

NCT2xx
NCT3xx
szerszámgép vezérlők

PLC programozás

Az n.15.2 szoftver változattól

Gyártó és fejlesztő: **NCT Ipari Elektronikai kft.**

H1148 Budapest Fogarasi út 7

✉ Levélcím: 1631 Bp. pf.26

☎ Telefon: (+36 1) 467 63 00

☎ Telefax: (+36 1) 467 63 09

Villanyposta: nct@nct.hu

Honlap: www.nct.hu

Tartalomjegyzék

1 A PLC program nyelve.	8
1.1 Mintavételezés, a be- és kimenetek kezelése.	9
1.2 A PLC program végrehajtási sorrendje.	9
1.3 A PLC program szerkesztése.	10
2 A PLC program által használt memória.	11
2.1 Dupla szó (DWORD) címezése.	13
2.2 Dupla szó (DWORD) indexelt címezése a “,” operátorral.	13
2.3 Bit közvetlen címezése és a “.” operátor.	15
2.4 Bit közvetett címezése a “.” operátorral.	17
2.5 Bit indexelt címezése a “,” operátorral.	18
2.6 Lebegőpontos szám (double) címezése.	19
2.7 Lebegőpontos szám (double) indexelt címezése a “,” operátorral.	20
3 A PLC program moduljai.	21
3.1 A Főprogram.	21
3.2 Az Int0 modul.	21
3.3 A PLC memória frissítése.	21
4 A PLC program által kezelt adatok.	23
4.1 Bites adatok kezelése.	23
4.2 Memóriabit felfutó élének lekérdezése a @ operátorral.	23
4.3 Memóriabit lefutó élének lekérdezése a % operátorral.	23
4.4 Bemenet azonnali lekérdezése, kimenet azonnali kiadása a ! operátorral.	24
4.5 Decimális, előjeles szám megadása a # operátorral.	24
4.6 Hexadecimális szám megadása a #\$ operátorral.	25
4.7 Előjel nélküli, BCD szám megadása a #\$ operátorral.	25
4.8 Lebegőpontos szám megadása a * operátorral.	25
4.9 Az IEEE754 szabvány dupla pontosságú lebegőpontos számábrázolása.	26
5 A PLC utasítások által állított státuszbit.	28
5.1 Az FL_ER (error) hiba jelző.	28
5.2 Az FL_UF (underflow) alulcsordulás jelző.	28
5.3 Az FL_OF (overflow) túlcsordulás jelző.	28
5.4 Az FL_CY (carry) átvitel jelző.	28
5.5 Az FL_GT (greater than) nagyobb, mint jelző.	29
5.6 Az FL_EQ (equal) egyenlő jelző.	29
5.7 Az FL_LT (lower than) kisebb, mint jelző.	29
6 A PLC program utasításai.	30
6.1 Bitműveleteket végző utasítások.	30
6.1.1 Záróérintkező: memóriabit lekérdezése.	30
6.1.2 Záróérintkező: dupla szó lekérdezése.	31
6.1.3 Nyitóérintkező: memóriabit negált lekérdezése.	32
6.1.4 Nyitóérintkező: dupla szó negált lekérdezése.	33

6.1.5 Relétekeres: memóriabit írása.	34
6.1.6 Negált relétekeres: memóriabit negált írása.	35
6.1.7 Memória bitjének beállítása: a SET utasítás.	35
6.1.8 Memória bitjének törlése: az RST utasítás.	36
6.1.9 Impulzusképzés felfutó élre: a DIFU utasítás.	37
6.1.10 Impulzusképzés lefutó élre: a DIFD utasítás.	38
6.1.11 A bitműveleteket végző utasítások és a ! operátor a két modulban.	39
6.2 A létrahálózat kialakításának alapszabályai.	41
6.2.1 Összekötő elemek.	42
6.2.2 A létradiagram logikai szektorainak kommentezése: a SEC utasítás.	43
6.3 Adatmozgató utasítások.	44
6.3.1 Dupla szó (DWORD) mozgatása: a MOV és az MVN utasítás.	45
6.3.2 Lebegőpontos (double) adat mozgatása: MOVF.	46
6.4 Időzítők.	47
6.4.1 Késleltetve behúzó időzítő TOND.	48
6.4.2 Késleltetve kiejtő időzítő TOFFD.	49
6.4.3 Programozható szélességű impulzus TPULSE.	50
6.5 Számlálók.	51
6.5.1 Egyszerű számláló CNT.	52
6.5.2 Reverzibilis számláló CNTR.	54
6.6 Forgatásvezérlő utasítás ROT.	56
6.7 Eltoló és forgató utasítások.	59
6.7.1 Shift regiszter SHTR.	59
6.7.2 Aritmetikai eltolások: ASHL, ASHR.	60
6.7.3 Aritmetikai forgatások: ARTL, ARTR.	62
6.8 Logikai utasítások.	65
6.8.1 Egyoperandusú utasítás: NEG.	65
6.8.2 Kétooperandusú utasítások: AND, OR, XOR.	66
6.9 Fixpontos aritmetikai utasítások.	69
6.9.1 Előjeles, fixpontos összeadás, átvitel nélkül: ADD.	70
6.9.2 Előjeles, fixpontos kivonás, átvitel nélkül: SUB.	71
6.9.3 Előjeles, fixpontos szorzás: MUL.	72
6.9.4 Előjeles, fixpontos osztás: DIV.	73
6.10 Lebegőpontos matematikai műveletek.	74
6.10.1 Lebegőpontos összeadás: +F.	75
6.10.2 Lebegőpontos kivonás: -F.	76
6.10.3 Lebegőpontos szorzás: *F.	77
6.10.4 Lebegőpontos osztás: /F.	78
6.10.5 Hatványozás: PWR.	79
6.10.6 Négyzetgyök: SQRT.	80
6.10.7 Szinusz: SIN.	81
6.10.8 Koszinusz: COS.	82
6.10.9 Tangens: TAN.	83
6.10.10 Arkusz szinusz: ASIN.	84
6.10.11 Arkusz koszinusz: ACOS.	85
6.10.12 Arkusz tangens: ATAN.	86
6.10.13 Természetes alapú (e) hatvány: EXP.	87
6.10.14 Természetes alapú (e) logaritmus: LOG.	88
6.11 Konverziós utasítások.	89

6.11.1 BCD szám bináris számmá alakítás: BIN.	90
6.11.2 Bináris szám BCD számmá alakítása: BCD.	91
6.11.3 Fixpontos szám lebegőpontos számmá alakítása: FLT.	92
6.11.4 Lebegőpontos szám fixpontos számmá alakítása: FIX.	93
6.11.5 Radiánban megadott szög fokra konvertálása: DEG.	94
6.11.6 Fokban megadott szög radiánra konvertálása: RAD.	95
6.12 Összehasonlító utasítások.	96
6.12.1 A CMP és FCMP utasítások.	96
6.12.2 Az érintkező típusú összehasonlító utasítások.	97
6.13 A PLC programból küldhető üzenetek.	99
6.13.1 Az üzenetküldő utasítások: MSG, MSGF, ALR, ALRF, REM, REMF.	101
6.14 Programvezérlő utasítások.	105
6.14.1 Modul vége utasítás: END.	105
6.14.2 Ugrás a PLC programmodulban: JMP és JME utasítások.	106
6.14.3 Szubrutin hívás utasításai: SBS, SBN és RET utasítások.	107
6.15 A tengelymozgató utasítás: MOVCMMD.	109
6.16 A globális makróváltozók írása és olvasása.	113
6.16.1 Globális makróváltozó olvasása: a MACR utasítás.	113
6.16.2 Globális makróváltozó írása: a MACW utasítás.	114
6.17 Az NC belső változóinak lekérdezése: az SCP utasítás.	115
6.18 Az NC belső változóinak tömbös olvasása és írása.	123
6.18.1 NC memóriatömb olvasása: az MR utasítás.	123
6.18.2 NC memóriatömb írása: az MW utasítás.	124
6.19 Adatok mozgatása a nem felejtő tár és a PLC memória között.	127
6.19.1 PLC változók adatainak kiolvasása nem felejtő tárból.	127
6.19.2 PLC változók adatainak kiírása nem felejtő tárba.	128
6.20 Paraméterek olvasása és írása PLC programból.	130
6.20.1 DWORD típusú paraméterek olvasása.	130
6.20.2 Double típusú paraméterek olvasása.	132
6.20.3 DWORD típusú paraméter írása.	134
6.20.4 Double típusú paraméterek írása.	136
6.21 Program kijelölése végrehajtásra.	138
6.21.1 Programszámával megadott program kijelölése automata végrehajtásra.	138
6.22 Hálózati kommunikációs utasítások.	139
6.22.1 Hálózati kapcsolat megnyitása.	139
6.22.2 Hálózati kapcsolat bezárása.	140
6.22.3 Hálózati adatcsomag fogadása.	141
6.22.4 Hálózati adatcsomag küldése.	142
6.23 Ablaknyitás a vezérlő képernyőjén.	144
6.24 Hajtásadatok kiírása.	147
6.25 Pozícióadatok beírása.	149
6.26 A szerszámkezelő táblázat adatainak írása és olvasása.	151
6.26.1 A szerszámkezelő táblázat.	151
6.26.2 A szerszámhely táblázat.	156
6.26.3 A szerszámalak táblázat.	158
6.26.4 Adatszám cseréje két különböző magazin két különböző zsebe között.	162
6.26.5 Üres zseb keresése.	163
6.26.6 Új szerszám regisztrációja a Szerszámkezelő táblázatban.	165
6.26.7 Szerszám Szerszámkezelői adatainak átírása.	169

6.26.8 Szerszám Szerszámkezelői adatainak beolvasása.	173
6.26.9 Szerszám Szerszámkezelő adatainak törlése.	176
6.26.10 Szerszám egy Szerszámkezelői adatának átírása.	177
6.26.11 Szerszám egy Szerszámkezelői adatának beolvasása.	180
6.26.12 Szerszám keresése Felhasználói adat alapján.	182
6.27 A palettakezelő táblázat adatainak írása és olvasása.	184
6.27.1 A palettakezelő táblázat.	184
6.27.2 Adatok cseréje két különböző palettamagazin két különböző helye között. . .	189
6.27.3 Paletta adatainak átírása.	191
6.27.4 Paletta adatainak beolvasása.	194
6.27.5 Paletta egy palettakezelő adatának átírása.	196
6.27.6 Paletta egy palettakezelő adatának beolvasása.	198
6.27.7 Paletta keresése adatszám értéke alapján.	200
6.27.8 Paletta magazin adott adatszámú értékeinek kiolvasása.	202
6.27.9 Palettaazonosítójával elérhető program kijelölése automata végrehajtásra. . .	203
6.28 Mailbox kommunikáció a PLC program és tetszőleges EtherCAT eszköz között. .	206
6.28.1 EtherCAT mailbox adatainak olvasása.	206
6.28.2 EtherCAT mailbox adatainak írása.	212
6.29 Az MR, MW utasítások teljesítési kódjai.	218
7 A PLC program és az NC közötti kommunikáció.	222
7.1 NCT gépi kezelőpanelek.	224
7.2 NCT kézikerekek.	231
7.3 Kétállapotú, 24 V-os interfész be-, kimenetek.	234
7.4 NCT tapintó illesztő kártyák be-, kimenetei.	236
7.5 NCT érzékelő bemenetek.	238
7.6 NCT analóg bemenetek.	240
7.7 Az EtherCAT-es NCT hajtások be-, kimenetei.	241
7.8 Jeladó fogadó bemenetek és analóg/léptető motor/CAN illesztő kimenetek.	250
7.9 PLC-ből elérhető funkciógombok.	256
7.10 A paraméteren állítható pozíció kapcsolók.	257
7.11 Hozzáférés a PLC Constants paramétercsoporthoz.	259
7.12 A globális változók.	262
7.12.1 A bites globális változók.	262
7.12.2 A duplaszavas globális változók.	268
7.13 A tengelykezelő változók.	269
7.13.1 A bites tengelyváltozók.	269
7.14 Az orsókezelő változók.	285
7.14.1 A bites orsóváltozók.	285
7.14.2 A duplaszavas orsóváltozók.	297
7.14.3 A lebegőpontos orsóváltozók.	299
7.15 A csatornakezelő változók.	300
7.15.1 A bites csatornaváltozók.	300
7.15.2 A duplaszavas csatornaváltozók.	332
7.15.3 A lebegőpontos csatornaváltozók.	344
Betűrendes index.	347

19.11.21

1 A PLC program nyelve

A PLC program nyelve **létradiagram** formátumú.

A létradiagram a relés **áramútervnek** a vezérléstechnikában alkalmazott, egyszerűsített, formája.

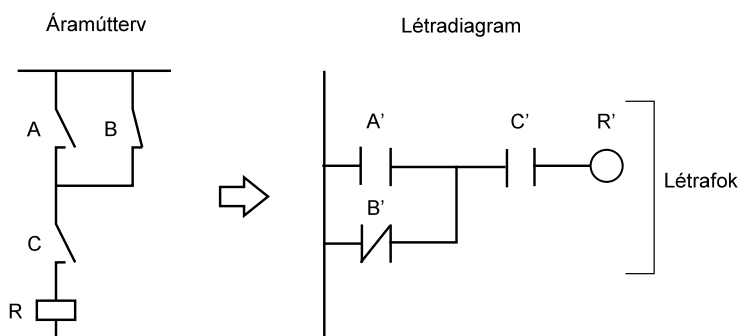
A mellékelt ábra egy példát mutat be áramutas logikai összefüggés létradiagrammos ábrázolására: A és C záró-, B nyitóérintkező, R relé.

A létradiagrammon a vezetékek (logikai vonalak) a baloldali, ún. referenciavezetékéből indulnak ki.

Ezután következnek az érintkezők. Ezek lehetnek a bemenetekkel / kimenetekkel vezérelt, vagy a belső segédrelékhez, tartórelékhez, időrelékhez tartozó záró- ill. bontóérintkezők. A logikai vonal jobboldali végén a kimenetek, időrelék, számlálók stb. "tekercsei", vagy utasítások vannak. Létrafoknak (angolul rung) nevezzük az egy kimenethez tartozó érintkezőket és vezetékeket.

Fontos szabály, hogy a létradiagramban és a hozzátartozó programban az egyes kimenetek, relék, időrelék, számlálók stb. csak egyszer szerepelhetnek. Ezek munkaérintkezői azonban a programban, korlátlan számban felhasználhatók.

A hardverhuzalozott logikai áramkörök és a szoftverben megvalósított PLC létraprogramok működése között lényeges különbségek vannak.



1.1 Mintavételezés, a be- és kimenetek kezelése

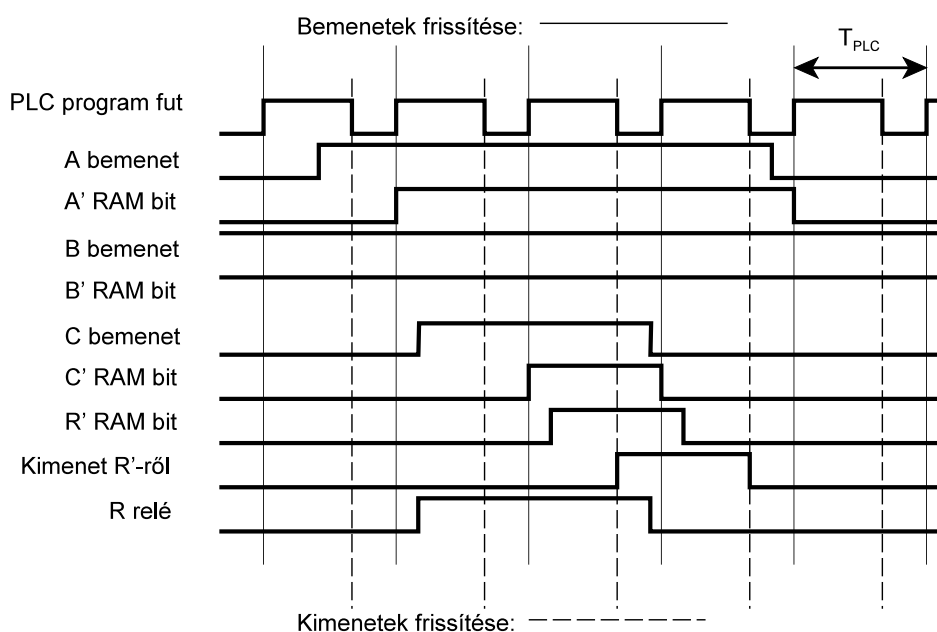
A PLC program T_{PLC} időközönként fut, ahol T_{PLC} -t nevezzük a **mintavételi időnek**.

Maradjunk az előző ábra példájánál. A PLC program, mielőtt az utasítások végrehajtását elkezdene, **mintát vesz** az A, B és C jelekből (PLC bemenetekből) és azokat elteszi a memóriába A', B' és C' címre.

Ezután következik a PLC **program futása**, amely A', B', C' értékéből kiszámítja R' értékét. R' értéke az utasítás végrehajtása során a memóriában kerül eltárolásra R' címen.

Miután a PLC program teljesen lefutott, a **kimenetek** a memóriából **frissülnek**, vagyis R' értéke ekkor íródik ki a RAM-ból a kimenetre.

Ezzel szemben a hardverben megvalósított áramkör azonnal reagál a változásokra. Ez azt jelenti, hogy az A, B, vagy C érintkezők jelei azonnal képesek az R relét aktiválni.



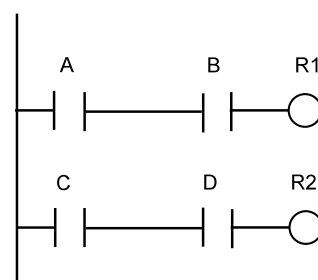
1.2 A PLC program végrehajtási sorrendje

A létradiagrammos PLC program, mint bármely más számítógépes program, sorról sorra, illetve létrafokról létrafokra hajtódik végre.

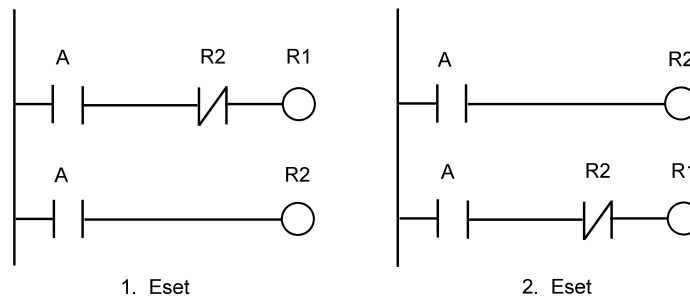
A mellékelt ábrán a PLC először kiszámítja az R1 relé értékét az A és B érintkezők állapotának függvényében (1. létrafok), majd az R2 reléét a C és D érintkezők függvényében. A végrehajtás mindig a leírás sorrendjében történik.

