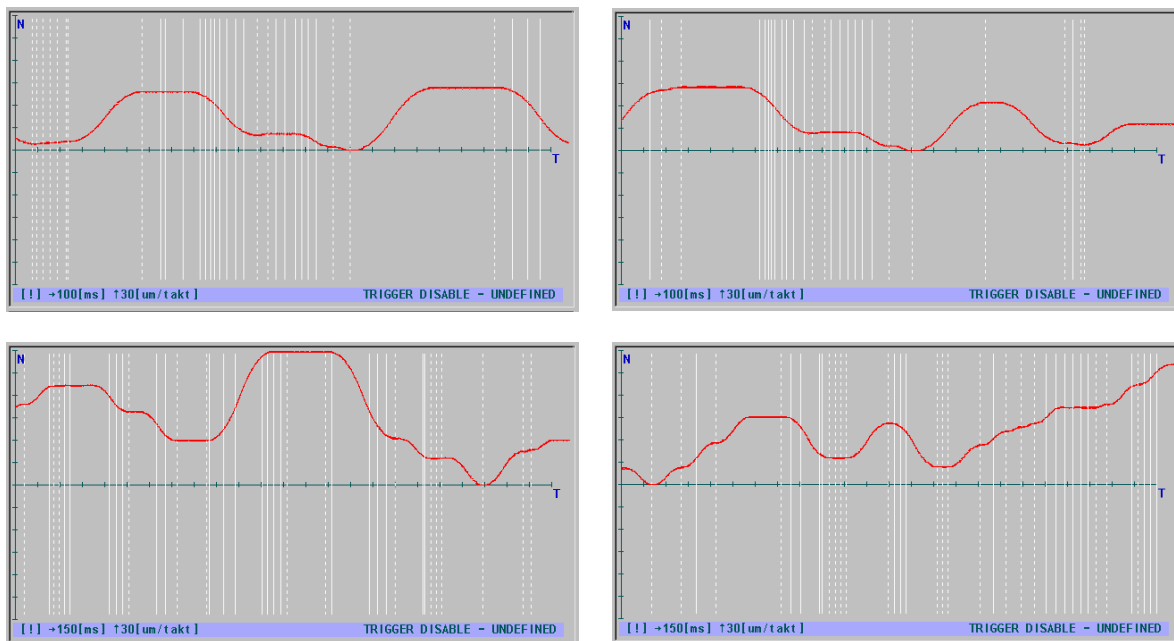
Systém analyzuje příští bloky partprogramu a na základě kritérií optimalizuje sjednocení bloků do větších částí (úseků), ve kterých určí časové funkce rychlosti s parabolickým průběhem. Tím je dosažen pokud možno optimální časový průběh rychlosti.

Pro nastavení všech hodnot pro řízení rychlosti a všech kritérií slouží čtyři sady parametrů pro řízení dynamiky (viz. dále). Další sada parametrů se uplatní pro rychloposuv a pro stop pohybu.

Přehled kritérií pro řízení rychlosti :

- **Změna zadané rychlosti.** Omezení rychlosti s dostatečným předstihem, pokud je v příštích blocích programovaná menší rychlost.
- **Konec plynulosti.** Konec úseku plynulého jetí, kdy prostorový úhel návaznosti dvou po sobě jdoucích bloků je větší než povolený limit. Například systém s předstihem brzdí při přechodu na jinou vrstevnici při prostorovém obrábění.
- **Kritéria pro kruhovou interpolaci.** Kritérium odstředivého zrychlení a přesnosti generace dráhy pro kruhovou interpolaci. Systém s předstihem omezí rychlost tak, aby byly dodrženy zadané parametry pro kruhovou interpolaci.
- **Dynamické kritérium.** Princip je založen na předpokladu, že soustava pohonů s celou mechanikou stroje má určité maximální povolené přetížení. Na základě programované rychlosti, úhlu návaznosti bloků a délky bloků se rychlost koriguje s dostatečným předstihem tak, aby nebylo toto maximální povolené přetížení překročeno.
- **Kritérium přesnosti.** Systém na základě programované rychlosti a úhlu návaznosti bloků s dostatečným předstihem omezuje rychlost tak, aby byla odchylka od ideální dráhy menší než zadaný limit.
- **Minimální doba trvání bloku.** Systém kontroluje skutečnou dobu trvání bloku a v případě potřeby s dostatečným předstihem omezí rychlost tak, aby doba trvání bloku byla větší nebo rovna minimální době trvání bloku pro plynulou jízdu (4 ms).

