

TG Motion

verze 4

Struktura Interpolátor

návod k obsluze

Historie revizí

datum	verze	popis
10. 1. 2017	1.0	
31. 7. 2017	1.1	upravena typografie

Obsah

1. Interpolátor	4
1.1 CNC modul	4
1.2 Interpolátor	4
1.3 Kompetence PLC a Windows aplikací	4
1.4 G-kód	5
2. Vyhazení trajektorie	7
2.1 Spline	7
2.2 IIR Filtr	8
3. Struktura Command	10
3.1 Registry Command_Parametr	10
4. Struktura LookAheadBuffer	11
4.1 Popis struktury	11
4.2 Použití struktury	11
5. Apendix	12
A. Přehled a popis registrů struktury Interpolátor	12
B. Přehled a popis registrů struktury LookAheadBuffer	13

1. Interpolátor

1.1 CNC modul

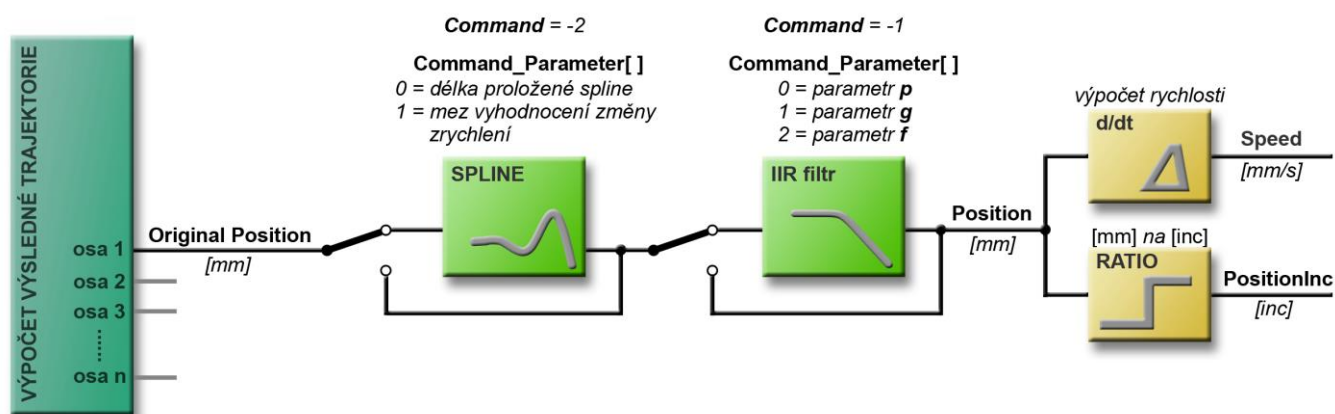
CNC je softwarový modul, který vykonává sekvenci pohybových (G-instrukce) a vstupně/výstupních příkazů (M-funkce) danou **G-kódem**. Jeho součástí je modul Interpolátor.

1.2 Interpolátor

Modul interpolátoru počítá polohy a rychlosti jednotlivých servopohonů (os) tak, aby byl výsledný pohyb všemi osami rovnoměrně prováděn. K dispozici je lineární interpolace (pohyb po přímce), kruhová interpolace (pohyb po kružnici prováděný dvěma libovolnými osami), případně šroubová interpolace (dvě osy vykonávají pohyb po kružnici, ostatní vykonávají lineární pohyb).

K výpočtu tří nezávislých finálních trajektorií víceosého pohybu nabízí systém **TG Motion** tři nezávislé **Interpolátory**, z nichž každý může pracovat až s deseti servopohony. Sdílená paměť **TGM_Interpolator** slouží jako rozhraní mezi CNC modulem a ostatními aplikacemi (PLC, Win aplikace). Většina registrů je určena jen pro čtení a zobrazuje aktuální hodnoty CNC modulu.

Protože G-kód je většinou psán v jednotkách [mm], pracují i Interpolátory s jednotkami [mm]. K nastavení převodu [mm] na [inc] slouží struktura Command nebo registr Ratio (viz dále).