A hardverhuzalozott relésávban ilyen végrehajtási sorrend nincs, minden relé kb. ugyanabban az időpillanatban működik.

Ez a sajátosság azt vonja maga után, hogy a létradiagram egyes esetekben nem feleltethető meg egy az egyben a relésávban le-



rajzoltaknak. Vegyük az alábbi példát.



– **Relésáv esetén:** Ha az ábrán látható két esetet relésávval valósítjuk meg, mindkét eset ugyanúgy működik: ha az “A” érintkező bekapcsol, R1 és R2 relé is meghúz egy pillanatra, majd miután R2 meghúzott R1 elejt.

– **Létradiagram esetén:**

Az 1. esetben, ha az “A” érintkező bekapcsol, R1 bekapcsol, mert R2 nincs húzva. Utána bekapcsol R2 relé is. A következő PLC ciklusban, T_{PLC} idővel később, elejt az R1 relé, mert R2 már húzva van. Tehát T_{PLC} ideig R1 is R2 is húzva vannak.

A 2. esetben, ha az “A” érintkező bekapcsol meghúzza R2 relét, ezért a következő sorban R1 relé már be sem kapcsol, mert R2 húzva van.

A fenti példából látszik, hogy a hardverhuzalozott megoldások csak meggondolás után ültethető át létradiagramos PLC programba.

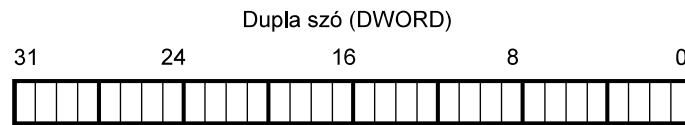
A PLC programban lehetőség van a végrehajtás sorrendjének feltételhez kötött megváltoztatására: ugrásra, szubrutin hívásra.

1.3 A PLC program szerkesztése

A létradiagram alapú PLC programot grafikusán kell szerkeszteni, ezért egy speciális, erre a célra fejlesztett szoftver kell a program írásához. A PLC program szerkesztőjének leírása nem ennek a könyvnek a témája.

2 A PLC program által használt memória

A PLC program által használt memória 10000 dupla szó hosszúságú összefüggő tárterület. Dupla szón (DWORD) egy 32 bitetes memóriaegységet értünk.



A PLC tárterülete 4 fő részre oszlik:

- A PLC program utasításai által állított státuszbitet tartalmazó 0000 című, FLAGS szimbolikus című duplaszóra,
- A PLC program és a ki-, bemeneti hardver egységek, valamint a PLC program és a rendszer közötti kommunikációt végző memóriaterületre,
- A PLC program belső változóit tartalmazó, kikapcsolásra megőrződő, PLCNVRAM szimbolikus címen kezdődő munkaterületre*, és
- A PLC program belső változóit tartalmazó, kikapcsolásra nem megőrződő PLCRAM szimbolikus címen kezdődő munkaterületre.

Az egyes területek címének határai típusonként változhatnak.

*** Figyelem!** Ez a memória terület csak régi típusú, akkumulátoros vezérlőkön kerül automatikusan megőrzésre. Újabb típusokban MW11 utasítással kell eltárolni az adatokat és bekapcsolás után MR10 utasítással kiolvasni azokat.

A PLC memória kiosztása

Szimbólum	Cím	Memória
FLAGS	0000	Státuszbit
		A PLC program és a ki-, bemeneti egységek, valamint a rendszer közötti jelzők
PLCNVRAM		Kikapcsolásra megőrződő PLC munkaterület
PLCRAM		PLC munkaterület
	9999	

A PLC programozónak az utasítások minden változóját is erre a tárterületre kell felvenni. Pl., ha szükség van egy időzítőre a PLC programban, az időzítő változóját, amely az idő múlását méri, ebben a memóriában kell deklarálni.

A státuszbit jelentését ebben a könyvben külön fejezet tárgyalja.

A PLC program és a ki-, bemeneti egységek, valamint a PLC program és a rendszer közötti kommunikációt végző tárterület bitjeinek és regisztereinek címkiosztását és jelentését későbbi fejezetek tárgyalják.

A memória alapegysége a dupla szó (DWORD, 32 bit). Ennél kisebb egység, mint szó (WORD, 16 bit) vagy byte (8 bit), címzésére nincs lehetőség. A memória minden része viszont bitenként elérhető.

A PLC program tag lehetőséget biztosít a memóriához való szimbolikus hozzáférésre. A memória minden területe, amely címezéssel elérhető, elérhető szimbolikusan is.

2.1 Dupla szó (DWORD) címzése

A memória tetszőleges dupla szavára hivatkozhatunk

számmal és

szimbolikusan is.

Számmal történő hivatkozás esetén mindig a **vezető nullák** beírásával 4 decimális számjegyet, kell megadni.

Szimbolikus hivatkozás esetén a szimbólumot előzőleg deklarálni kell a szimbólum tárban.

		Memória
Szimbólum	Cím	
ALMA	0056	
BARACK	0057	
EPER	0058	
MEGGY	0059	

A fenti példa alapján az 56-os memóriarekeszre hivatkozhatunk

0056, vagy

ALMA

beírásával is.

2.2 Dupla szó (DWORD) indexelt címzése a “,” operátorral

Az indexelt címzés két részből áll:

a **báziscím**ből és

az **eltolás**ból.

A két részt a

, (vessző)

operátor választja el egymástól. A cím úgy alakul ki, hogy a báziscímhez hozzáadja az eltolás értékét:

cím=báziscím+eltolás

A **báziscím**re a dupla szó címzésére vonatkozó szabályok érvényesek:

megadható 4 decimális számjegyen, vagy

szimbolikusan.

Az **eltolás értékét** megadhatjuk

közvetlenül számmal, vagy konstans szimbóllummal, vagy

közvetetten regiszterhivatkozással.

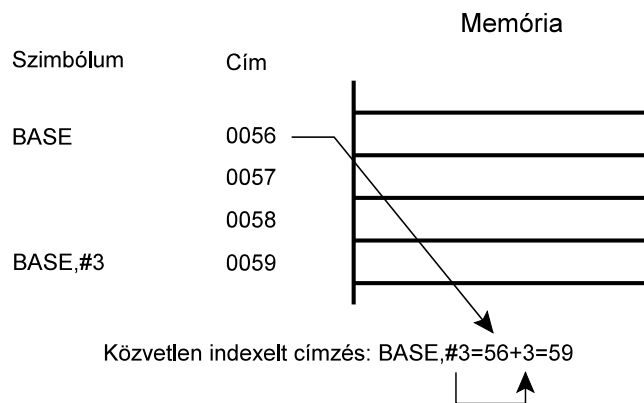
– **Közvetlen**, direkt eltolás megadás esetén a “#” decimális számbevivő operátort kell használni.

Ha a címet 3-mal akarjuk a BASE báziscímhez képest eltolni

0056,#3, vagy

BASE,#3,vagy

hivatkozást kell beírni, feltéve, hogy a BASE szimbólumot a 0056 címhez deklaráltuk.



Ugyanezt az eredményt kapjuk, ha a szimbólumtáblában a BIAS szimbólumot konstans szimbólumnak deklaráljuk: BIAS #3. Ekkor a programban

BASE,BIAS

hivatkozást kell írni.

- **Közvetett**, indirekt, regiszteren keresztüli eltolás megadás esetén a “,” (vessző) operátor után annak a regiszternek a címét kell írni, amelyik az eltolás értékét tartalmazza.

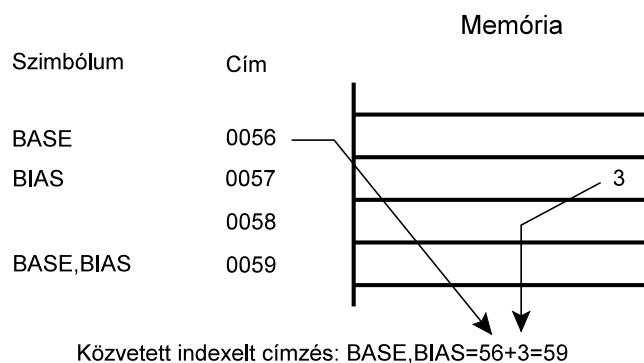
Az eltolást tartalmazó regiszter címére a normál címzési szabályok érvényesek: megadhatók számmal és szimbólummal is.

Ha a BASE báziscímhez deklarálunk egy BIAS regisztert, a címet megadhatjuk:

BASE,BIAS vagy

0056,0057

hivatkozással, feltéve, hogy a BASE szimbólumot a 0056 címhez, a BIAS szimbólumot pedig a 0057 címhez deklaráltuk. Az eltolás értékét a BIAS (0057) címen levő regiszterből veszi.



2.3 Bit közvetlen címezése és a “.” operátor

A PLC memóriaterületének bármely bitjét megcímezhetjük. A bitekre hivatkozhatunk a “.” (pont) operátor használatával, vagy az adott bitre deklarált szimbólummal is.

Ha a **“.”** **operátort** alkalmazzuk, a bites címzés két részből áll:

a dupla szó címéből és

a dupla szón belül a **bitcím**ből

A két részt a

• (pont)

operátor választja el egymástól.

A duplaszó címére hivatkozhatunk 4 decimális számjeggyel, vagy szimbolikusan is.

A bit címére, a “.” operátor után

értékkal,

hivatkozhatunk. A “.” után mindig két számjegyet kell megadni, azaz a vezető nullákat is ki kell írni. A bitcím értéktartománya:

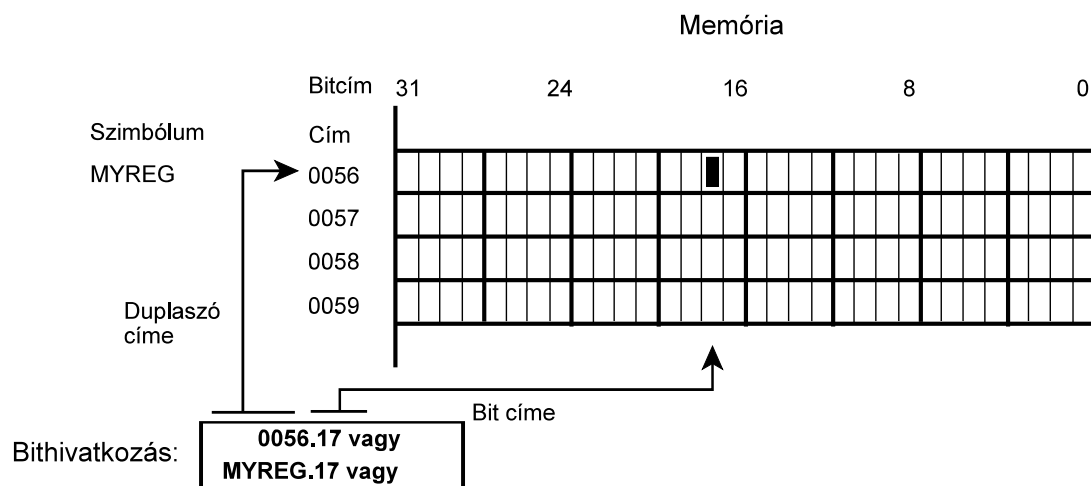
00 ... 31

Ha pl. a 0056 című duplaszó 17-es bitjére akarunk hivatkozni, akkor

0056.17 vagy

MYREG.17

hivatkozást is használhatunk, feltéve, hogy MYREG a 0056 címre lett előzőleg deklarálva.

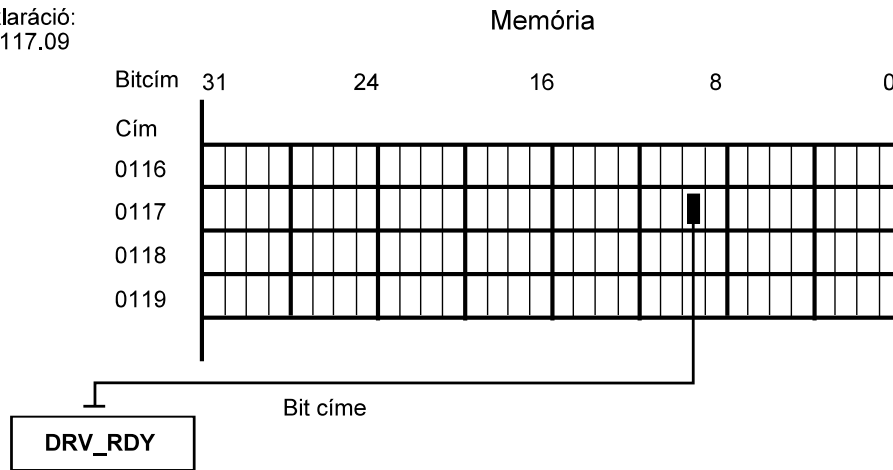


Ha egy bit címére szimbólumot deklaráltunk a **szimbólummal** közvetlenül a bitre hivatkozhatunk.
Ha pl. a DRV_RDY szimbólumot a 0117.09 bitcímre deklaráltuk, akkor a

DRV_RDY

szimbólummal közvetlenül a 0117.09 bitet érhetjük el.

Szimbólum deklaráció:
DRV_RDY 0117.09



2.4 Bit közvetett címzése a “:” operátorral

Ha a “:” operátort alkalmazzuk, a bites címzés két részből áll:

- a **dupla szó címéből** és
- a dupla szón belül a **bitcím**ből

A két részt a

: (kettőspont)

operátor választja el egymástól.

A duplaszó címére hivatkozhatunk

- 4 decimális számjeggyel,
- vagy szimbolikusan is.

A bit címére, a “:” operátor után hivatkozhatunk

- 4 decimális számjeggyel megadott regiszter címmel,
- vagy regiszterre mutató szimbolikus címmel.

Például a

0056:0058

hivatkozás a 0056 regiszter annyiadik bitjére vonatkozik, amennyi a 0058 regiszter tartalma.

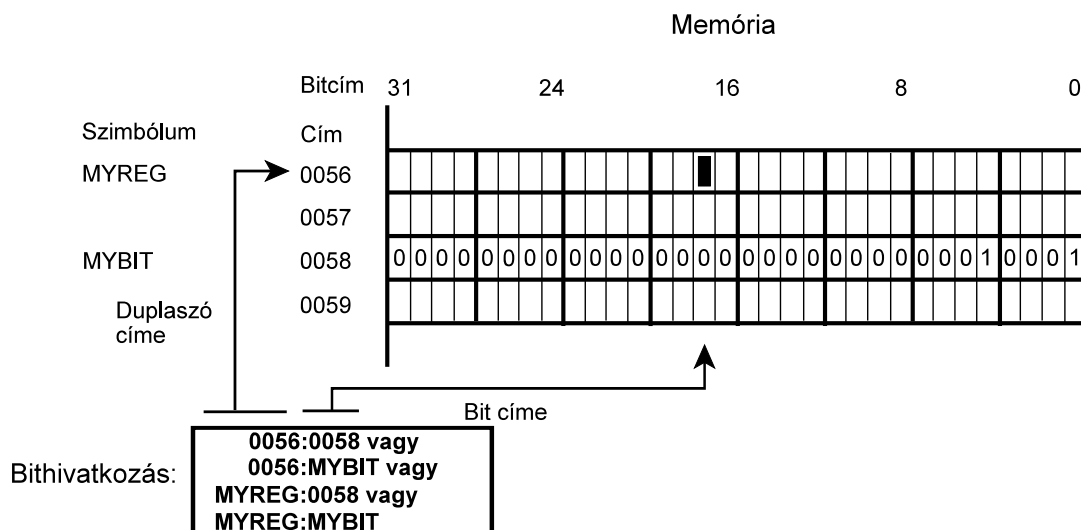
Ha a

MYREG szimbólumot 0056 címre, a

MYBIT szimbólumot pedig a 0058 címre

deklaráltuk, írhatjuk azt is, hogy

MYREG:MYBIT.



2.5 Bit indexelt címzése a “,” operátorral

Bit indexelt címzése hasonló a dupla szóéhoz. A címzés két részből áll:

a **bit báziscímből (bázis.bit)** és

az eltolásból.

A bit báziscím már egy dupla szó (bázis) konkrét bitjére (bit) vonatkozik. Az eltolás értéke mindig a dupla szavak címét tolja el.

A két részt a

, (vessző)

operátor választja el egymástól. A cím úgy alakul ki, hogy a bit báziscímnek a bázis tagjához hozzáadja az eltolás értékét:

bitcím=(bázis+eltolás).bit

A **bit báziscímre** a bit címzésére vonatkozó szabályok érvényesek:

szimbolikusan

megadható “.” pont operátorral, vagy

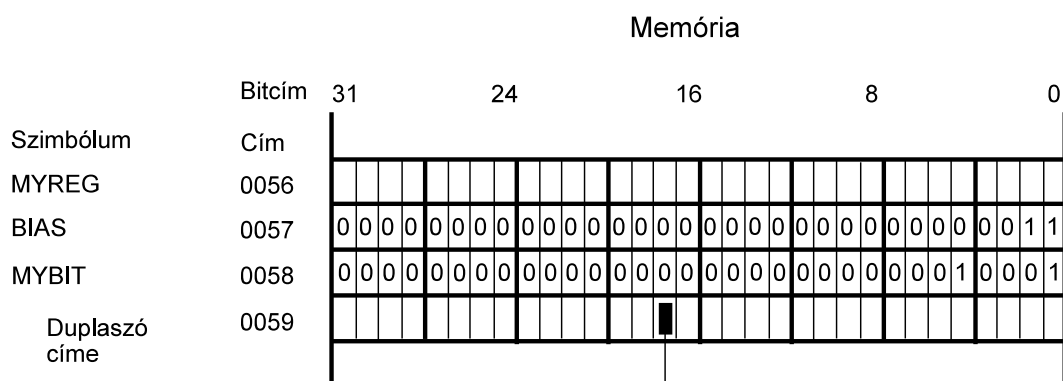
közvetetten “:” operátorral.