Příklady časových průběhů výsledné prostorové rychlosti obrábění (bílé čáry znázorňují předěly bloků):



Příloha A7.3- přehled parametrů pro aktivaci a činnost obávkové rychlosti

Parametr	Příklad	Popis
CNC836.KNF: \$57 = 2	2	Dva průchody záznamu programu. Povolení dvouprůchodového záznamu programu do paměti při volbě programu. Druhý průchod záznamu umožní analýzu kritérií pro plynulou jízdu (kromě toho i grafické vykreslení dráhy s poloměrovou korekcí).
2. dekáda R338 = 2	2	Obávková rychlost s parabolickým průběhem rychlosti. Předvídání pro korekci rychlosti max.500 bloků předem.
R39	60.000	Toleranční úhel pro plynulou návaznost. Konstanta udává úhel tečen v koncovém bodu předchozího bloku a aktuálního bloku. Pokud je programována funkce G23, je pracovní posuv, a úhel tečen je menší než zadaný limit, považuje se dráha za plynulou a může dojít jen ke korekcím rychlosti na základě dalších kritérií. Pokud je úhel tečen větší než zadaný limit, dojde k zpomalení na konci předchozího bloku až na nulovou rychlost (zkontrolování minimální odchylky) a k opětovnému rozjezdu v následujícím bloku. Velikost úhlu v R39 se doporučuje nastavit na větší hodnotu (například 60 stupňů), protože další kritéria korekce rychlosti mohou odpovídajícím způsobem přizpůsobit rychlost aktuálnímu úhlu návaznosti bloků, a celkový pohyb je pak plynulejší.

1. dekáda R97 = 2	2	Automatické rozpoznání plynulé návaznosti. Zařadí se pomocí funkce G23, kdy systém sám testuje prostorový úhel mezi směrnicemi v místě napojení bloků, porovnává jej s tolerančním úhlem zadaným ve strojní konstantě R39 a rozhoduje o plynulosti.
R383	0.005	Délka bloku při plynulé návaznosti pro vynechání bloku. Když je délka bloku menší než zadaný limit, je tento blok přidružen k sousednímu bloku. Znaménko minus aktivaci limitu blokuje. Parametr se může uplatnit pro obálkovou rychlost, kdy vynechání velmi malých bloků zbytečně nezpomaluje obrábění. Velmi malé bloky (několik mikrometrů) mohou být generovány některými návrhovými programy CAD. Přidružení bloků je povoleno jen pro lineární pohyb a jen tehdy, když v bloku nejsou programovány žádné technologické funkce. Nesmí být programována změna rychlosti a také nesmí být programována funkce G91.
R387	50.000	Počet bloků pro grafický náhled. Vykreslení grafického náhledu se řídí strojní konstantou R387, kde se zadává počet bloků pro vykreslení. Pokud je konstanta R327 nulová, vykreslí se celý program (pokud má program například 1000000 bloků, může tak pokreslit celou obrazovku a tvar bude málo čitelný). Když je strojní konstanta R387 nastavena například na hodnotu 50000, vykreslí se jen prvních 50000 bloků programu. Při stisku tlačítka ENTER se obrazovka smaže a vykreslí se dalších 50000 bloků.
R395	0.500	Kritérium času pro nárůst rychlosti Kritérium času slouží pro test, zda systém zachová v rámci bloku (úseku) rychlosti vypočtené z dynamického kritéria a kritéria přesnosti na začátku a na konci bloku nebo je povolen nárůst rychlosti maximálně na programovanou rychlost a opětovný sestup rychlosti v rámci jednoho bloku. Systém si předem připraví oba případy (s nárůstem rychlosti a bez nárůstu) a vypočte rozdíl dob trvání obou případů. Pokud je rozdíl dob trvání menší než zadaný limit v konstantě R395, neoplatí se v rámci bloku zrychlovat, systém proto zachová rychlosti v rámci bloku vypočtené z kritérií. Pokud je rozdíl dob výkonu bloků větší než zadaný limit v R395, je povolen nárůst rychlosti maximálně na programovanou rychlost a opětovný sestup rychlosti v rámci jednoho bloku. Systém proto zachová přechodové rychlosti vypočtené podle dynamického kritéria a kritéria přesnosti jen na začátku a na konci bloku. Čas se zadává v milisekundách.
R406 až R411		1. sada pro dynamické řízení rychlosti (popis jednotlivých sad následuje dále)
R412 až R417		2. sada pro dynamické řízení rychlosti (popis jednotlivých sad následuje dále)
R418 až R423		3. sada pro dynamické řízení rychlosti (popis jednotlivých sad následuje dále)
R424 až R429		4. sada pro dynamické řízení rychlosti (popis jednotlivých sad následuje dále)
R430 až R432		Sada pro dynamické řízení rychlosti pro rychloposuv (popis jednotlivých sad následuje dále)