Obr. Blokové schéma Interpolátoru

1.3 Kompetence PLC a Windows aplikací

PLC

nastavuje základní parametry Interpolátoru, obsluhuje M-funkce a řeší havarijní zastavení Interpolátoru. Z PLC Interpolátor spustit nelze.

Windows aplikace

převádějí G-kód do binárního mezikódu, plní jím buffer Interpolátoru ve sdílené paměti a spouštějí i zastavují Interpolátor. Buffer Interpolátoru není z PLC ani Windows aplikací přístupný.

1.4 G-kód

G-kód je název programovacího jazyka, který řídí NC a CNC obráběcí stroje. Jedná se o kód v textovém formátu, který říká obráběcímu stroji, jakou akci má vykonat. V G-kódu mezi nejpoužívanější příkazy patří **G-instrukce** a **M-funkce**. Jsou vždy uvozeny písmenem (G, M) a číselným parametrem, který specifikuje, co má daná instrukce vykonávat. Jednotlivé G-instrukce jsou přesně definovány, většinu M-funkcí může definovat uživatel.

Ukázka části G-kódu

```
G00 X60.6051 Y40.7723
G42
M51
G02 X0 Y-2.5202 I3.0336 J-1.2601
G02 X0.922 Y-1.5646 I6.0604 J2.5173
G02 X1.079 Y-1.0269 I4.2143 J3.3479
G02 X2.2624 Y-1.3678 I9.8772 J13.7817
G02 X1.9856 Y-0.8109 I6.1778 J12.2908
G02 X3.015 Y-0.7892 I8.8515 J27.6632
G02 X6.2889 Y0 I3.1444 J33.011
M50
G40
G00 X-28.9085 Y30.4365
G42
M51
G40
M2
```

G-instrukce

Slouží většinou k rychlému polohování, posuvnému pohybu po přímce nebo oblouku, vrtání nebo řezání. O jejich plnění se stará Interpolátor. Průběh vykonávání **G-instrukcí** může být přerušeno plněním **M-funkcí**. V tom případě Interpolátor může čekat na vykonání **M-funkce** a pak pokračovat dále ve vykonávání **G-kódu** (viz **Chyba! Nenalezen zdroj odkazů.**).

Tabulka podporovaných G-instrukcí

základní G-instrukce		rozšířené G-instrukce	
G0	rychlý posuv	G4	prodleva [sec]
G1	lineární interpolace	G25	volání podprogramu
G2	kruhová interpolace CW	G26	volání cyklu
G3	kruhová interpolace CCW	G27	programový skok
G40	zrušení korekce	G29	návěští nebo textová poznámka
G41	korekce vlevo od kontury	G53	zrušení posunutí nulového bodu
G42	korekce vpravo od kontury	G54	posunutí nulového bodu
G90	absolutní programování	G92	nastavení hodnot souřadnic
G91	inkrementální (přírůstkové) programování		

M-funkce

Pokud je součástí kódu libovolná **M-funkce**, její obslužení je třeba zabezpečit v uživatelském PLC kódu. Interpolátor se M-funkcemi nezabývá, jen u některých čeká na jejich obslužení.

M-funkce mohou být až na výjimky uživatelsky definovány (viz registr M_Funkce_Parametr).

Pohybové a průběžné M-funkce

Pohybové M-funkce (Mx, kde x < 1000) – vykonávání G-kódu se zastaví, PLC obsluží M-Funkci a po jejím dokončení nastaví registr M_Func = 0. Interpolátor pak pokračuje ve vykonávání G-kódu další instrukcí.

Průběžné M-funkce (M_x, kde x > 1000) – PLC spustí obsluhu M-funkce, ale Interpolátor nečeká na obsluhu M-funkce a pokračuje dál ve vykonávání G-kódu.