Az eltolás értékét megadhatjuk

közvetlenül számmal, vagy

közvetetten regiszterhivatkozással.

Ha a 0056.17 című bitet el akarjuk tolni 3-mal, azt az alábbi ábrán látható módokon tehetjük meg:



Szimbólum deklaráció:
BIT 0056.17

Lehetséges hivatkozások:

```

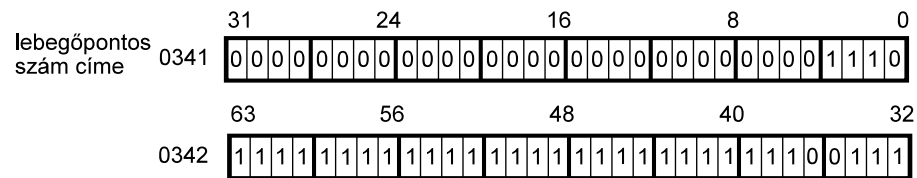
BIT, BIAS
0056.17, #3
MYREG.17, #3
0056:MYBIT, #3
MYREG:MYBIT, #3
BIT, #3
0056.17, BIAS
MYREG.17, BIAS
0056:MYBIT, BIAS
MYREG:MYBIT, BIAS

```

2.6 Lebegőpontos szám (double) címzése

A PLC program utasításai 64 bites lebegőpontos számokat (double) kezelnek. A lebegőpontos számok a memóriaterület két egymást követő regiszterét (DWORD) foglalják el.

Szimbólum: FLOAT 0341



A lebegőpontos számra hivatkozhatunk

számmal és

szimbolikusan is.

Számmal történő hivatkozás esetén mindig a **vezető nullák** beírásával 4 decimális számjegyet, kell megadni.

Szimbolikus hivatkozás esetén a szimbólumot előzőleg deklarálni kell a szimbólum tárban.

Lebegőpontos számot tartalmazó memóriaterületre való hivatkozásnál mindig az első szó címét kell megadni (a lebegőpontos szám alsó 32 bitjét tartalmazza).

A fenti példában a lebegőpontos számra hivatkozhatunk a következőképp:

0341 vagy

FLOAT.

2.7 Lebegőpontos szám (double) indexelt címzése a “,” operátorral

Lebegőpontos számok indexelt címzésére a DWORD regiszterek indexelt címzésénél elmondott szabályok érvényesek.

☞ Figyelem!

Lebegőpontos számokat mindig páros számokkal kell indexelni, mivel két egymást követő regisztert foglalnak el!

Szimbólum	Cím	Memória
BIAS	0055	#4
BASE	0056	Double 1 low
	0057	Double 1 high
	0058	Double 2 low
	0059	Double 2 high
BASE,BIAS	0060	Double 3 low
	0061	Double 3 high

Közvetlen indexelt címzés: $\text{BASE},\#4=56+4=60$

Közvetett indexelt címzés: $\text{BASE,BIAS}=56+4=60$

3 A PLC program moduljai

A PLC program **2 modul**ból áll:

a **főprogram**ból és
az **Int0** modulból.

Mindkét modul egymástól független létrahálózat és önállóan fut. A két modul között kapcsolatot létesíteni a PLC memórián keresztül lehet.

A két modul közötti különbség a futások gyakoriságában van.

3.1 A Főprogram

A **főprogram** T_{PLC} ciklusidővel fut. A ciklusidő értéke típusfüggő. Értéke az NCT2xx vezérlőkön a Diagnosztika ablak PlcPeriod sorából olvasható ki msec-ben (pl: 10msec). Ebben a modulban írjuk le az összes PLC tevékenységet, kivéve azokat, amelyek T_{PLC} ciklusidőnél gyorsabb beavatkozást igényelnek.

3.2 Az Int0 modul

Az **Int0 modul** a T_{PLC} ciklusidőnél nagyobb gyakorisággal fut. Értéke az NCT2xx vezérlőkön a Diagnosztika ablak TimeSlice sorából olvasható ki μ sec-ben (pl: 2000 μ sec). Ebben a modulban csak azokat a PLC tevékenységeket szabad leírni, amelyek gyors beavatkozást igényelnek: például egy interfész bemenő jelre gyorsan be kell kapcsolni egy interfész kimenetet. Ha nagyon túlterheljük a modult, PLC időn túl hibaüzenetet kapunk.

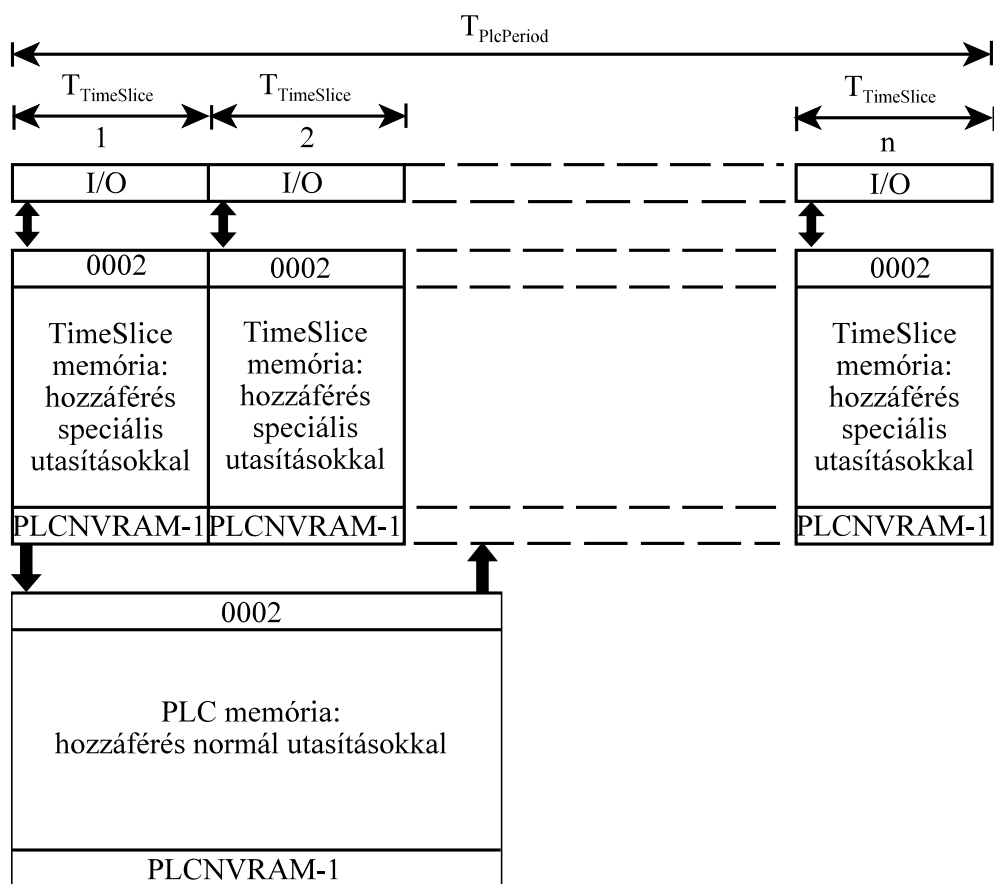
3.3 A PLC memória frissítése

Az interfész be- és kimenetek, valamint a PLC program és a rendszer közötti kommunikációt végző memóriaterület az **Int0** modul, azaz a **TimeSlice** idő **gyakoriságával** kerülnek beolvasásra illetve kiírásra. Ez a memóriaterület a 0002 címtől a PLCNVRAM-1 címig tart.

A fizikai bemenetek (pl. interfész bemenő jelek) beolvasásra kerülnek a RAM-ba, a fizikai kimeneteket (pl. interfész kimenő jeleket) pedig kiírja a RAM-ból a hardverre. Ezt a TimeSlice gyakorisággal frissítésre kerülő RAM területet, a **TimeSlice memóriát**, a PLC program csak **speciális utasítások** segítségével tudja írni, illetve olvasni. Ezeket az utasításokat csak az Int0 modulban van értelme használni!

A normál PLC utasítások segítségével hozzáférhető **PLC bemeneti memóriaterület a PLC főprogram indulása előtt** a TimeSlice memóriából kerül frissítésre, azaz **szinkronizálásra**. A PLC Főprogram több TimeSlice időszeleten keresztül futhat, így a futása során időben összetartozó bemeneti képből tud dolgozni. A PLC program által normál utasításokkal kapcsolt **kimenetek a PLC főprogram lefutása után** kerülnek **frissítésre** a TimeSlice memóriában.

Mindkét modul, azaz a Főprogram és az Int0 modul normál utasításai ugyanazt a szinkronizált memóriaterületet látják.



4 A PLC program által kezelt adatok

A PLC program fixpontos, lebegőpontos és bites adatokat tud kezelni.

A memóriában tárolt fixpontos adatok lehetnek bináris és BCD (binárisan kódolt, decimális) formában is.

A bites adatokat lekérdezhetjük változásvizsgálatra is.

4.1 Bites adatok kezelése

A PLC program leggyakrabban használt adategysége a bites adat. A PLC teljes memóriaterületének bármely bitjét lekérdezhetjük, azaz érintkezőt definiálhatunk a bitcímre, és bármely kimeneti bitjét beírhatjuk, azaz pl. relétekercsként felhasználhatjuk. Ezen kívül a teljes memóriaterület bármely bitjére változásvizsgálatot indíthatunk.

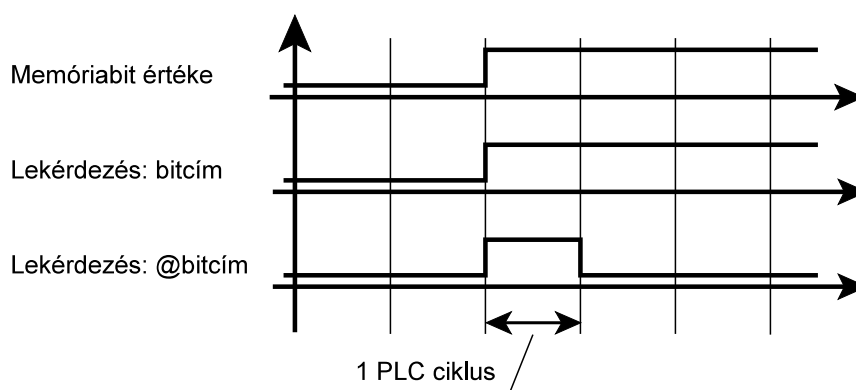
4.2 Memóriabit felfutó élének lekérdezése a @ operátorral

A PLC program által elérhető memória bármely bitjének 0-ból 1-be történő változása lekérdezhető, ha a memóriabit címe elé a @ operátort írunk:

@bitcím: felfutó él lekérdezése

A bitcím megadása az összes lehetséges módon történhet: közvetlenül, vagy közvetetten, szimbolikusan, vagy indexelve is.

Az így lekérdezett adat 1 PLC ciklus erejéig lesz IGAZ:



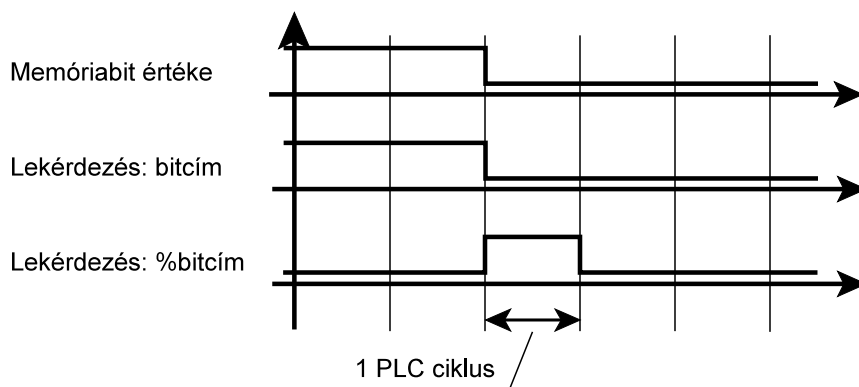
4.3 Memóriabit lefutó élének lekérdezése a % operátorral

A PLC program által elérhető memória bármely bitjének 1-ből 0-ba történő változása lekérdezhető, ha a memóriabit címe elé a % operátort írunk:

%bitcím: lefutó él lekérdezése

A bitcím megadása az összes lehetséges módon történhet: közvetlenül, vagy közvetetten, szimbolikusan, vagy indexelve is.

Az így lekérdezett adat 1 PLC ciklus erejéig lesz IGAZ:



4.4 Bemenet azonnali lekérdezése, kimenet azonnali kiadása a ! operátorral

A PLC program által elérhető memória bármely bite azonnal lekérdezhető a TimeSlice memóriából, illetve azonnal kiírható a TimeSlice memóriába, ha a memóriabit címe elé a ! operátort írunk:

!bitcím: TimeSlice memória bites olvasása, írása

A bitcím megadása az összes lehetséges módon történhet: közvetlenül, vagy közvetetten, szimbolikusan, vagy indexelve is. Az Int0 modulban gyors reakciót követelő beavatkozásokhoz célszerű használni.

☞ **Figyelem!** A ! operátor nem használható együtt a @, vagy a % élképző operátorral!

4.5 Decimális, előjeles szám megadása a # operátorral

A # operátorral decimális, előjeles konstans adható meg a

$$-2147483648 \leq \text{konstans} \leq 2147483647$$

tartományban. A PLC szerkesztő hibát jelez, ha a bevitt érték a megadott határon kívül esik.

A bevitt szám a tárba **binárisan, 2-es komplementben** ábrázolva kerül be.

A 31-es bit az előjelbit. A + (pozitív) előjelet nem kell kitenni.

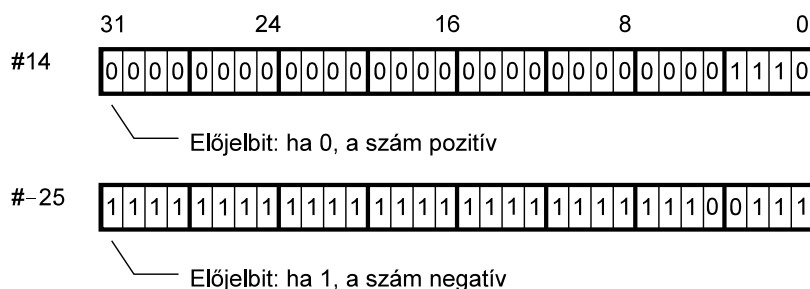
Tehát, ha +14-et akarunk a tárba írni

#14-et,

ha -25-öt, akkor

#-25-öt

kell írni.



4.6 Hexadecimális szám megadása a #\$ operátorral

A #\$ operátorral egy max. 8 számjegyű hexadecimális konstans adható meg a

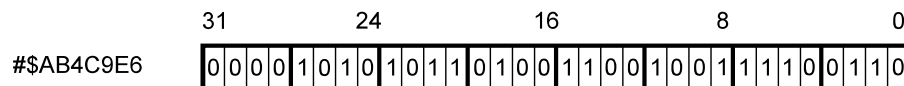
$$0 \leq \text{konstans} \leq \text{FFFFFFFF}$$

tartományban. A PLC szerkesztő hibát jelez, ha a bevitt érték a megadott határon kívül esik. A vezető nullák elhagyhatók.

Például, ha 0AB4C9E6 hexadecimális értéket kívánunk a tárba írni

#\$AB4C9E6-ot

kell írni.



4.7 Előjel nélküli, BCD szám megadása a #\$ operátorral

A #\$ operátorral egy max. 8 számjegyű BCD (binárisan kódolt decimális), előjel nélküli konstans adható meg a

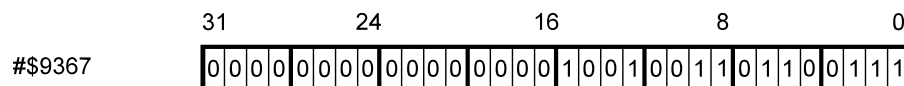
$$0 \leq \text{konstans} \leq 99999999$$

tartományban. Tehát, ha BCD adatot kívánunk bevinni a tárba, azt hexadecimális számként kezeljük, de csak a 0, 1, ..., 9 számjegyeket használjuk. A vezető nullák elhagyhatók.

Például, ha a 9367 decimális számot BCD kódolásban kívánjuk a tárba írni

#\$9367-et

kell írni.



4.8 Lebegőpontos szám megadása a * operátorral

A PLC program lebegőpontos szám ábrázolása az IEEE754-es szabvány dupla pontosságú, lebegőpontos számábrázolását követi. Ezeket a számokat 64 biten ábrázoljuk. Tehát, ha egy lebegőpontos adatot kívánunk a tárba írni, mindig

két duplaszó (64 bit)

helyet kell lefoglalni. Ha a lebegőpontos adatot az n címre írjuk,

az n és az n+1 címen

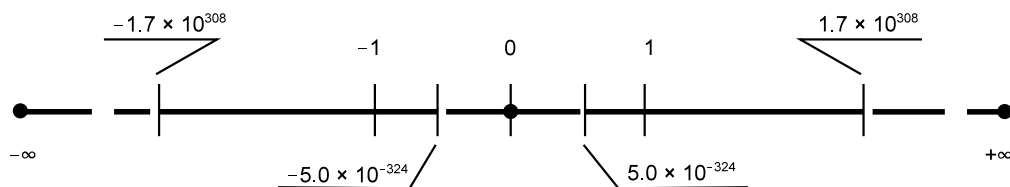
fog az adat elhelyezkedni. Lebegőpontos adatot emiatt a 9999 címre tilos írni!

A * operátorral kb. a

$$\pm 5.0 \times 10^{-324} \text{ -től a } \pm 1.7 \times 10^{308} \text{ -ig}$$

terjedő nagyságú lebegőpontos számokat és a 0-t lehet ábrázolni, 15-16 számjegy pontossággal.

A bevétel során **tizedespontot** (.) kell használni.



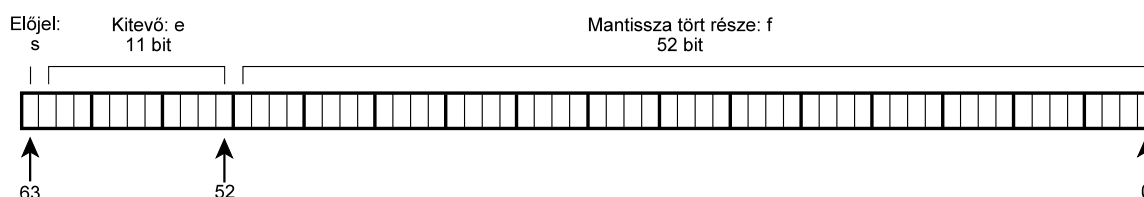
Ha például a -124.753 -as számot kívánjuk bevinni

* -124.753 -at

kell beírni. Sem vezető, sem követő nullákat nem kell beadni. A + (pozitív) előjel elhagyható.