R406 až R411 (1. sada pro dynamické řízení rychlosti)

Implicitní sada parametrů, která je platná po zapnutí stroje, nebo se uplatní naprogramováním funkce: **G23 II**

První strojní konstanta sady je řídícím slovem, přitom 1.dekáda povoluje celou sadu a další dekády určují zda u následujících položek se jedná o přímé zadání hodnot nebo o nepřímé prostřednictvím parametrů. Další dvě konstanty určují parabolické a lineární zrychlení a další tři konstanty určují kritéria přesnosti, přetížení a odstředivé zrychlení.

	dekáda	hodnota	popis
R406	1. dekáda	0	Sady pro dynamické řízení rychlosti nejsou použity. Pro nastavení dynamiky průběhu rychlosti platí strojní konstanty R232, R236, R384 a R385
		1	Sady pro dynamické řízení rychlosti jsou použity. Pro nastavení dynamiky průběhu rychlosti platí strojní konstanty R406 až R457. Konstanty R232, R236, R384 a R385 jsou neúčinné.
	2. dekáda	0	Hodnota parabolického zrychlení P je přímo zadána v konstantě R407 .
		1	Hodnota parabolického zrychlení P je v parametru, přitom konstanta R407 obsahuje odkaz na použitý parametr (00-94).
	3. dekáda	0	Hodnota lineárního zrychlení A je přímo zadána v konstantě R408 .
		1	Hodnota lineárního zrychlení A je v parametru, přitom konstanta R408 obsahuje odkaz na použitý parametr (00-94).
	4. dekáda	0	Hodnota pro kritérium přesnosti je přímo zadána v konstantě R409 .
		1	Hodnota pro kritérium přesnosti je zadána v parametru, přitom konstanta R409 obsahuje odkaz na použitý parametr (00-94).
	5. dekáda	0	Hodnota pro kritérium přetížení je přímo zadána v konstantě R410 .
		1	Hodnota pro kritérium přetížení je zadána v parametru, přitom konstanta R410 obsahuje odkaz na použitý parametr (00-94).
	6. dekáda	0	Hodnota pro odstředivé zrychlení na kruhu je přímo zadána v konstantě R411 .
		1	Hodnota pro odstředivé zrychlení na kruhu je zadána v parametru, přitom konstanta R411 obsahuje odkaz na použitý parametr (00-94).
R407	1. až 7. dekáda	Např. 50.000	Parabolické zrychlení (P) (pokud není parametrické zadání). Hodnotu je nutno nastavit pomocí osciloskopu pro daný typ stroje. Pro běžné stroje se hodnota pohybuje v rozsahu 10.000 (povolná, malé stoupání paraboly) až 150.000 (strmé, velké stoupání paraboly).
	8. dekáda	0	Základní rozsah pro zadání parabolického zrychlení
		2	Rozsah zadání pro parabolické zrychlení je 100x menší. Používá se pro zadání velmi malých hodnot stoupání parabol. (Například systém pro vodní paprsek).
		3	Rozsah zadání pro parabolické zrychlení je 1000x menší (od 6.314).
R408	1. až 8. dekáda	Např. 0.600	Lineární zrychlení (A) (pokud není parametrické zadání). Význam konstanty je stejný jako u konstanty R236. Když stoupání paraboly dosáhne hodnoty zadané v této konstantě, plynule se parabolický průběh rychlosti změní na lineární. Jedná se o maximální změnu rychlosti, která bude pro danou sadu povolena. Pokud bude zadána příliš velká hodnota vzhledem k parabolickému zrychlení a vzhledem k maximální rychlosti, budou navazovat na sebe jen paraboly a lineární zrychlení se neuplatní.
R409	1. až 4. dekáda	Např. 0.008	Kritérium přesnosti (pokud není parametrické zadání). Význam je stejný jako u strojní konstanty R384 (popsané v popisu parametrů pro lineární obálkovou rychlost), která určuje omezení rychlosti na základě kritéria přesnosti vypočteného podle úhlu návaznosti sousedních bloků.
	5. až 6. dekáda	Např. 20	V 5. až 6. dekadě se zadává procento poklesu rychlosti pro grafický náhled.