Tabulka podporovaných M-funkcí

základní M-funkce	
M0	programový stop
M2	ukončení G-kódu, lze předefinovat
M3	roztočení vřetene CW, lze předefinovat
M4	roztočení vřetene CCW, lze předefinovat
M5	zastavení vřetene, lze předefinovat
M6	výměnu nástroje, lze předefinovat
M7	zapnutí chlazení, lze předefinovat
M8	zapnutí mazání, lze předefinovat
M9	vypnutí chlazení i mazání, lze předefinovat
M17	návrat z podprogramu (RETURN), lze předefinovat
M29	výstup textového hlášení (PRINT), lze předefinovat
M30	ukončení G-kódu, lze předefinovat
M80	vypnutí zrcadlení, lze předefinovat
M81	zrcadlení v ose x, lze předefinovat
M82	zrcadlení v ose y, lze předefinovat
M83	zrcadlení v ose z, lze předefinovat
M84	zrcadlení v osách x a y, lze předefinovat
M85	zrcadlení v osách x a z, lze předefinovat
M86	zrcadlení v osách y a z, lze předefinovat
M87	zrcadlení v osách x, y a z, lze předefinovat
M99	definování výchozí hodnoty posuvu, lze předefinovat

2. Vyhlazení trajektorie

G-kód často pracuje s polynomy složenými z krátkých úseček, někdy může také docházet k okamžitým změnám rychlosti vlivem nesprávně zapsaného kódu. Všechny okamžité změny rychlosti se mohou projevit nežádoucím zrychlením v některých z os, a tím i mechanickým rázem v některých servopohonech. K vyhlazení vypočtené trajektorie slouží dva nástroje.

2.1 Spline

Spline – funkce vyhlazení

Funkci **Spline** je vhodné použít v případě nekorektně vytvořených G-kódů, ve kterých jednotlivé úseky na sebe plynule či spojitě nenavazují, nebo v případě G-kódů, v nichž je výsledná trajektorie vytvořena pomocí lineárních úseků s hrubým dělením. **Spline** je také vhodný k vyhlazení skokové změny rychlosti, která vyplývá z matematického modelu mechaniky (např. naklápěcí hlavy).

Funkce **Spline** se aktivuje ve všech osách současně, nelze ji aplikovat jen na některé osy. Po její aktivaci je v každé ose kontrolována změna zrychlení (jerk). V případě, že je nalezena změna větší než povolená, proloží se daným místem spline křivka.



*Čím je větší délka spline (velikost Spline bufferu), tím je vyhlazení lepší. Vyhlazení pomocí funkce **Spline** je na úkor přesnosti polohování.*

Aktivace a parametrizace Spline

Funkce **Spline** se aktivuje a parametrizuje pomocí struktury **Command** daného Interpolátoru. Pro hodnotu registru **Command** = -2 mají **Command** parametry následující význam:

Command_Parametr[0] – určuje počet bodů, kterými se prokládá spline. Potažmo se tak určuje velikost Spline bufferu. Rozsah nastavení parametru je 50–500 bodů/kroků. Při **Cycle_Time** = 500 μ s je krok výpočtu 100 μ s, při **Cycle_Time** = 250 μ s je krok výpočtu 50 μ s.

Vypnutí funkce **Spline** je možné nastavením délky spline na nulu (**Command_Parametr[0]** = 0).

Command_Parametr[1] – určuje mez vyhodnocení změny zrychlení (jerk), tedy od jakého zrychlení se funkce Spline aktivuje. Udává se v jednotkách [mm/s³]. Vhodná hodnota nastavení je 1 000 000 mm/s³.

Po nastavení obou parametrů je třeba nastavit registr **Command** na hodnotu -2. Po provedení příkazu TG Motion tento registr vynuluje.

Zpoždění pohybu

Pohyb po průchodu funkcí **Spline** je oproti polohování vypočteného interpolátorem zpožděn o délku Spline bufferu. Všechny osy polohují neustále synchronně, protože velikost bufferů je pro všechny osy stejná. I M-funkce jsou volány synchronně s výsledným pohybem, protože je pro ně použit buffer stejné velikosti.