4.9 Az IEEE754 szabvány dupla pontosságú lebegőpontos számábrázolása

Tájékoztatásul röviden leírjuk az IEEE754 dupla pontosságú lebegőpontos formátum (double precision floating point format) felépítését, ám *ezzel a PLC programozónak egyáltalán nem kell foglalkoznia.*



A dupla pontosságú lebegőpontos szám három részből áll:

- **Előjelbit (s):** ha értéke 0 a szám pozitív, ha 1 a szám negatív.
- **Kitevő** vagy **exponens (e):** 11 bit hosszúságú, és az alapja 2. Ezen a mezőn pozitív és negatív kitevőket is ábrázolni kell. Hogy ezt elérjük, egy **eltolást** adunk a tényleges kitevőhöz, amelynek értéke **1023**. Tehát, ha egy eltolt kitevő értéke $e=1201$, a tényleges kitevőt az $1201 - 1023 = 178$ összefüggésből kapjuk. A csupa 0 (000h) és a csupa 1 (7FFh) exponens érték speciális használatra van fenntartva.
- **Mantissza:** A mantissza 53 bit hosszúságú. Két részből tevődik össze: egy **egybitese egész részből** és egy **tört részből (f)**, amely 52 bit hosszúságú. Az egész rész értéke 1 normalizált számok, illetve 0 denormalizált számok esetén. Következésképp a mantissza 53 bites, de csak 52 bit helyet foglal a számábrázolás során, mert csak a tört részét ábrázoljuk.

A kitevő (e) és a mantissza tört részének (f) értéke hatással van a számábrázolásra:

- **Normalizált szám:** ha a kitevő nem csupa 0 ($e \neq 000h$) és nem csupa 1 ($e \neq 7FFh$) a szám normalizált. Ekkor a mantissza egész részére 1-t tételezünk fel és a számot az alábbi összefüggésből kapjuk:

$$(-1)^s \times 1.f \times 2^{e-1023}, \text{ ahol "s" az előjelbit, "f" a mantissza tört rész, "e" a kitevő.}$$

Így az ábrázolható legnagyobb abszolút érték, $1.f = 2 - 2^{-52}$ -t behelyettesítve:

$$(-1)^s \times (2 - 2^{-52}) \times 2^{1023} \approx \pm 1.7 \times 10^{308}$$

- **Túlsordulás:** Ha egy lebegőpontos művelet eredménye nem ábrázolható, mert a kapott szám abszolút értéke meghaladja az ábrázolható maximumot, a művelet beírja az **FL_OF** túlsordulás (overflow) státuszbitet.
- **Denormalizált szám:** ha a kitevő csupa 0 ($e=000h$) de a mantissza tört része nem 0, $f \neq 0$, a

szám denormalizált. Ezt a formátumot nagyon kis számok ábrázolására használjuk. Ekkor a mantissza egész részére 0-t tételezünk fel és a számot az alábbi összefüggésből kapjuk:

$$(-1)^s \times 0.f \times 2^{-1022}, \text{ ahol "s" az előjelbit, "f" a mantissza tört része.}$$

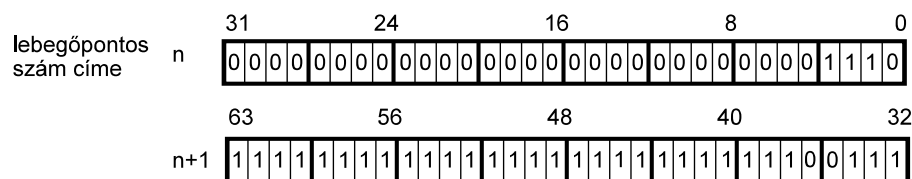
Így az ábrázolható legkisebb abszolút érték, $0.f=2^{-52}$ -t behelyettesítve:

$$(-1)^s \times 2^{-52} \times 2^{-1022} = (-1)^s \times 2^{-1074} \approx \pm 5.0 \times 10^{-324}$$

- **Alulcsordulás:** Ha a lebegőpontos művelet eredménye nem ábrázolható, mert a kapott szám abszolút értéke kisebb, mint az ábrázolható minimum, a művelet beírja az **FL_UF** alulcsordulás (underflow) státuszbitet.
- **Nulla:** az a szám nulla, amelynek az exponense is és a mantissza tört része is 0:
 $e=0$ és $f=0$
 A szabvány az előjelbit függvényében ($s=0$, vagy $s=1$) megkülönböztet $+0$ -t és -0 -t. A nulla tehát egy speciális, denormalizált szám.
- **Végtelen:** Ha a kitevő csupa 1 és a mantissza tört része 0
 $e=2047(=7FFh)$ és $f=0$
 a szám $\pm\infty$, az s előjelbit függvényében.
- **Nem Szám, NaN (Not a Number):** akkor nem tekintjük érvényes számnak az ábrázolt értéket, ha ha a kitevő csupa 1 és a mantissza tört része nem 0:
 $e=2047(=7FFh)$ és $f \neq 0$
- **Hibajelzés Nem Szám értékre:** Ha a lebegőpontos műveletben részt vevő valamelyik szám Nem Szám (NaN), pl. egy elmaradt inicializálás következtében, a vezérlő beírja az **FL_ER** státuszbitet.

Lebegőpontos szám elhelyezkedése a memóriában

A lebegőpontos szám alsó 32 bitje a megadott címen helyezkedik el, míg a felső 32 bitje a megadott cím utáni dupla szót foglalja el:



5 A PLC utasítások által állított státuszbiték

A PLC utasítások által állított jelzők valamilyen művelet eredményének, vagy hibájának jelzésére, státuszára szolgálnak.

A státuszbiték a 0000 fizikai, FLAGS szimbolikus című duplaszón találhatók. Erre a szóra a PLC programból **csak olvasással hivatkozhatunk, írni nem szabad.**

5.1 Az FL_ER (error) hiba jelző

A rendszer a jelzőt 1-be írja, ha a végrehajtott PLC utasításban hibát talál. Ezek a következők lehetnek:

- a megadott cím kívül esik a 0000....9999 értékhatáron,
- a megadott adat kívül esik a lehetséges értékhatáron,
- a megadott adat nem jó formátumban van,
- lebegőpontos művelet egyik adata NaN (Nem Szám)
- az utasítás bemenő paraméterei hibásak,
- az utasítás nem hajtható végre.

Az egyes utasítások tárgyalásánál külön foglalkozunk az FL_ER jelző kezelésével.

5.2 Az FL_UF (underflow) alulcsordulás jelző

A rendszer a jelzőt 1-be írja, ha:

- ha két negatív szám fixpontos összeadása után keletkező összeg a 00000000 ... 07FFFFFF pozitív számok tartományába esik, egyébként 0,
- ha egy negatív számból egy pozitív számot fixpontosan kivonva a különbség a 00000000 ... 07FFFFFF pozitív számok tartományába esik, egyébként 0,
- ha lebegőpontos műveleteknél az eredmény abszolút értéke olyan kicsi, hogy nem ábrázolható dupla pontosságú lebegőpontos számábrázolással.

5.3 Az FL_OF (overflow) túlcsordulás jelző

A rendszer a jelzőt 1-be írja, ha:

- ha két pozitív szám fixpontos összeadása után keletkező összeg a 80000000...FFFFFFFF negatív számok tartományába esik, egyébként 0,
- ha egy pozitív számból egy negatív számot fixpontosan kivonva a különbség a 80000000 ... FFFFFFFF negatív számok tartományába esik, egyébként 0,
- ha lebegőpontos műveleteknél az eredmény abszolút értéke olyan nagy, hogy nem ábrázolható dupla pontosságú lebegőpontos számábrázolással.

5.4 Az FL_CY (carry) átvitel jelző

A rendszer a jelzőt 1-be írja, ha:

- ha fixpontos összeadás során átvitel képződik, vagyis az összeg nem fér el 32 biten,
- ha fixpontos kivonás során kölcsön képződik, egyébként 0.

5.5 Az FL_GT (greater than) nagyobb, mint jelző

A rendszer a jelzőt 1-be írja,

- ha két szám összehasonlítása során a bal oldali érték nagyobb, mint a jobboldali: $A > B$.

5.6 Az FL_EQ (equal) egyenlő jelző

A rendszer a jelzőt 1-be írja,

- ha két szám összehasonlítása során a két szám egyenlő: $A = B$.

5.7 Az FL_LT (lower than) kisebb, mint jelző

A rendszer a jelzőt 1-be írja,

- ha két szám összehasonlítása során a bal oldali érték kisebb, mint a jobboldali: $A < B$.

6 A PLC program utasításai

6.1 Bitműveleteket végző utasítások

A bitműveleteket végző utasítások két értékkel, 0-val és 1-gyel dolgoznak. A 0 értéket HAMIS, az 1 értéket IGAZ állapotnak tekintjük. A bites utasításokkal különböző Bool algebrai műveletek valósíthatók meg, mint az ÉS, a VAGY és a KIZÁRÓ VAGY. A logikai műveletek bemenetei mindig a memória megfelelő bitjeire vonatkoznak, a műveletek eredményei pedig a memória megfelelő bitjeit írják.

6.1.1 Záróérintkező: memóriabit lekérdezése

A záróérintkező szimbóluma:

A záróérintkező bemenő paramétere:

– **Address of Bit:**

Értékhatár (n.i): a duplaszó címére: n=0000...9999, a bitcímre: i=00..31

Bitcím számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

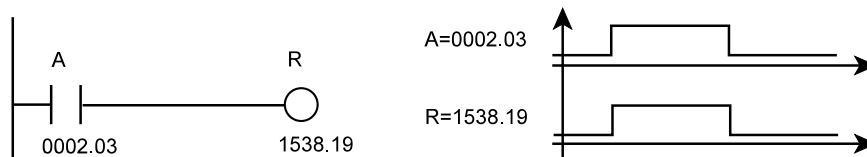
Lehetséges módosító tagok:

cím előtt: @ felfutó él vagy % lefutó él kezelése,

cím után: „,” inxelési operátor.

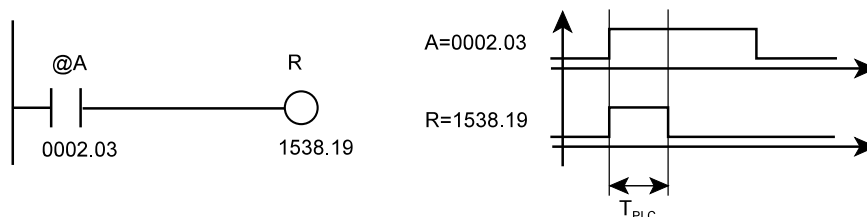
A záróérintkező működése:

A záróérintkezőn áram folyik (az érintkező zárva van), ha a memóriabit értéke IGAZ (1), nem folyik áram (az érintkező nyitva van), ha a memóriabit értéke HAMIS (0). Az érintkező után kapcsolt relétekercs ennek megfelelően behúz, vagy elenged.



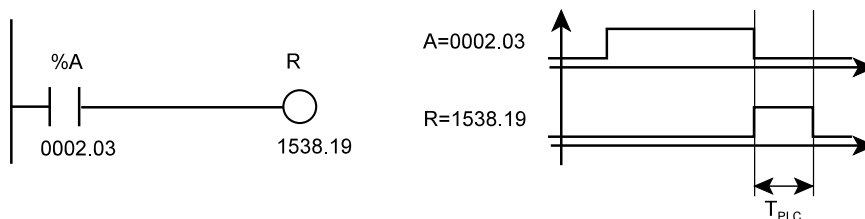
A záróérintkező működése a felfutó él @ operátorral

Ha a záróérintkező címe elé az @ operátort írjuk, a záróérintkezőn csak a jel felfutó éle (a memóriabit 0-ból 1-be váltása) után 1 PLC ciklus ideig folyik áram, azaz az R relé T_{PLC} ideig van húzva.



A záróérintkező működése a lefutó él % operátorral

Ha a záróérintkező címe elé a % operátort írjuk, a záróérintkezőn csak a jel lefutó éle (a memória-bit 1-ből 0-ba váltása) után 1 PLC ciklus ideig folyik áram, azaz az R relé T_{PLC} ideig van húzva.

A záróérintkező működése az azonnali lekérdezés ! operátorral

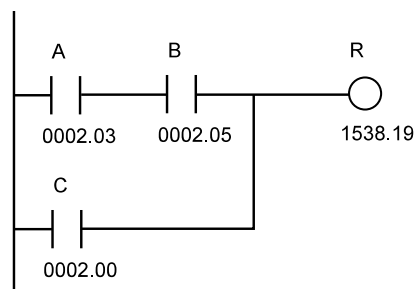
Ha a záróérintkező címe elé a ! operátort írjuk, a záróérintkező állapotát a TimeSlice memóriából veszi.

Példa záróérintkezők alkalmazására

A sorba kötött záróérintkezőkkel logikai ÉS kapcsolat valósítható meg, míg a párhuzamos kötésekkkel logikai VAGY.

A mellékelt ábra az alábbi logikai kapcsolatra mutat példát:

$$(A \text{ AND } B) \text{ OR } C = R$$

**6.1.2 Záróérintkező: dupla szó lekérdezése**

A záróérintkező szimbóluma:

A záróérintkező bemenő paramétere:

– **Address of Bit:**

Értékhatár: $n=0000\dots9999$

Duplaszó címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Lehetséges módosító tagok:

cím előtt: @ felfutó él vagy % lefutó él kezelése,

cím után: „,” inxelési operátor.

A záróérintkező működése:

A záróérintkezőn áram folyik (az érintkező zárva van), ha a dupla szó értéke >0 , nem folyik áram (az érintkező nyitva van), ha a dupla szó értéke $=0$. Az érintkező után kapcsolt relétekercs ennek megfelelően behúz, vagy elenged.

Az utasítás időzítők, számlálók stb. vizsgálatára használható. Például annak eldöntésére, hogy az időzítő még jár-e (>0), azaz az érintkezőn áram folyik, vagy már lejárt ($=0$), azaz az érintkezőn nem folyik áram.

6.1.3 Nyitóérintkező: memóriabit negált lekérdezése

A nyitóérintkező szimbóluma:

A nyitóérintkező bemenő paramétere:

– **Address of Bit:**

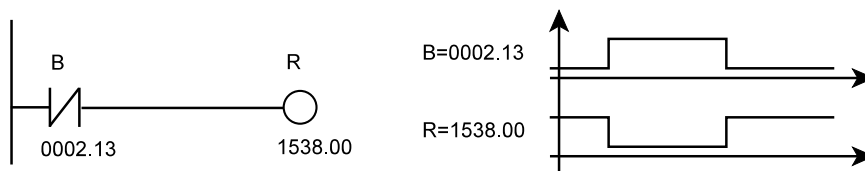
Értékhatár (n.i): a duplaszó címére: n=0000...9999, a bitcímre: i=00..31

Bitcím számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Lehetséges módosító tagok:

cím előtt: @ felfutó él vagy % lefutó él kezelése,

cím után: “,” inxelési operátor.

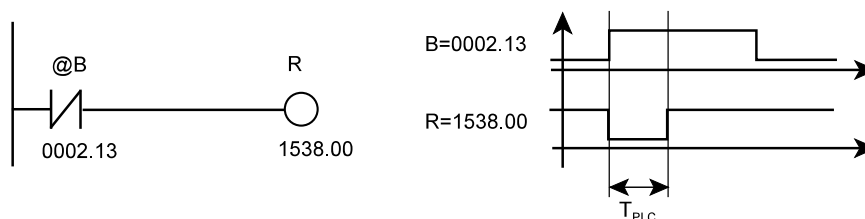


A nyitóérintkező működése:

A nyitóérintkezőn áram folyik (az érintkező zárva van), ha a memóriabit értéke HAMIS (0), nem folyik áram (az érintkező nyitva van), ha a memóriabit értéke IGAZ (1). Az érintkező után kapcsolt relétekercs ennek megfelelően elenged, vagy behúz.

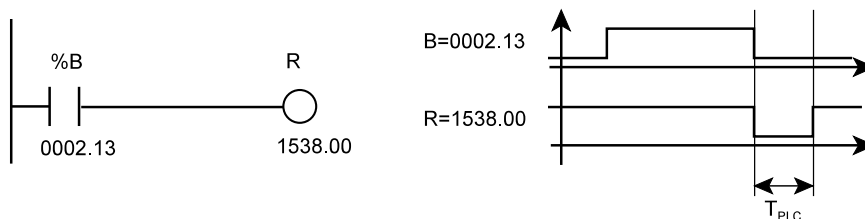
A nyitóérintkező működése a felfutó él @ operátorral

Ha a nyitóérintkező címe elé az @ operátort írjuk, a nyitóérintkezőn csak a jel felfutó éle (a memóriabit 0-ból 1-be váltása) után 1 PLC ciklus ideig nem folyik áram, azaz az R relé T_{PLC} ideig ejt ki.



A nyitóérintkező működése a lefutó él % operátorral

Ha a nyitóérintkező címe elé a % operátort írjuk, a nyitóérintkezőn csak a jel lefutó éle (a memóriabit 1-ből 0-ba váltása) után 1 PLC ciklus ideig nem folyik áram, azaz az R relé T_{PLC} ideig ejt ki.

A nyitóérintkező működése az azonnali lekérdezés ! operátorral

Ha a nyitóérintkező címe elé a ! operátort írjuk, a nyitóérintkező állapotát a TimeSlice memóriából veszi.

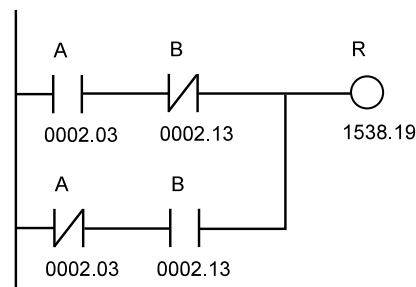
Példa nyitóérintkezők alkalmazására

Nyitó és záróérintkezők alkalmazásával KIZÁRÓ VAGY kapcsolat valósítható meg.