R410	1. až 7. dekáda	Např. 10.000	Kritérium přetížení (pokud není parametrické zadání). Význam je stejný jako u strojní konstanty R385 (popsané v popisu parametrů pro lineární obálkovou rychlost), která určuje omezení rychlosti na základě kritéria přetížení vypočteného podle úhlu návaznosti sousedních bloků.
	8. dekáda	1,2,3,4	Korekce kritéria přetížení na aktuální délku bloku, podobně jako u R385.
R411	1. až 7. dekáda	Např. 200.000	Kritérium odstředivého zrychlení pro kruhovou interpolaci (pokud není parametrické zadání). Starší způsob zadání v procentech úměrnosti k odmocnině poloměru, když 8. dekáda konstanty je nulová. Význam zadání je stejný jako u konstanty R232, horní čtyři dekády (dynamické kritérium)
		Např. 4.000	Novější způsob zadání pomocí odstředivého zrychlení, když 8. dekáda konstanty je nastavena na hodnotu 1. Zadává se maximálně povolená hodnota odstředivého zrychlení na kruhu v mm.s^{-2} . Pro lepší nastavování se zadává hodnota zvětšená 10x.
	8. dekáda	0	Starší způsob zadání v procentech úměrnosti k odmocnině poloměru (R232).
		1	Novější způsob zadání pomocí odstředivého zrychlení.

R412 až R417 (2. sada pro dynamické řízení rychlosti)

Sada parametrů, která se uplatní po naprogramování funkce: **G23 I2** nebo z PLC programu nastavením buňky **DYNAM_SET** na hodnotu **1**.

R412	Řídící slovo pro 2. sadu parametrů řízení rychlosti.
R413	P - parabolické zrychlení.
R414	A – lineární zrychlení.
R415	Kritérium přesnosti
R416	Kritérium přetížení
R417	Kritérium odstředivého zrychlení pro kruhovou interpolaci

Význam parametrů pro 2. sadu parametrů je stejný jako pro 1.sadu.

Pokud 1.dekáda konstanty R412 je nulová (sada není povolena) použije se 1. sada parametrů R406 až R411. V případě, že některý parametr sady nebude nastaven (bude mít nulovou hodnotu), také se použije odpovídající parametr z 1.sady.

R418 až R423 (3. sada pro dynamické řízení rychlosti)

Sada parametrů, která se uplatní po naprogramování funkce: **G23 I3** nebo z PLC programu nastavením buňky **DYNAM_SET** na hodnotu **2**.

R418	Řídící slovo pro 3. sadu parametrů řízení rychlosti.
R419	P - parabolické zrychlení.
R420	A – lineární zrychlení.
R421	Kritérium přesnosti
R422	Kritérium přetížení
R423	Kritérium odstředivého zrychlení pro kruhovou interpolaci

Význam parametrů pro 3. sadu parametrů je stejný jako pro 1.sadu.

Pokud 1.dekáda konstanty R418 je nulová (sada není povolena), použije se 1. sada parametrů R406 až R411. V případě, že některý parametr sady nebude nastaven (bude mít nulovou hodnotu), také se použije odpovídající parametr z 1.sady.

R424 až R429 (4. sada pro dynamické řízení rychlosti)

Sada parametrů, která se uplatní po naprogramování funkce: **G23 I4** nebo z PLC programu nastavením buňky **DYNAM_SET** na hodnotu 3.

R424	Řídící slovo pro 4. sadu parametrů řízení rychlosti.
R425	P - parabolické zrychlení.
R426	A – lineární zrychlení.
R427	Kritérium přesnosti
R428	Kritérium přetížení
R429	Kritérium odstředivého zrychlení pro kruhovou interpolaci

Význam parametrů pro 4. sadu parametrů je stejný jako pro 1.sadu.

Pokud 1.dekáda konstanty R424 je nulová (sada není povolena), použije se 1. sada parametrů R406 až R411. V případě, že některý parametr sady nebude nastaven (bude mít nulovou hodnotu), také se použije odpovídající parametr z 1.sady.