Příklad nastavení Spline

Příklad zapnutí funkce **Spline** s délkou spline 20 ms (při **Cycle_Time** = 500 μ s). Nejdříve je třeba nastavit hodnoty **Command_Parametrů** a jako poslední hodnotu registru **Command** = -2.

```
Interpolator.Command_Parametr[0] = 200;  
Interpolator.Command_Parametr[1] = 1000000;  
Interpolator.Command = -2;
```

2.2 IIR Filtr

IIR Filtr – funkce vyhlazení

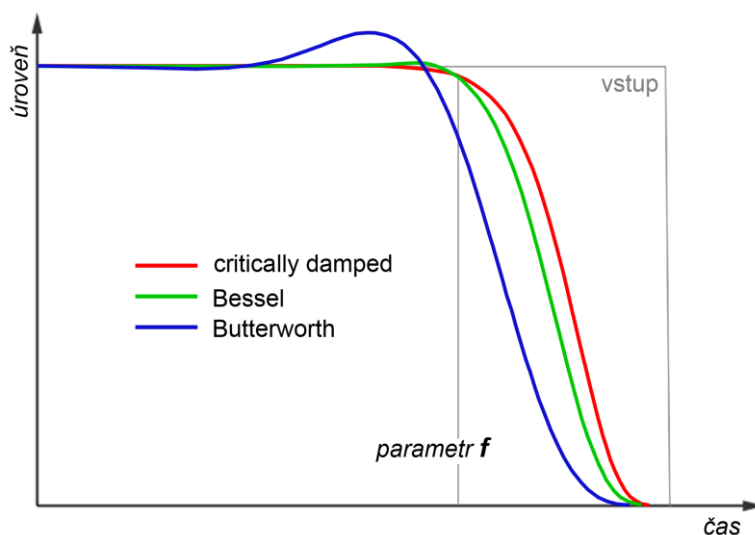
K vyhlazení výsledné rychlosti pohybu a odstranění nežádoucích rychlých změn lze použít **IIR Filtr** (Infinite Impulse Response). Jedná se o matematický model lineárního filtru typu dolní propust (Low-pass) se strmostí 12 dB na oktávu (2-Pole) počítaný podle vztahu $H(s) = q / (s^2 + p*s + q)$ s možností nastavení tří parametrů:

p, q – parametry nastavující průběh filtru (viz tabulku)

f (cutoff) – čas, za který se filtrovaná trajektorie vrátí k původní vypočtené trajektorii

Hodnoty parametrů pro některé typy filtrů

p	q	typ filtru	popis
$\sqrt{2}$	2	Butterworth	před f dochází ke zdvihu, malé zpoždění
3	3	Bessel	mírný zdvih, hladký průběh, větší zpoždění
2	1	critically damped	hladký pozvolný průběh bez zdvihu, největší zpoždění



Obr. Grafické znázornění průběhu některých typů filtrů

Aktivace a parametrizace IIR Filtru

Hodnoty parametrů **IIR Filtru** lze nastavit pomocí struktury **Command**. Pro hodnotu registru **Command = -1** mají **Command** parametry následující význam:

Command_Parametr[0] – nastavuje parametr **p** filtru.

Command_Parametr[1] – nastavuje parametr **q** filtru.

Command_Parametr[2] – určuje **f** – čas návratu k původní trajektorii.



*Hodnoty parametrů jsou typu Integer, skutečné fyzické hodnoty parametru se dosáhne dělením hodnoty parametru číslem 10^6 (**Command_Parametr[1]** = 1 000 000, parametr filtru = 1,0).*

Command_Parametr[3] – nastavuje bitovou masku aktivity filtrů pro jednotlivé osy. Používá se pouze prvních 10 bitů pro 10 os, ostatní bity jsou ignorovány (viz registry).

Zpoždění pohybu

IIR Filtry vypočítaný pohyb zpožďují. Zpoždění je závislé na vstupních datech, dynamicky se mění a jeho velikost nelze předem určit. M-funkce jsou z běžícího G-kódu signalizovány při „stojícím“ Interpolátoru, fyzický pohyb servopohonů ale nemusí být ještě ukončen.