Mint ismeretes a kizáró vagy kapcsolat felírható a következő alakban:

$$A \text{ XOR } B = (A \text{ AND } (\text{NOT } B)) \text{ OR } ((\text{NOT } A) \text{ AND } B)$$

A mellékelt ábra ennek az összefüggésnek a megvalósítását mutatja nyitó- és záróérintkezőkkel.

**6.1.4 Nyitóérintkező: dupla szó negált lekérdezése**

A nyitóérintkező szimbóluma:

A nyitóérintkező bemenő paramétere:

– **Address of Bit:**

Értékhatár: $n=0000\dots9999$

Duplaszó címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Lehetséges módosító tagok:

cím előtt: @ felfutó él vagy % lefutó él kezelése,

cím után: „,” inxelési operátor.

A nyitóérintkező működése:

A nyitóérintkezőn áram folyik (az érintkező zárva van), ha a dupla szó értéke=0, nem folyik áram (az érintkező nyitva van), ha a dupla szó értéke >0. Az érintkező után kapcsolt relétekercs ennek megfelelően elenged, vagy behúz.

Az utasítás időzítők, számlálók stb. vizsgálatára használható. Például annak eldöntésére, hogy az időzítő még jár-e (>0), azaz az érintkezőn nem folyik áram, vagy már lejárt (=0), azaz az érintkezőn áram folyik.

6.1.5 Relétekercs: memóriabit írása

A relétekercs szimbóluma: 

A relétekercs bemenő paramétere:

– **Address of Bit:**

Értékhatár (n.i): a duplaszó címére: n=0000...9999, a bitcímre: i=00..31

Bitcím számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Lehetséges módosító tagok:

cím után: “,” inxelési operátor.

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a létrafok kommentje.

A relétekercs működése az azonnali kiadás ! operátorral

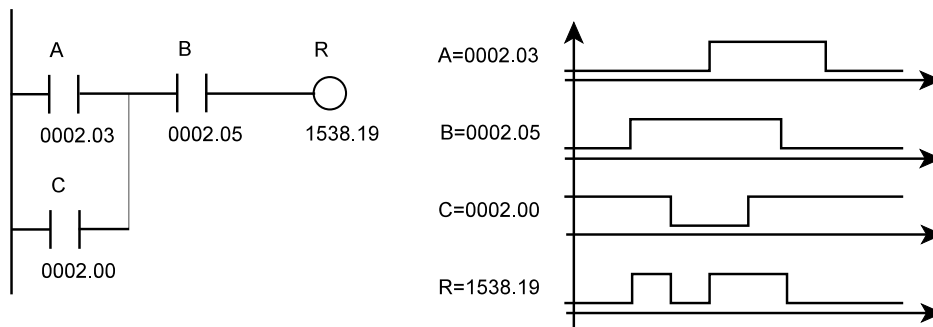
Ha a relétekercs címe elé a ! operátort írjuk, a relétekercs állapotát a TimeSlice memóriába adjuk ki.

A relétekercs működése

Ha a relétekercs bemenetébe “áram” folyik, vagyis, ha a tekercs előtti feltételek összessége IGAZ állapotú a relé meghúz, azaz a tekercshez tartozó memóriabitet az utasítás 1-be írja.

Fordítva, ha a relétekercs bemenetébe nem folyik “áram”, vagyis, ha a tekercs előtti feltételek összessége HAMIS állapotú a relé kiejt, azaz a tekercshez tartozó memóriabitet az utasítás 0-ba írja.

A relétekercs ú.n. záróelem, azaz nincs kimenete, utána már nem köthető semmi.



6.1.6 Negált relétekercs: memóriabit negált írása

Az negált relétekercs szimbóluma: 

Az negált relétekercs bemenő paramétere:

– **Address of Bit:**

Értékhatár (n.i): a duplaszó címére: n=0000...9999, a bitcímre: i=00..31

Bitcím számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Lehetséges módosító tagok:

cím után: “,” inxelési operátor.

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a létrafok kommentje.

A negált relétekercs működése az azonnali kiadás ! operátorral

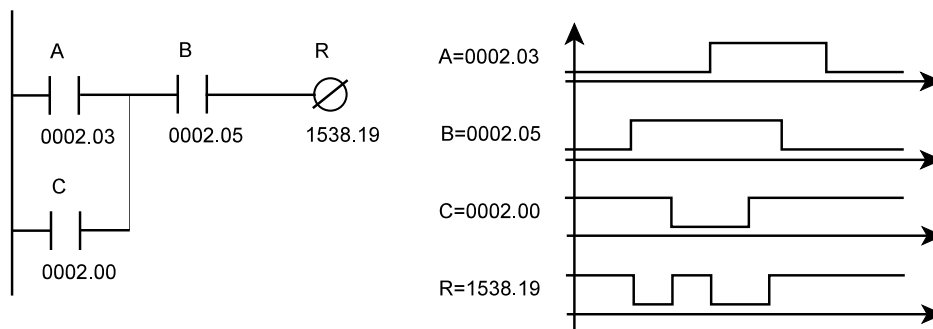
Ha a negált relétekercs címe elé a ! operátort írjuk, a negált relétekercs állapotát a TimeSlice memóriába adja ki.

Az negált relétekercs működése

Ha a negált relétekercs bemenetében “áram” folyik, vagyis, ha a tekercs előtti feltételek összessége IGAZ állapotú a relé kiejt, azaz a tekercshez tartozó memóriabitet az utasítás 0-ba írja.

Fordítva, ha a relétekercs bemenetében nem folyik “áram”, vagyis, ha a tekercs előtti feltételek összessége HAMIS állapotú a relé meghúzza, azaz a tekercshez tartozó memóriabitet az utasítás 1-be írja.

A negált relétekercs ú.n. záróelem, azaz nincs kimenete, utána már nem köthető semmi.



6.1.7 Memória bitjének beállítása: a SET utasítás

A SET utasítás szimbóluma: 

A SET utasítás bemenő paramétere:

– **Address of Bit:**

Értékhatár (n.i): a duplaszó címére: n=0000...9999, a bitcímre: i=00..31

Bitcím számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Lehetséges módosító tagok:

cím után: “,” inxelési operátor.

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a létrafok kommentje.

A SET utasítás működése az azonnali kiadás ! operátorral

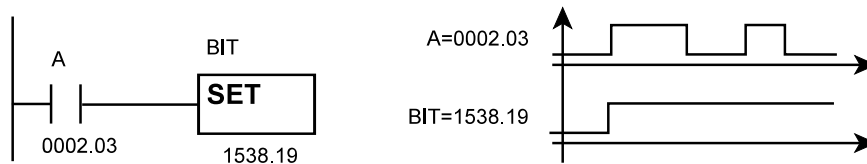
Ha a SET utasítás címe elé a ! operátort írjuk, a bitet a TimeSlice memóriába írja be.

A SET utasítás működése

Ha az utasítás előtti feltételek összessége IGAZ állapotba kapcsolódik, a vonatkozó memóriabitet az utasítás 1-be írja.

Ha ezután az utasítás előtti feltételek összessége HAMIS állapotúra vált a memóriabit változatlan marad.

A SET utasítás ú.n. záróelem, azaz nincs kimenete, utána már nem köthető semmi.



6.1.8 Memória bitjének törlése: az RST utasítás

Az RST utasítás szimbóluma:

Az RST utasítás bemenő paramétere:

– **Address of Bit:**

Értékhatar (n.i): a duplaszó címére: n=0000...9999, a bitcímre: i=00..31

Bitcím számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Lehetséges módosító tagok:

cím után: “,” inxelési operátor.

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a létrafok kommentje.

Az RST utasítás működése az azonnali kiadás ! operátorral

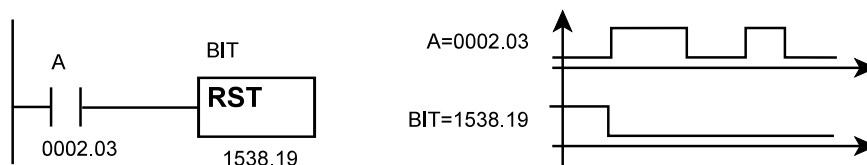
Ha az RST utasítás címe elé a ! operátort írjuk, a bitet a TimeSlice memóriában törli.

Az RST utasítás működése

Ha az utasítás előtti feltételek összessége IGAZ állapotba kapcsolódik, a vonatkozó memóriabitet az utasítás 0-ba írja.

Ha ezután az utasítás előtti feltételek összessége HAMIS állapotúra vált a memóriabit változatlan marad.

Az RST utasítás ú.n. záróelem, azaz nincs kimenete, utána már nem köthető semmi.



6.1.9 Impulzusképzés felfutó élre: a DIFU utasítás

A DIFU utasítás szimbóluma: 

A DIFU utasítás bemenő paramétere:

– **Address of Bit:**

Értékhatár (n.i): a duplaszó címére: n=0000...9999, a bitcímre: i=00..31

Bitcím számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Lehetséges módosító tagok:

cím után: “,” inxelési operátor.

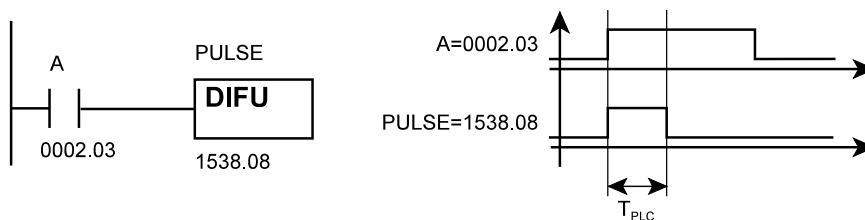
– **Remark:**

Megjegyzés: a Remark ~~szöveg~~ írt szöveg lesz a létrafok kommentje.

A DIFU utasítás működése

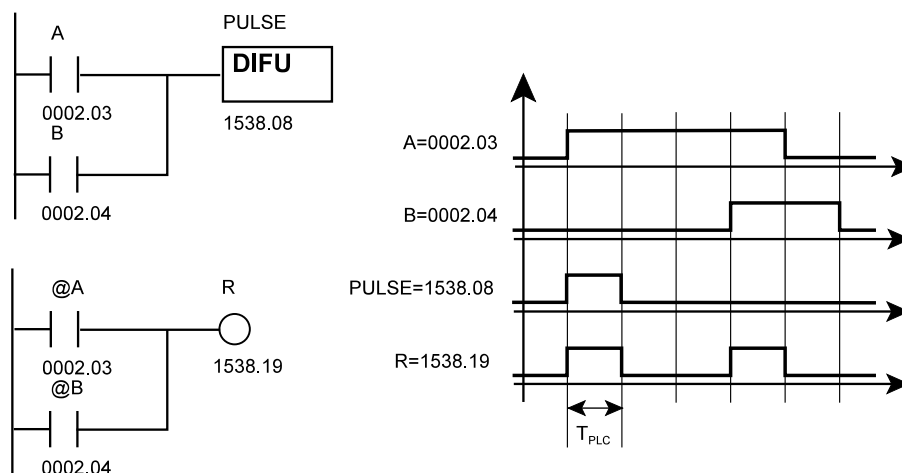
Ha az utasítás előtti feltételek összessége HAMIS (0) állapotból IGAZ (1) állapotba vált, a vonatkozó memóriabitet az utasítás 1-be írja 1 PLC ciklus (T_{PLC} idő) erejéig, vagyis a jel felfutó élére differenciál.

A DIFU utasítás ú.n. záróelem, azaz nincs kimenete, utána már nem köthető semmi.



A DIFU utasítás alkalmazása

A DIFU utasítás és az @ operátor alkalmazásának különbségére hívja fel a figyelmet az alábbi példa. Ha az (A OR B) összefüggés eredményét differenciáljuk DIFU utasítással, más eredményt kapunk, mintha az A felfutó élet vagyoljuk a B felfutó élével és az eredményt az R relén tároljuk.



6.1.10 Impulzusképzés lefutó élre: a DIFD utasítás

A DIFD utasítás szimbóluma: 

A DIFD utasítás bemenő paramétere:

– **Address of Bit:**

Értékhatár (n.i): a duplaszó címére: n=0000...9999, a bitcímre: i=00..31

Bitcím számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Lehetséges módosító tagok:

cím után: “,” inxelési operátor.

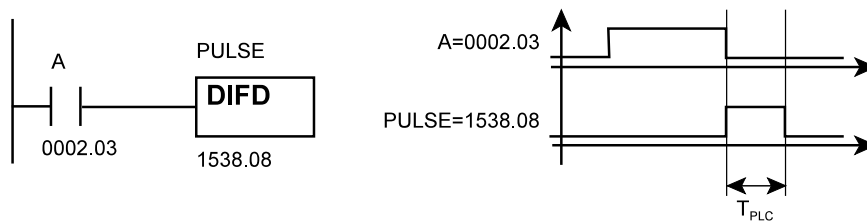
– **Remark:**

Megjegyzés: a Remark ~~szövege~~ írt szöveg lesz a létrafok kommentje.

A DIFD utasítás működése

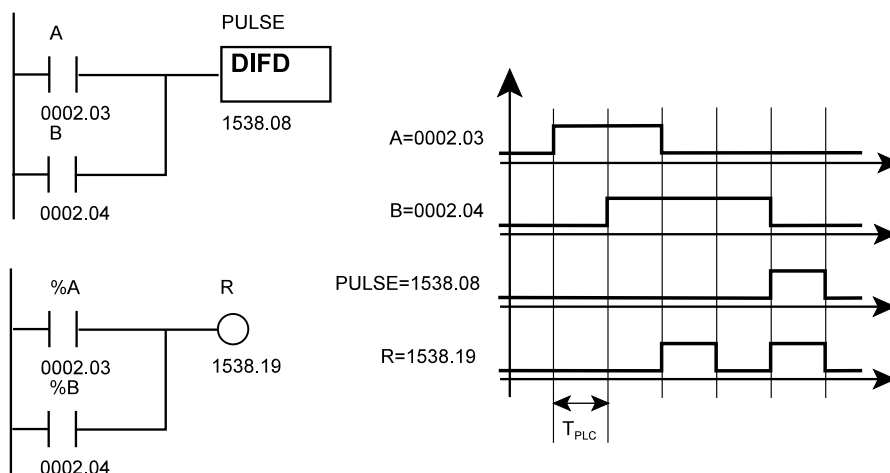
Ha az utasítás előtti feltételek összessége IGAZ (1) állapotból HAMIS (0) állapotba vált, a vonatkozó memóriabitet az utasítás 1-be írja 1 PLC ciklus (T_{PLC} idő) erejéig, vagyis a jel lefutó élére differenciál.

A DIFD utasítás ú.n. záróelem, azaz nincs kimenete, utána már nem köthető semmi.



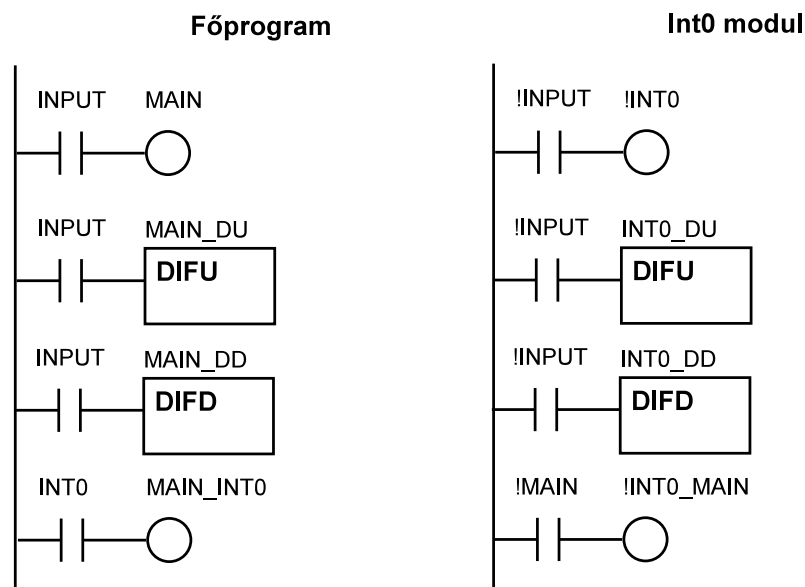
A DIFD utasítás alkalmazása

A DIFD utasítás és a % operátor alkalmazásának különbségére hívja fel a figyelmet az alábbi példa. Ha az (A OR B) összefüggés eredményét differenciáljuk DIFD utasítással, más eredményt kapunk, mintha az A lefutó élét vagyoljuk a B lefutó élével és az eredményt az R relén tároljuk.



6.1.11 A bitműveleteket végző utasítások és a ! operátor a két modulban

Az alábbi két programrészlet alapján vizsgáljuk meg a bitműveleteket végző utasításokat a Főprogramban és az Int0 modulban.



Az INPUT nevű bemenetet

- a Főprogram a PLC memóriából olvassa,
- míg az Int0 modul !INPUT címmel a TimeSlice memóriából.

A MAIN kimenetet

- a Főprogram a PLC memóriába írja,

míg az INT0 kimenetet

- az Int0 modul a TimeSlice memóriába írja !INT0 címmel.

A DIFU, DIFD utasításokat mindkét modulban használjuk.

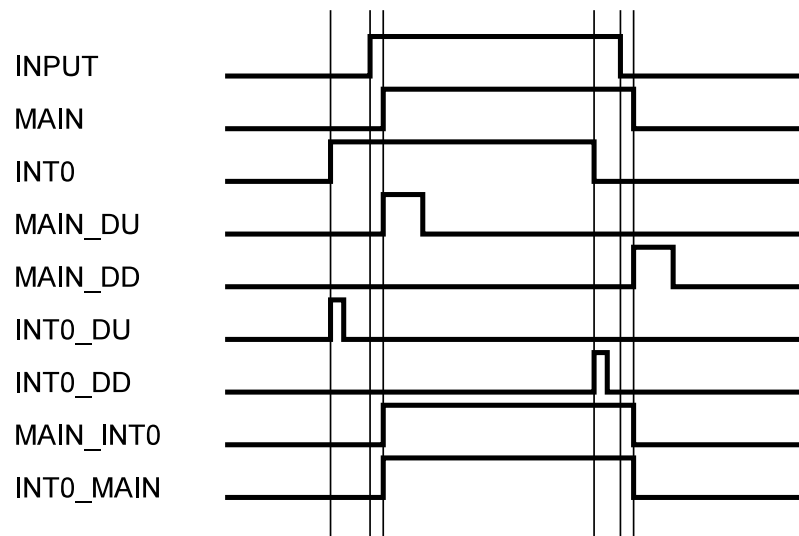
A Főprogramban a MAIN_DU, MAIN_DD címeiken, az Int0 modulban az INT0_DU, INT0_DD címeiken tároljuk a kimeneteket.