R430 až R432 (sada pro dynamické řízení rychlosti pro rychloposuv)

Sada parametrů, která se automaticky uplatní, když je programován rychloposuv: **G00** nebo při stopu.

R430	Řídící slovo pro sadu parametrů řízení rychlosti pro G00 a pro STOP.
R431	P - parabolické zrychlení.
R432	A – lineární zrychlení.

Význam parametrů pro 4. sadu parametrů je stejný jako pro 1.sadu.

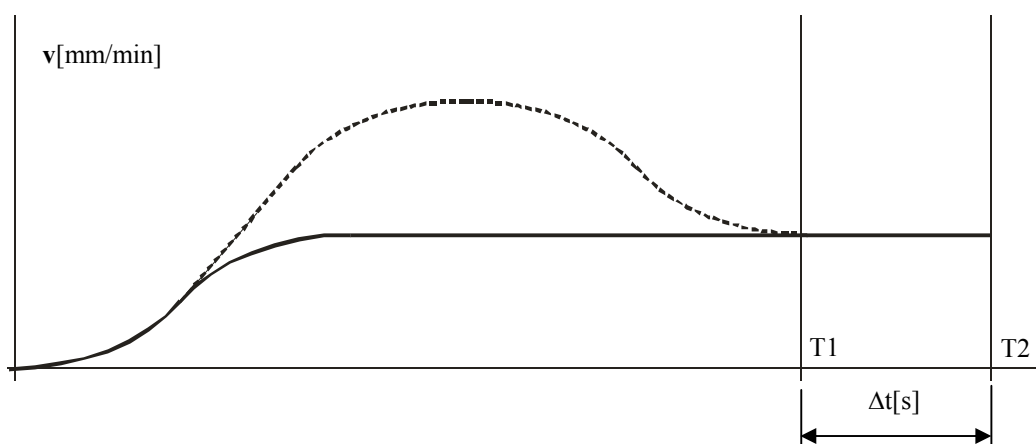
Pokud 1.dekáda konstanty R430 je nulová (sada není povolena) použije se 1. sada parametrů R406 až R408. V případě, že některý parametr sady nebude nastaven (bude mít nulovou hodnotu), také se použije odpovídající parametr z 1.sady.

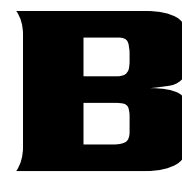
Příloha A7.4 – kritérium času pro nárůst rychlosti, parametr R395.

Kritérium času slouží pro test, zda systém zachová v rámci bloku (úseku) rychlosti vypočtené z dynamického kritéria a kritéria přesnosti na začátku a na konci bloku nebo je povolen nárůst rychlosti maximálně na programovanou rychlost a opětovný sestup rychlosti v rámci jednoho bloku. Systém si předem připraví oba případy (s nárůstem rychlosti a bez nárůstu) a vypočte rozdíl dob trvání obou případů. Pokud je rozdíl dob trvání menší než zadaný limit v konstantě R395, neoplatí se v rámci bloku zrychlovat, systém proto zachová rychlosti v rámci bloku vypočtené z kritérií.

Pokud je rozdíl dob výkonu bloků větší než zadaný limit v R395, je povolen nárůst rychlosti maximálně na programovanou rychlost a opětovný sestup rychlosti v rámci jednoho bloku. Systém proto zachová přechodové rychlosti vypočtené podle dynamického kritéria a kritéria přesnosti jen na začátku a na konci bloku.

Dva způsoby průběhu rychlosti v bloku (úseku):





Příloha B - prioritní blok

(Platí pro verze panelu 20.23 a vyšší.)

Při volbě partprogramu a po centrální anulaci systém naplní pracovní paměť tzv. prioritním blokem, ve kterém jsou uvedeny následující adresy a funkce :

F0	nulová rychlost
S0	nulové otáčky
&0	vyřazení délkových korekcí
G1	lineární interpolace
G17	korekční rovina G17
G98	zrušení plynulé návaznosti bloků
G94	posuv v mm/min
G40	zrušení poloměrových korekcí
G53	zařazené posunutí počátku
G80	zrušení pevných cyklů
G90	absolutní programování
M5	stop vřetena
M36	posuv programován přímo
M9	vypnutí chlazení 1 a 2
M53	vypnutí chlazení 3 a 4
M48	zrušení překlenutí FEED OVERRIDE

Pokud není v **prvním bloku** partprogramu programováno jinak, tyto adresy a funkce se uplatní. Všechny tyto adresy a funkce se uplatní, t.j. vyšlou také při **CENTRÁLNÍ ANULACI**.