*Při použití **IIR Filtru** bude servopohon „plavat“ za Interpolátorem a je vždy třeba počkat, až je fyzický pohyb všech servopohonů skutečně dokončen.*

Nastavení parametrů **IIR Filtrů** je společné pro všechny osy Interpolátoru. Filtr je možné aktivovat nezávisle pro jednotlivé osy podle bitové masky registru **Command_Parametr[3]**. Je tedy třeba počítat s tím, že výsledný pohyb os po aplikaci filtru nemusí být vzájemně synchronní tak, jak byl původně vypočten.

3. Struktura Command

3.1 Registry Command_Parametr

V kapitole je popsán význam registrů Command_Parametr[] pro vybrané hodnoty registru Command.

Nastavení převodového poměru mm na inc – Command = 1024

V G-kódu se většinou implicitně pracuje s jednotkou [mm], proto Interpolátor pracuje také v jednotkách [mm]. Protože servopohony pracují s inkrementy [inc], je třeba podle počtu inc/otáčku servopohonu (Servo[x].Resolution) stanovit převodový poměr inc/mm. Ten se nastavuje pomocí struktury **Command** příslušného Interpolátoru. Převodovým poměrem se pak násobí vypočítaná poloha v dané ose a takto přepočtená se pak posílá do servopohonu.

Command_Parametr	popis
0	určuje číslo osy, u níž se poměr nastavuje [0–9]
1	čitatel převodového poměru [inc]
2	jmenovatel převodového poměru [mm]

Nastavení aktuální polohy – Command = 2048 a Command = 2049

Pomocí Command = 2048 a Command = 2049 se nastavuje aktuální poloha interpolátoru. Po provedení příkazu 2048 systém hlásí polohu na trajektorii, v druhém případě polohu mimo trajektorii. Pro obě varianty nabývají všechny Command_Parametr[0-9] stejného významu.

Command_Parametr	popis
0	aktuální poloha v ose 0 [inc]
...	...
9	aktuální poloha v ose 9 [inc]

Nastavení parametrů IIR Filtru – Command = -1

Command_Parametr	popis
0	parametr p filtru – nastavení průběhu filtru (v kombinaci s parametrem q)
1	parametr q filtru – nastavení průběhu filtru (v kombinaci s parametrem p)
2	parametr f filtru – čas, za který se filtrovaná trajektorie vrátí k původní trajektorii
3	bitová maska aktivity IIR Filtrů pro jednotlivé osy Interpolátoru <i>příklad: xxxx xx00 0000 0101 – aktivní je pouze IIR Filtr pro osu 0 a osu 2</i>

Nastavení parametrů Spline – Command = -2

Command_Parametr	popis
0	určuje počet bodů, kterými se spline prokládá; rozsah nastavení je 50–500 bodů
1	horní mez vyhodnocení změny zrychlení [mm/s ³]

4. Struktura LookAheadBuffer

4.1 Popis struktury

Struktura **LookAheadBuffer** je tabulka důležitých parametrů osmi úseků – položek G-kódu jdoucích po sobě. První položkou tabulky je vždy právě prováděný úsek, dalšími položkami je sedm bezprostředně následujících úseků.

LookAheadBuffer funguje jako posuvný buffer. Po provedení aktuálního úseku se data tabulky posunou, první je opět vykonávaný úsek a na poslední místo se vloží 8. následující úsek od aktuálně vykonávaného. Strukturu **LookAheadBuffer** naplňuje Interpolátor. Z hlediska PLC jsou její registry pouze ke čtení.

Úsek G-kódu

Úsekem se rozumí jedna položka G-kódu, a to buď G-instrukce (G0–G3) nebo pohybová M-funkce, která má význam z hlediska pohybu (M3–M999).

4.2 Použití struktury

Struktura **LookAheadBuffer** slouží k přizpůsobení technologií podle instrukcí a funkcí následujících úseků, případně podle jejich hodnot. Protože z PLC nelze Interpolátor spustit ani zastavit (kromě havarijního stop), je třeba během obsluhy existujících M-funkcí obhospodařit vše potřebné. To musí zabezpečit tvůrce PLC kódu.