Az alábbi idődiagrammon az egyes jelek időbeni lefutása látható.

Az INPUT jel a PLC memóriából kerül frissítésre, míg az INT0 jel a TimeSlice memóriából veszi a bemenetet és oda írja a kimenetet. Ez az oka annak, hogy az INT0 jel időben megelőzi az INPUT jelet.

A DIFU, DIFD utasítások kimenetén jól látszik, hogy az Int0 modul nagyobb gyakorisággal fut, mint a Főprogram, ezért az INT0_DU, INT0_DD kimenetek rövidebb ideig vannak bekapcsolva, mint a MAIN_DU, MAIN_DD kimenetek.

A MAIN_INT0 és az INT0_MAIN jelek között nincs különbség. A MAIN_INT0 a jelet a PLC memóriából veszi (INT0 utasítás), ami T_{PLC} időnként frissül. Az INT0_MAIN jelet hiába írjuk a TimeSlice memóriába, a bemenetet hiába vesszük a TimeSlice memóriából, mivel a MAIN jel T_{PLC} időközönként frissül, ezért a két jel MAIN_INT0 és az INT0_MAIN időben azonos lefolyású.



A grafikus elemek sorokba és oszlopokba rendeződnek. Egy létrafok egy logikai egység, amely egy vagy több sorból állhat. A PLC programban a

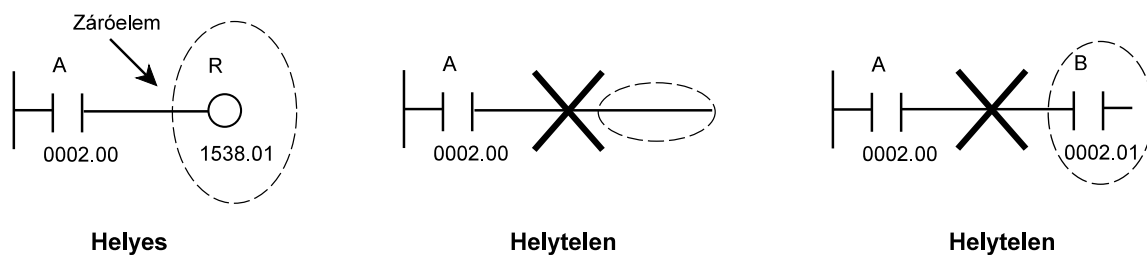
oszlopok száma

[illegible]

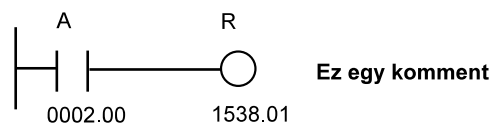
Helyes

Helytelen

41

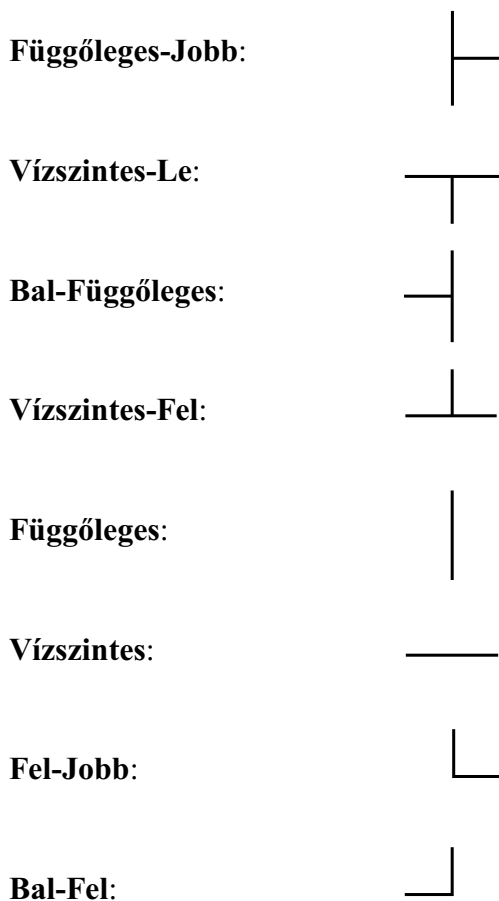


Záróelem esetén mindig megadhatunk **kommentet** a Remark paraméter kitöltésével. A komment a záróelem sorába íródik, a záróelemet követő oszloptól kezdődően.



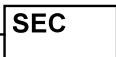
6.2.1 Összekötő elemek

Az összekötő elemek nem utasítások, semmilyen be-, vagy kimeneti paraméterük nincs, csak az “áramút” jelölésére szolgáló “vezetékek”, grafikus elemek, amelyek helyfoglalása egy cella. A létradiagrammos PLC az alábbi összekötő elemeket használja:



6.2.2 A létradiagram logikai szektorainak kommentezése: a SEC utasítás

Mint már láttuk a záróelemek után komment is írható. Létezik az utasítások között kizárólag komment megadására szolgáló utasítás is.

A SEC utasítás szimbóluma: 

Figyelem! A SEC utasítás, bár záróelem, kizárólag az 1. oszlopba írható!

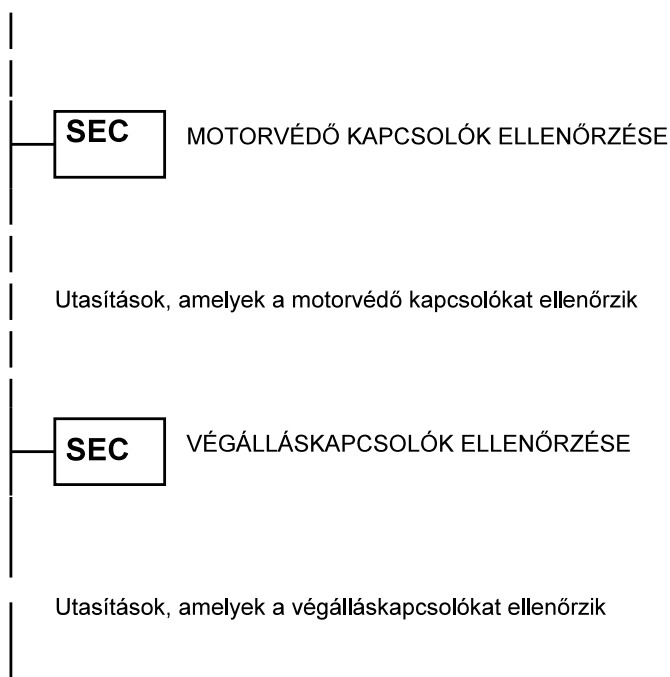
A SEC utasítás bemenő paramétere:

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a logikai egység kommentje.

A SEC utasítás leírása

A SEC utasítás nem végez semmilyen műveletet, kizárólag arra szolgál, hogy a PLC program egyik logikai szektorát elkülönítsük a másiktól. Bemenő paraméterként csak komment szöveg adható meg.



6.3 Adatmozgató utasítások

Az adatmozgató utasításokkal a PLC memória tetszőleges regiszterének adhatunk értéket, vagy tetszőleges regiszterét másolhatjuk át egy másik címre.

Értékadásnál mindig meg kell adni, hogy a bevivendő adat milyen formátumú:

#: előjeles decimális egész szám,

#\$: hexadecimális szám, vagy

*: lebegőpontos szám.

A regiszterek címezésére (szimbólikus, szám, vagy indexelt) a korábbiakban elmodottak érvényesek.

Az adatmozgató utasítások ki-, és bemenete

Minden adatmozgató utasításnak van egy

engedélyező bemenete.

Az engedélyező bemenet igaz állapotában végzi el a mozgatót.

Az adatmozgató utasításoknak *konfigurálhatunk*

kimenetet.

A kimenet IGAZ állapotot vesz fel, ha az utasítás bemenete IGAZ és a mozgató műveletet végrehajtotta. A kimenet HAMIS állapotot vesz fel, ha a bemenete HAMIS, vagy, ha az utasítás nem hajtható végre, azaz ha az utasítás beállítja az FL_ER hibajelzőt!

A kimenetre akár újabb adatmozgató utasítást is köthetünk.

Az összes mozgató utasításnak közősek a bemenő paraméterei.

Az adatmozgató utasítások bemenő paraméterei

– **Address of Result:**

Értékhatár: n...9999 (n: ahol a felhasználói címek kezdődnek)

A célregiszter címe számszerűen, vagy szimbólikusan. Indexelt megadás lehetséges.

A megadott érték, vagy a mozgató regiszter tartalma erre a címre kerül.

– **Operand:**

A mozgató adat értéke, vagy a mozgató regiszter címe.

Ha adatot adunk meg, mindig az utasításnak megfelelő formátumú adatot kell beírni.

Regisztercím esetén a cím lehet szimbólikus, szám, vagy indexelt.

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

– **Remark:**

Ha a kimenet nincs engedélyezve az ide írt megjegyzés kommentként kerül megjelenítésre, illetve, ha nem írunk ide semmit, az Address of Result mezőben megadott szimbólum kommentjét írja ide.

6.3.1 Dupla szó (DWORD) mozgatása: a MOV és az MVN utasítás

A

MOV

utasítás, ha a bemenetén a feltétel teljesül a DWORD hosszúságú

operátorral megadott előjeles decimális számot, vagy

#\$ operátorral megadott hexadecimális számot, vagy

a DWORD hosszúságú regiszter tartalmát

beírja a célregiszterbe.

Az

MVN

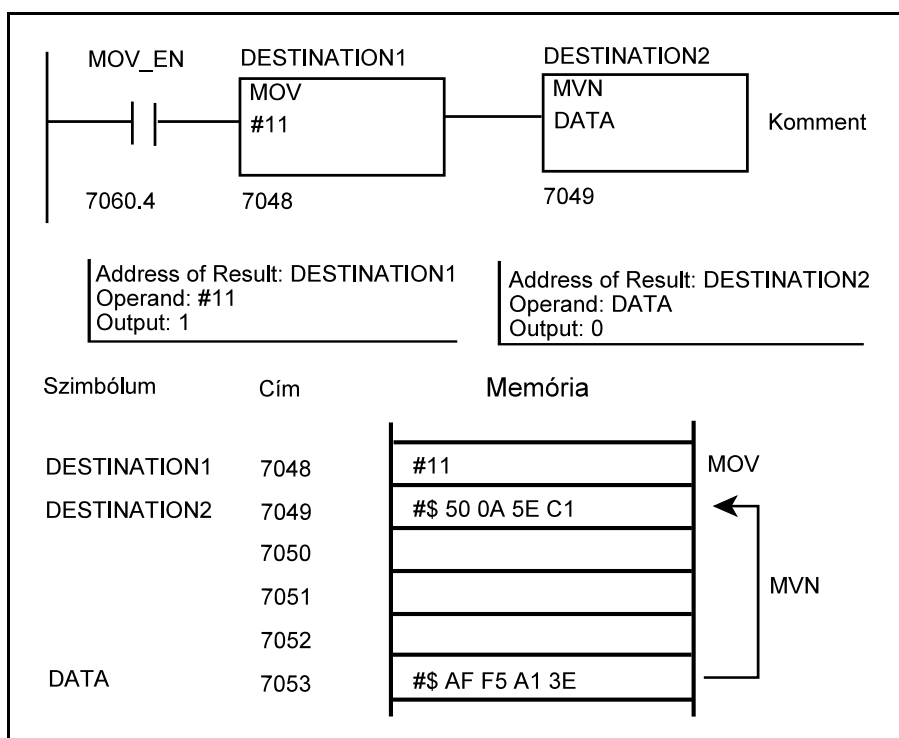
utasítás, ha a bemenetén a feltétel teljesül a DWORD hosszúságú

#\$ operátorral megadott hexadecimális számot, vagy

a DWORD hosszúságú regiszter tartalmát

bitenként negálja és beírja a célregiszterbe.

Ha az utasítás Output paramétere engedélyezve van, további utasításokat írhatunk utána.



6.3.2 Lebegőpontos (double) adat mozgatása: MOVF

A

MOVF

utasítás, ha a bemenetén a feltétel teljesül a 2 DWORD hosszúságú

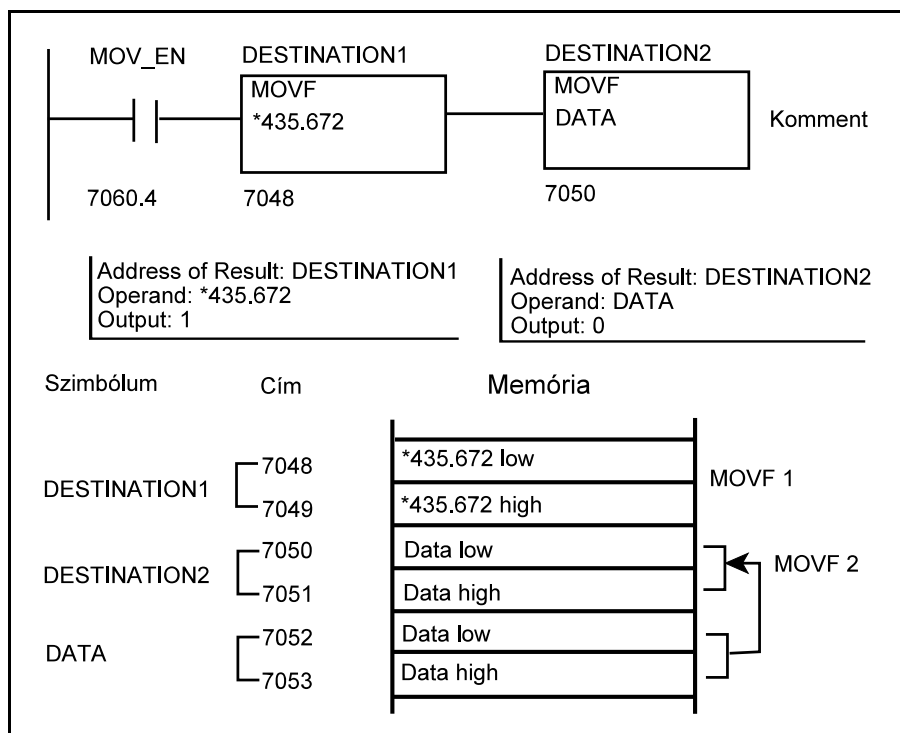
* operátorral megadott double hosszúságú lebegőpontos számot, vagy

2 darab DWORD hosszúságú regiszter tartalmát

beírja a célregiszterekbe.

☞ **Figyelem!** Az Address of Result és az Operand számára is 2 DWORD hosszúságú regisztert kell lefoglalni.

Ha az utasítás Output paramétere engedélyezve van, további utasításokat írhatunk utána.



6.4 Időzítők

Az időzítők segítségével különböző típusú késleltetéseket és különböző szélességű impulzusokat tudunk beállítani a PLC programban.

Az időzítők ki-, és bemenete

Minden időzítőnek van egy

engedélyező bemenete.

Az engedélyező bemenet vezérli az időzítő működését.

Az összes időzítőknak van

kimenete.

A kimeneten a jel az időzítő működésének megfelelően, késleltetve jelenik meg.

Az összes időzítőnek közősek a bemenő paraméterei.

Az időzítők bemenő paraméterei

– **Address of Timer:**

Értékhatár: n...9999 (n: ahol a felhasználói címek kezdődnek)

Az időzítő címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Az időzítőt a programozónak kell definiálnia a felhasználói tárterületen.

Az időzítő címe az időzítő pillantnyi értékét tartalmazó regiszterre mutat, ahol az időzítő számol, és amely címről az időzítő pillanatnyi értékét kiolvashatjuk. Amikor az időzítő a bemenetén felfutó élet kap, ebbe a regiszterbe a Set Value paraméteren megadott és a Time Basis paraméterrel módosított érték íródik. Bármikor, akár az időzítő működése közben is ez a regiszter átírható.

Mindig az ebbe a regiszterbe írt érték számol lefele milliszekundumos léptékkel. Ha a le-számolás megtörtént a regiszter értéke 0 lesz mindaddig, amíg a bemenetén újra engedélyezést nem kap.

☞ **Figyelem:** *Az időzítő címét, vagy szimbolikus nevét felhasználhatjuk feltételvizsgáló utasítás-ban, azaz érintkezőként is. A feltétel értéke:*

IGAZ: ha a címről kiolvasott érték $\neq 0$

HAMIS: ha a címről kiolvasott érték $= 0$.

– **Set Value:**

Értékhatár: 0...2³¹

Az időzítés értéke.

Fixpontos szám, vagy regiszterhivatkozás. Indexelt megadás lehetséges.

Ha az időzítőt a bemenetén engedélyezzük ez a szám íródik az Address of Timer paraméteren megadott című regiszterbe.

– **Time Basis:**

Értékhatár: 0, 1, 2, 3

A Set Value paraméteren megadott érték értelmezését határozza meg:

0: msec (milliszekundum)

1: sec (másodperc)

2: min (perc)

3: hours (óra)

Fixpontos szám, vagy konstans szimbólum (Pl. MSEC #0, SEC #1, MIN #2, HOURS #3), de lehet regiszterhivatkozás is. Indexelt megadás lehetséges.