V některých případech může uživatel požadovat další funkce, případně adresy, nebo naopak některé funkce či adresy z bloku vyřadit.

Uvádíme některé případy, které může uživatel využít:

1. Požadavek na jiný prioritní počátek než G53. V některých případech může vadit, že centrální anulace vždy nastavuje počátek na G53. Je možné si zvolit jiné prioritní nastavení, např. G54. Je také možné nezadat do prioritního bloku žádnou funkci pro posunutí počátku. V tomto případě centrální anulace ani volba programu počátek nezmění a bude platit naposledy zadaná funkce (např. z partprogramu nebo z režimu RUP). V každém případě se doporučuje vždy na začátek partprogramu programovat příslušné posunutí počátku, což patří k dobrým programátorským zvyklostem.
2. Zrušení adresy F0. Po centrální anulaci se rychlost nebude nulovat a zůstane navolená poslední zadaná. V režimu RUP potom není nutné rychlost pro G01 znovu zadávat.

V každé případě je nutné pečlivě zvážit eventuelní změny v prioritním bloku. Při změně prioritního bloku doporučujeme konzultaci s výrobcem.

Příloha B1 - nastavení prioritního bloku

Nastavení se řídí strojní konstantou číslo 283, 1 dekadou.

Hodnota 1.dekady	Popis
0	Prioritní blok má hodnoty uvedené v úvodu této kapitoly. Systém provádí toto nastavení implicitně. Toto nastavení platí také pro verze systému 20.22 a nižší.
1	Prioritní blok má hodnoty uvedené v souboru BLOCKCNF.SYS. Tyto hodnoty se po každém zapnutí systému z tohoto systémového souboru načtou.
2	Při volbě partprogramu BLOCKCNF.NCP se vytvoří (nebo přepíše, pokud již existuje) systémový soubor BLOCKCNF.SYS.

Nastavení na 0 je kompatibilní se staršími verzemi a vyhovuje ve většině případů. Systém provádí toto nastavení implicitně.

Změna prioritního bloku se provede následujícím postupem:

V systému je k dispozici soubor partprogramu s (povinným) názvem **BLOCKCNF.NCP**, který má následující obsah prvního bloku (je totožný s implicitním nastavením):

%1

**N1 F0 S0 &0 G1 G17 G98 G94 G40 G53 G80 G90 M5 M36 M9 M53 M48
N2 M30**

1. Uživatel si editorem upraví blok N1 podle svých požadavků, např. změni G53 na G54.
2. Nastaví se první dekáda strojní konstanty číslo 283 na 2.
3. Provede se volba partprogramu BLOCKCNF.NCP. Po volbě se objeví zpráva o vytvoření nového prioritního bloku.
4. Nastaví se první dekáda strojní konstanty číslo 283 na 1.
5. Po novém zapnutí systému (a při každém dalším) se již bude pracovat s novým prioritním blokem.
6. Po ověření správné činnosti je možné soubor BLOCKCNF.NCP z paměti vymazat, neboť data jsou uchována v systémovém souboru BLOCKCNF.SYS.

Pozn.1:

Je důležité po vytvoření prioritního bloku nastavit první dekádu zpět na 1, jinak by se při každé volbě souboru BLOCKCNF.NCP vytvářel systémový soubor BLOCKCNF.SYS znovu.

Pozn.2:

Z důvodů bezpečnosti vytvořených dat s prioritním blokem má systémový soubor BLOCKCNF.SYS nastavený atribut „skrytý“ a není k dispozici ani přes systémový editor. Pokud chceme uživatelský prioritní blok odstavit z činnosti, nastaví se první dekáda na 0.

Pozn.3:

Doporučuje se soubor BLOCKCNF.NCP vymazat z paměti. Pokud vymažeme soubor BLOCKCNF.NCP z paměti, můžeme ho v případě potřeby znovu nahrát z paměti EPROM (A:\PROG) u systémů bez pevného disku nebo z pevného disku (C:\SYST\PROG) u systémů s pevným diskem.

Pozn.4:

Soubor BLOCKCNF.SYS je binární soubor a nemůže být archivován pomocí systémových archivačních prostředků, tj. nelze jej zapsat do souboru ARCHIV.SYS. Do souboru ARCHIV.SYS lze eventuálně zapsat pouze jeho textový tvar BLOCKCNF.NCP.