PLC může během M-funkce například změnit úhel obráběcí hlavy v rohu čtverce podle tangenty následující funkce; může pomocí registru Rel_Speed zpomalit pohyb, pokud ví, že bude následovat velká změna tangenty. Při pohybu po kružnici je k dispozici neustále aktuální tangenta, která se dá použít pro natáčení hlavy. Při krátkých úsečkách lze také postupným natáčením hlavy vyhladit lomenou úsečku. K vykonání všech těchto úkonů je nutné vědět, co bude následovat za právě vykonávaným úsekem. K tomu lze právě využít struktury **LookAheadBuffer**.



Pozn.: Někdy se pro Movement_Type = 2 může v registru Movement_Code objevit záporné číslo M-funkce. Jedná se o vnitřní funkci TG Motion, s níž není třeba nijak pracovat.

Jednotlivé registry struktury **LookAheadBuffer** jsou popsány v kapitole Appendix.

5. Apendix

A. Přehled a popis registrů struktury Interpolátor

Interpolátor			
název	přístup	offset	popis
Number	R	0	číslo interpolátoru, může nabývat hodnot 0, 1, 2
Number_Axes	RW	4	počet os, se kterými interpolátor pracuje; povolené rozmezí je 1–10
Buffer_Size	R	8	maximální počet úseků G-kódu, povolené hodnoty jsou 1000–100000
Command	RW	12	příkaz: 4 = havarijní stop na trajektorii 5 = havarijní stop (po stopu hlásí mimo trajektorii) 8 = normální stop na trajektorii 9 = normální stop (mimo trajektorii) 1024 = nastavení převodních poměrů (mm na inc) – viz kap. Command 2048 = nastavení aktuální polohy (na trajektorii) – viz kap. Command 2049 = nastavení aktuální polohy (mimo trajektorii) – viz kap. Command -1 = nastavení parametrů IIR Filtru – viz kap. Command -2 = nastavení parametrů Spline – viz kap. Command
Command_Parametr [0-11] (12 registrů)	RW	16	12 parametrů typu integer, jejichž význam a hodnoty závisejí na typu funkce Command
Command_Status	R	64	aktuální stav prováděného příkazu: 0 = předchozí příkaz byl úspěšně vykonán a lze aktivovat další příkaz 1 = aktuální příkaz se provádí -1 = během vykonávání příkazu došlo k chybě
Status	R	68	aktuální stav interpolátoru: 1 = probíhá pohyb po trajektorii 3 = stop na konci trajektorie 4 = v bufferu je nejméně jeden úsek trajektorie, lze volat start 6 = zastavování po havarijní rampě 7 = interpolátor se zastavil po havarijním stopu 8 = interpolátor byl zastaven během plnění bufferu
Act_Part	R	72	číslo aktuálně prováděného úseku
Address_External_Position	RW	76	offset adresy TGM_Data, kde je uložena poloha externího snímače; hodnota je typu integer
M_Func	RW	80	hodnota M-funkce; pro M < 1000 Interpolátor zastaví a čeká na vynulování této hodnoty; pak pokračuje dalším úsekem; pro M > 1000 se pokračuje ve vykonávání G-kódu
Act_G_Func	R	84	hodnota právě prováděné G-instrukce
Act_M_Func	R	88	hodnota právě prováděné M-funkce
Last_Cont_M_Func	R	92	uložená hodnota poslední průběžné M-funkce (M > 1000)
Run_Flag	R	96	- spodních 16 bitů ukazuje stav aktuálního úseku: 0 = STOP (interpolátor je neaktivní) 1 = RUN (vykonává se pohybová G-instrukce) 2 = WAIT_WINDOW (pro otáčkové obrábění) 3 = WAIT_PULSE (pro otáčkové obrábění) 4 = WAIT_MFUNC (začalo provádění M-funkce) 5 = WAIT_MFUNC_WAIT_FOR_END (čekání na ukončení M-funkce) - horních 16 bitů pak stav rychlostního průběhu: 1 = žádný pohyb 2 = zrychlování 3 = dosažena požadovaná rychlost pohybu 4 = zpomalování 5 = další rychlostní úsek 6 = zpomalování na posledním úseku 7 = zpomalování po havarijní rampě <i>pozn.: od verze TGM420</i>
Tool_Number	R	100	číslo aktuálního nástroje (vrtání, fréza, ...) <i>pozn.: od verze TGM420</i>
Orig_Position (10 registrů)	R	376	vypočítané souřadnice všech os [mm]
Position (10 registrů)	R	456	souřadnice upravené funkcí Spline nebo IIR Filtrem [mm]
PositionInc (10 registrů)	R	536	souřadnice Position vynásobené převodním poměrem Ratio [inc]
Backlash (10 registrů)	R	616	aktuální hodnota vůle pro jednotlivé osy [inc]
Offset (10 registrů)	RW	696	hodnoty offsetu se připočítávají k poloze PositionInc, tyto hodnoty nastavuje uživatel [inc]
Speed (10 registrů)	R	776	aktuální rychlost jednotlivých os po Spline a IIR [mm/s]
Ratio (10 registrů)	RW	856	převodní poměr (násobitel) jednotek G-kódu (obvykle mm) na inkrementy (polohy servopohonů)