A Set Value paraméterre megadott értékhatár a msec-ben megadott időzítésekre vonatkozik. Az időzítők az Address of Timer címen megadott regiszterben mindig msec-es időá-lappal számolnak. Ha az időzítő Time Basis paraméterén megadott érték msec-től eltérő,

a következő értékhatárok adhatók be:

=1 sec: 2147483

=2 min: 35791

=3 hours: 596

– **Remark:**

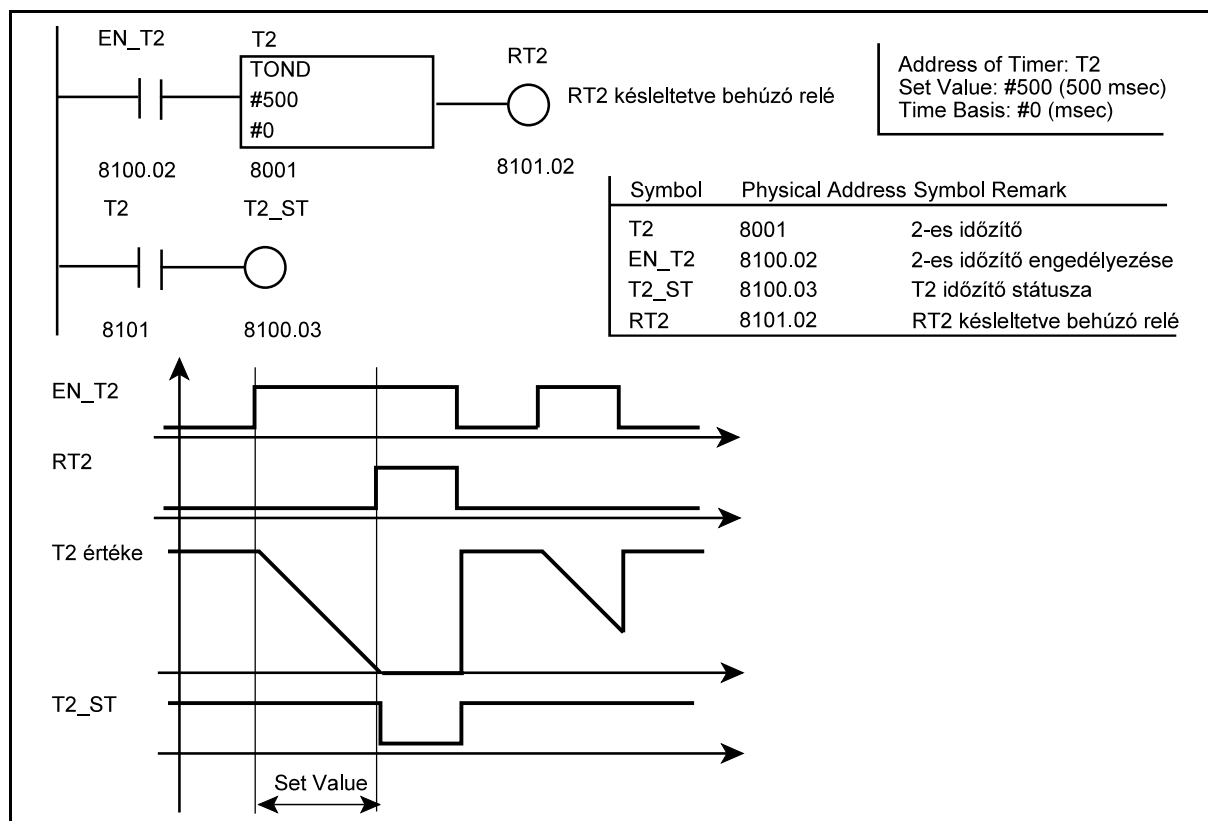
Nem használt.

6.4.1 Késleltetve behúzó időzítő TOND

A késleltetve behúzó időzítő (Timer, ON Delay) kimenetén a bemeneten megjelenő IGAZ érték az időzítőben beállított késleltetés után jelenik meg.

Ha az engedélyező bemenet értéke IGAZ, az időzítő elkezd lefele számolni, és ha leszámolt, a kimenete IGAZ értékre vált. Ha az engedélyező bemenet HAMISra vált, az Address of Timer címen megadott regiszter tartalma újra felveszi a Set Value paraméteren megadott, a Time Basis paraméterrel módosított értéket. Az időzítő kimenete HAMIS értéket vesz fel.

Az Address of Timer címen megadott regiszterre feltételvizsgálattal is hivatkozhatunk: ha a regiszter értéke $\neq 0$, a feltétel IGAZ, ha $=0$ a feltétel értéke HAMIS lesz.

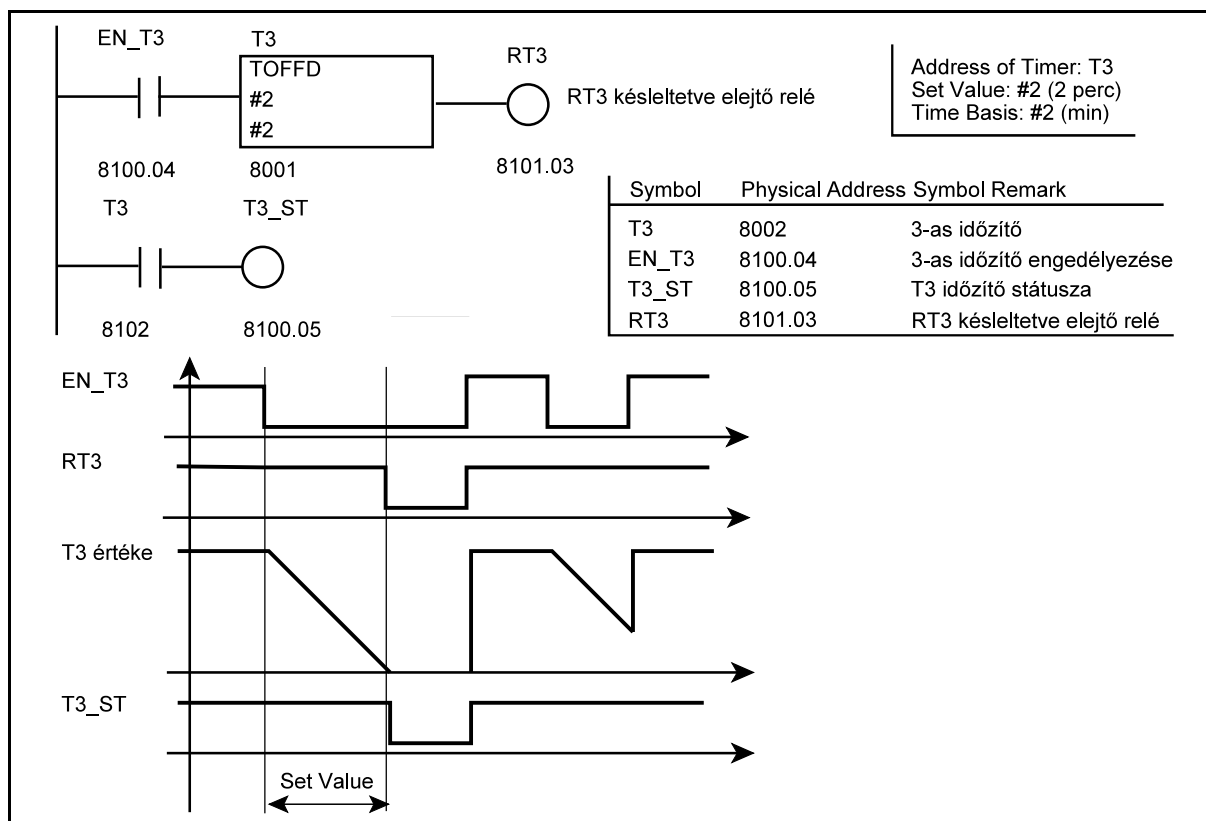


6.4.2 Késleltetve kiejtő időzítő TOFFD

A késleltetve kiejtő időzítő (Timer, OFF Delay) kimenetén a bemeneten megjelenő HAMIS érték az időzítőben beállított késleltetés után jelenik meg.

Ha az engedélyező bemenet értéke HAMIS, az időzítő elkezd lefele számolni, és ha leszámolt, a kimenete HAMIS értékre vált. Ha az engedélyező bemenet IGAZra vált, az Address of Timer címen megadott regiszter tartalma újra felveszi a Set Value paraméteren megadott, a Time Basis paraméterrel módosított értéket. Az időzítő kimenete IGAZ értéket vesz fel.

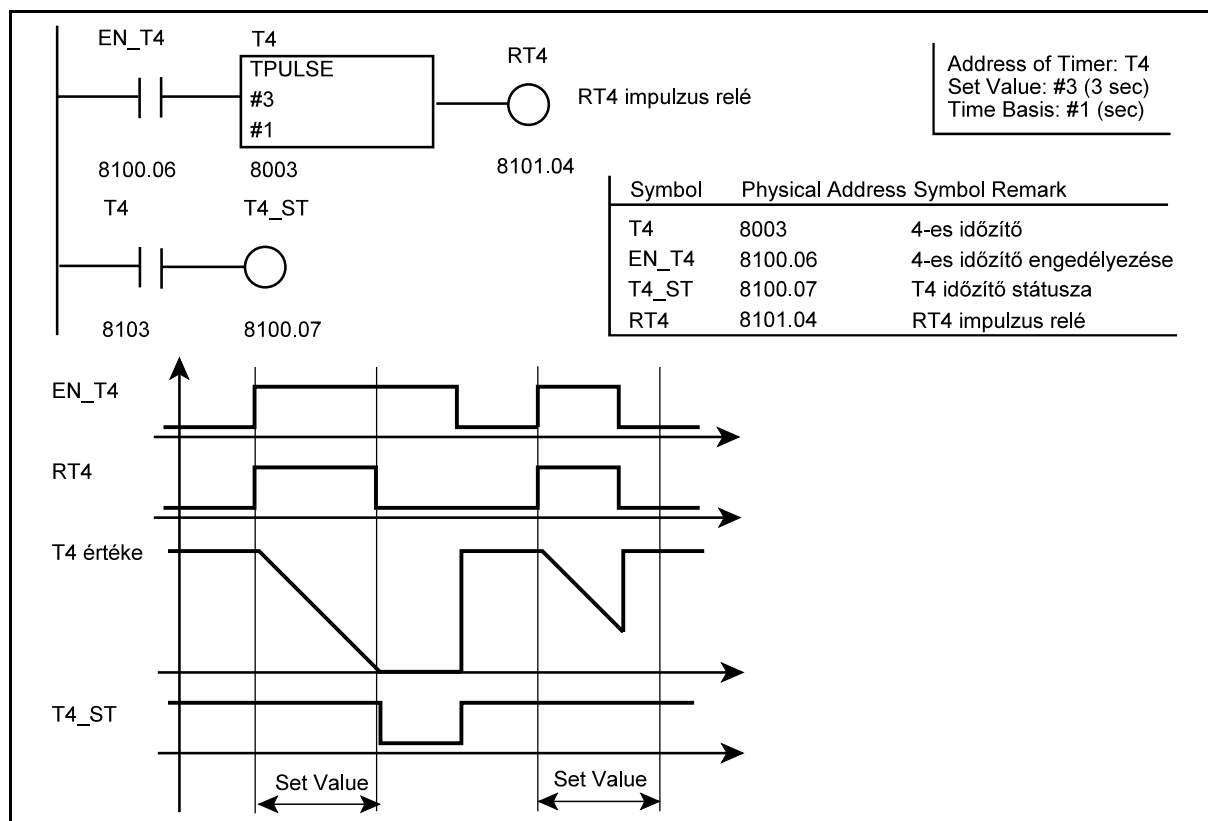
Az Address of Timer címen megadott regiszterre feltételvizsgálattal is hivatkozhatunk: ha a regiszter értéke $\neq 0$, a feltétel IGAZ, ha $=0$ a feltétel értéke HAMIS lesz.



6.4.3 Programozható szélességű impulzus TPULSE

Az impulzus időzítő kimenetén a bemeneten megjelenő IGAZ érték azonnal megjelenik, majd az időzítőben beállított késleltetés után a kimenet HAMIS állapotot vesz fel.

Ha az engedélyező bemenet értéke IGAZ, kimenete bekapcsol, IGAZ lesz, és az időzítő elkezd lefele számolni. Ha leszámolt, a kimenete HAMIS értékre vált. Ha az engedélyező bemenet HAMISra vált, az Address of Timer címen megadott regiszter tartalma újra felveszi a Set Value paraméteren megadott, a Time Basis paraméterrel módosított értéket és a kimenet is HAMIS lesz. Az Address of Timer címen megadott regiszterre feltételvizsgálattal is hivatkozhatunk: ha a regiszter értéke $\neq 0$, a feltétel IGAZ, ha $=0$ a feltétel értéke HAMIS lesz.



6.5 Számlálók

A számlálók segítségével impulzusokat tudunk megszámolni a PLC programban.

A számlálók bemenetei és kimenete

Minden számlálónak van egy

- órajel (CK),
- egy irány (U/D) és egy
- alaphelyzetbe hozó (R)

bemenete.

- **CK bemenet:** A számlálót a CK bemenetére érkező impulzus felfutó éle működteti.
- **U/D bemenet:** A számláló CK bemenetére érkező impulzus a számláló értékét növeli, ha az U/D bemenete HAMIS, illetve csökkenti, ha az U/D bemenet IGAZ.
- **R bemenet:** Ha az R bemenet IGAZ, a számláló felveszi a kezdeti értéket. Az R bemenet precedenciája magasabb, mint a CK bemeneté: ha állapota IGAZ a számláló nem számlál, még ha a CK bemenetére impulzusok is érkeznek.

A számláló

- **kimenete** IGAZ értéket vesz fel, ha a számláló tartalma egyenlő lesz egy előre beállított számmal.

Az összes számlálónak közősek a bemenő paraméterei.

A számlálók bemenő paraméterei

- **Address of Counter:**

Értékhatár: $n \dots 9999$ (n: ahol a felhasználói címek kezdődnek)

A számláló címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

A számlálót a programozónak kell definiálnia a felhasználói tárterületen.

A számláló címe a számláló pillantnyi értékét tartalmazó regiszterre mutat, ahol a számláló számol, és amely címről a számláló pillanatnyi értékét kiolvashatjuk. Amikor a számláló R bemenete IGAZ, ebbe a regiszterbe a Starting Value paraméteren megadott érték íródik. Bármikor, akár a számláló működése közben is ez a regiszter átírható.

- ☞ **Figyelem:** *A számláló címét, vagy szimbolikus nevét felhasználhatjuk feltételvizsgáló utasításban, azaz érintkezőként is. A feltétel értéke:*

IGAZ: ha a címről kiolvasott érték $\neq 0$

HAMIS: ha a címről kiolvasott érték $= 0$.

- **Starting Value:**

Értékhatár: $0 \dots 2^{31}$

Fixpontos szám, vagy regiszterhivatkozás. Indexelt megadás lehetséges.

A számláló kezdeti értéke, amit az R bemeneten megjelenő IGAZ állapot hatására felvesz. Gyűrűs számlálóknál a számláló átfordulásának alsó határa.

- **Ending Value:**

Értékhatár: $0 \dots 2^{31}$

Fixpontos szám, vagy regiszterhivatkozás. Indexelt megadás lehetséges.

Az egyszerű számlálók eddig az értékig számolnak, gyűrűs számlálóknál az átfordulás felső határa.

- **Compare Value:**

Értékhatár: $0 \dots 2^{31}$

Fixpontos szám, vagy regiszterhivatkozás. Indexelt megadás lehetséges.

A számláló kimenete akkor vesz fel IGAZ állapotot, ha a számláló tartalma megegyezik az itt megadott számmal.

– **Remark:**

Megjegyzés.

6.5.1 Egyszerű számláló CNT

Az egyszerű számláló addig számol, amíg az Ending Value paraméterén megadott értéket eléri. Ezután leáll, mégha a CK bemenetén újabb impulzusok is érkeznek. A számláló újraindítható az alaphelyzetbe hozó, reset bemenetének IGAZ állapotba kapcsolásával.

A következőkben két példát mutatunk a számláló használatára.

Lefele számláló

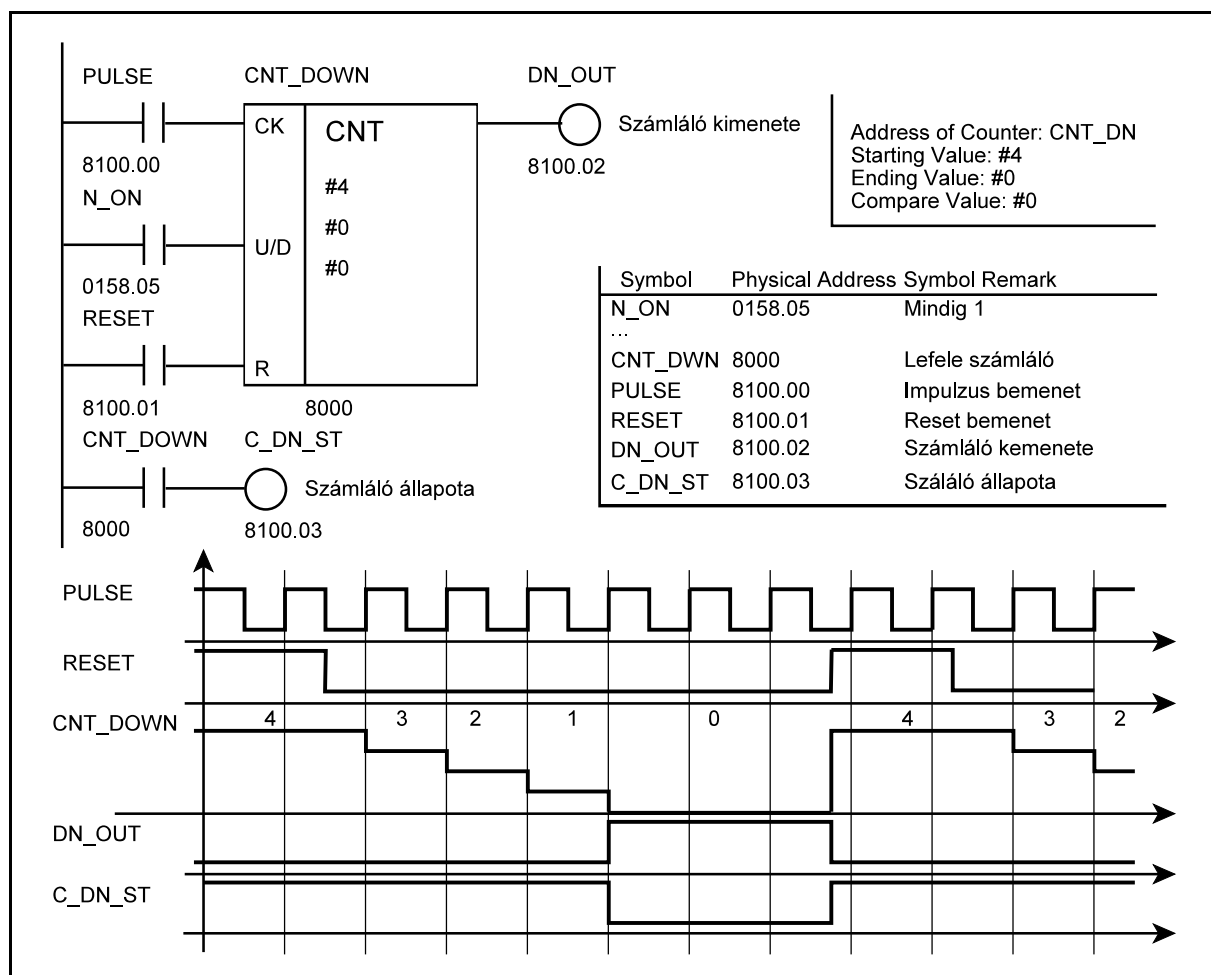
A számláló U/D bemenetét fixen 1-be kapcsoljuk a mindig IGAZ állapotú N_ON jelző használatával, hogy lefele számoljon.