název	přístup	offset	popis
M_Funkce_Parametr (32 registrů)	R	936	- parametry M funkce z G-kódu, celkem 26 hodnot podle písmen v abecedě; některá písmena nelze použít, protože jsou vyhrazena systémem - rezervované parametry (indexy jsou počítány od 0): G = index 6 M = index 12 N = index 13 P = index 15
Rel_Speed	RW	1192	relativní rychlost interpolovaného pohybu, koeficient v rozmezí 0,01–2 (1–200 %)
Set_Speed	R	1200	požadovaná rychlost z G-kódu (daná F-instrukcí G-kódu nebo tabulkou rychlostí) [mm/min]
Act_Speed	R	1208	aktuální rychlost [mm/min]
Move_Distance	R	1216	aktuální celková ujetá vzdálenost [mm]
LookAheadBuffer	R	2048	tabulka informací o úsecích, které budou následovat; celkem 8 položek struktury popsané v následující tabulce; první položka popisuje právě prováděný úsek; velikost každé položky je 1792 bytů

pozn.: od verze TGM420

B. Přehled a popis registrů struktury LookAheadBuffer

Struktura LookAheadBuffer

pozn.: od verze TGM420

název	přístup	offset	popis
AllParams (26 registrů)	R	0	všechny zadané adresy M-funkcí z konkrétního G-kódu pro daný úsek (26 písmen anglické abecedy)
Tangens	R	208	aktuální tečna pohybu v rovině XY; pokud je právě prováděný úsek oblouk (G2 nebo G3), tečna se mění průběžně; pro budoucí úseky určuje počáteční tečnu pohybu
MovementType	R	216	typ záznamu: 0 = neplatný záznam (počet zbývajících úseků k provedení je menší než aktuální index tabulky LookAheadBuffer), do konce provedení pohybu zbývá méně než 8 úseků 1 = pohybová funkce (G0, G1, G2, G3) 2 = M funkce
MovementCode	R	220	číslo G-instrukce nebo pohybové M-funkce (číslo průběžné M-funkce se v tomto registru neobjeví, ale objeví se v AllParams v písmenu M)
Plane	R	224	- rovina kruhové interpolace: 17 = XY 18 = XZ 19 = YZ - ostatní roviny nejsou definovány
Tool	R	228	číslo nástroje
EndPos (10 registrů)	R	232	koncová poloha úseku jednotlivých os [mm] (absolutní souřadnice)
StartAngle	R	312	počáteční úhel oblouku [°] – úhel spojnice středu oblouku s počátečním bodem; hodnota registru je cyklická (0–360); pro lineární pohyb či M-funkce je hodnota registru větší než 10 ³⁸
EndAngle	R	320	koncový úhel oblouku [°] – úhel spojnice středu oblouku s koncovým bodem; hodnota registru je cyklická (0–360); pro lineární pohyb či M-funkce je větší než 10 ³⁸
Radius	R	328	- poloměr oblouku [mm] - pro lineární pohyb či M-funkce je registr nulový