A számláló a RESET jel hatására 4-ről indul, mert a Starting Value paraméter értéke 4, és 0-ig számol mert az Ending Value paraméter 0. A CK bemenet felfutó élére számol.

A számláló kimenete IGAZ állapotba kerül, ha a tartalma 0, mert a Compare Value paraméter 0.

A számláló a RESET jel hatására újraindul.

Az Address of Counter címen megadott regiszterre feltételvizsgálattal is hivatkozhatunk: ha a regiszter értéke $\neq 0$, a feltétel IGAZ, ha $=0$ a feltétel értéke HAMIS lesz.



Felfele számláló

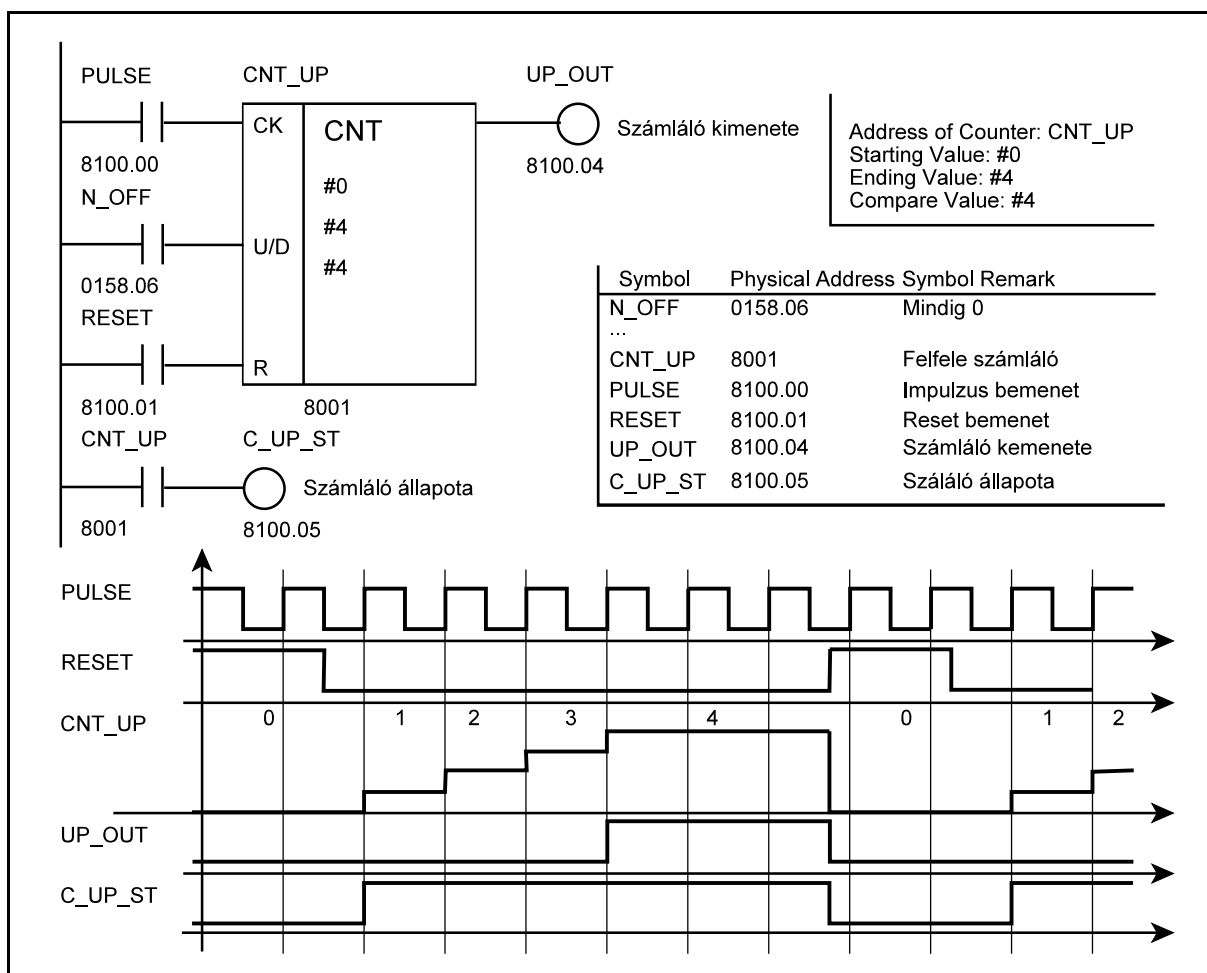
A számláló U/D bemenetét fixen 0-ba kapcsoljuk a mindig HAMIS állapotú N_OFF jelző használatával, hogy felfele számoljon.

A számláló a RESET jel hatására 0-ról indul, mert a Starting Value paraméter értéke 0, és 4-ig számol mert az Ending Value paraméter 4. A CK bemenet felfutó élére számol.

A számláló kimenete IGAZ állapotba kerül, ha a tartalma 4, mert a Compare Value paraméter 4.

A számláló a RESET jel hatására újraindul.

Az Address of Counter címen megadott regiszterre feltételvizsgálattal is hivatkozhatunk: ha a regiszter értéke $\neq 0$, a feltétel IGAZ, ha $=0$ a feltétel értéke HAMIS lesz.



6.5.2 Reverzibilis számláló CNTR

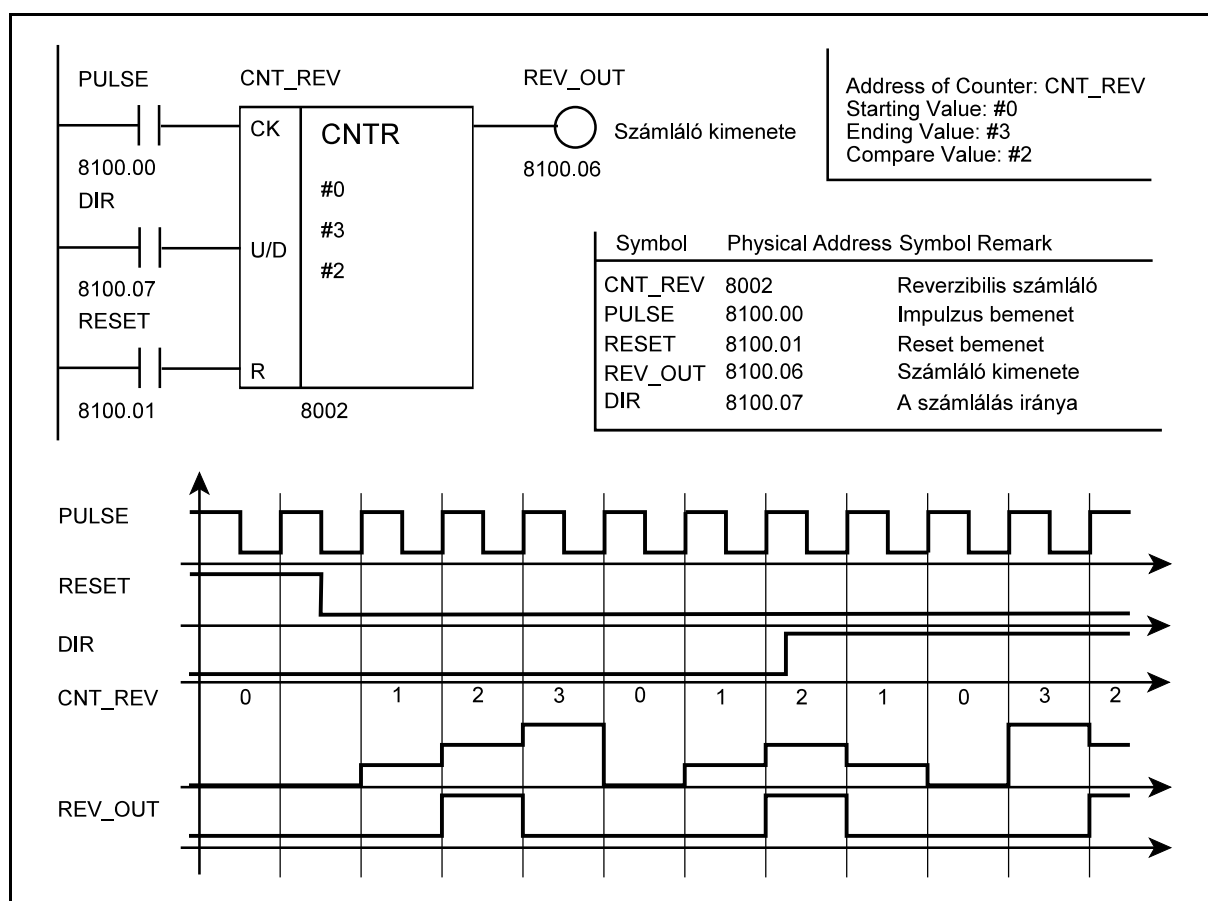
Ha a reverzibilis számláló az Ending Value paraméterén megadott értéket eléri fölfele számlálás közben, a következő impulzus hatására átfordul és a Starting Value paraméteren megadott értéktől folytatja a számlálást.

Ellenkezőleg, ha lefele számlálás közben a számláló értéke eléri a Starting Value paraméteren megadott értéket, a következő impulzus hatására átfordul, és az Ending Value paraméteren megadott értéktől folytatja a számlálást.

A számláló az alaphelyzetbe hozó, reset (R) bemenetének IGAZ állapotba kapcsolásával felveszi a Starting Value paraméteren meghatározott értéket.

A számláló kimenetét a Compare Value paraméteren megadott érték vezérli. Ha a számláló értéke megegyezik a paraméterrel kimenete IGAZ állapotba kerül.

Az alábbi ábrán a reverzibilis számláló működése látható:



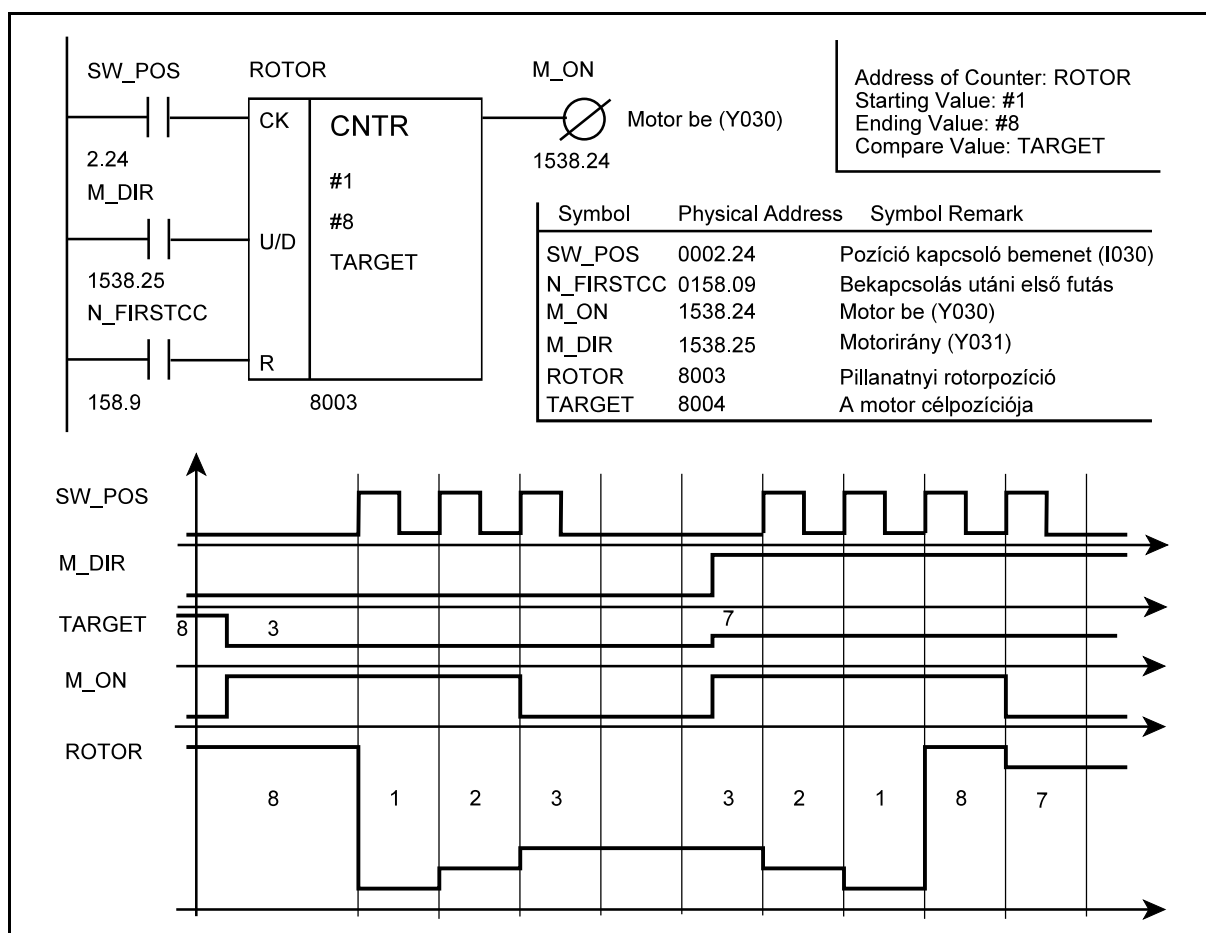
Példa a reverzibilis számlálók alkalmazására

Gyakran előforduló feladat, hogy egy forgó eszközt kell meghatározott pozícióra mozgatni és a pozícióját nyilvántartani a PLC programban. Ilyen eszköz lehet egy revolverfej, magazin, vagy osztóasztal.

Legyen a forgó eszközünk 8 pozíciós és a pozícióit számozzuk 1-től 8-ig. A reverzibilis számláló Starting Value paramétere tehát 1, Ending Value paramétere 8. A számláló Compare Value paramétere legyen egy szimbolikus regiszter TARGET, amibe a célpozíciót írjuk. A számláló neve legyen ROTOR.

Ha a PLC új parancsot kap átírja a TARGET regiszter értékét. Mivel a számláló értéke nem egyezik meg a célpozícióval, kimenete HAMIS állapotba kerül, ami invertálás után bekapcsolja az M_ON motor be kimenetet. A kimenet addig lesz bekapcsolva, amíg a számláló pillanatnyi értéke egyenlő nem lesz a TARGET regiszterben megadott célpozícióval.

A számláló U/D iránybemenetére a motor forgásirányát meghatározó M_DIR kapcsoló jelét kötjük, míg a CK órabemenetre a forgó eszköz pozícióját jelző kapcsoló bemenetet.



6.6 Forgatásvezérlő utasítás ROT

A ROT utasítás a megadott bemenő paraméterek és a megfelelően beállított bemenetek alapján ki tudja számítani, hogy egy forgó eszközt (pl: revolverfejet, osztóasztalt, magazint) milyen irányban és hány lépéssel kell elforgatni, hogy a kívánt pozícióba jusson.

A ROT utasítás bemenetei és kimenete

A ROT utasításnak 5 bemenete van. Ezek a következők:

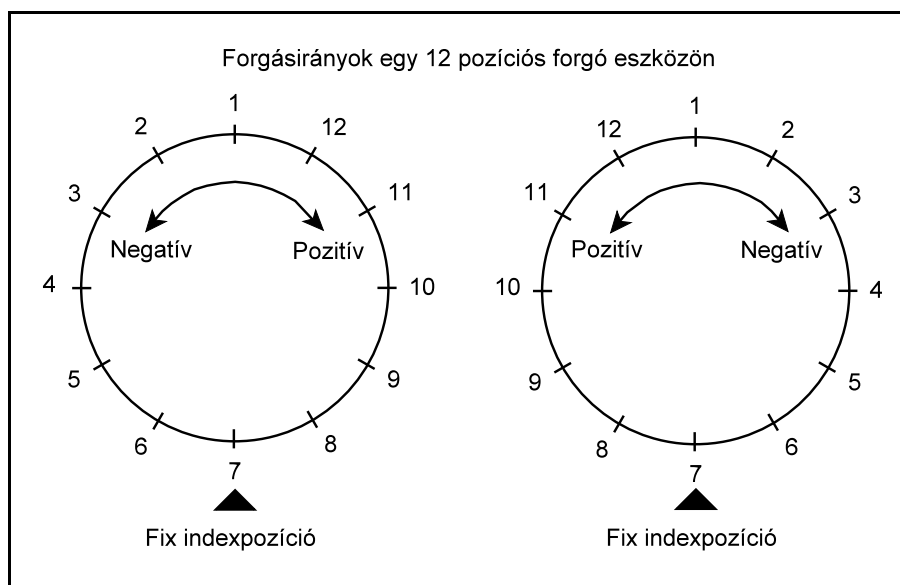
- **EN bemenet:** az engedélyező EN bemenet *felfutó élére* az utasítás bemenő paraméterei és bemenetei alapján kiszámítja az elérendő pozíció értékét és a forgásirányt beállítja a kimeneten.
- **BID bemenet:** a kétirányú forgatásválasztó BID bemenet HAMIS állapotában a DIR bemeneten megadott irány szerint számolja az elérendő pozíció értékét, a kimenetet pedig ennek megfelelően állítja be. A bemenet IGAZ állapotában a rövidebb úton való forgatáshoz számolja ki az elérendő pozíció értékét és a kimenetet kikapcsolja, ha a rövidebb úthoz tartozó forgásirány pozitív, illetve bekapcsolja, ha a szükséges forgásirány negatív.
- **DIR bemenet:** A BID bemenet hamis állapotában a DIR bemeneten megadott irány szerint számolja az elérendő pozíció értékét, a kimenetet pedig ennek megfelelően állítja be: ha DIR=0 a kimenet is nulla, egyébként 1.
- **INC bemenet:** ha a bemenet értéke HAMIS az elérendő pozíciót abszolút értékben számolja ki, ha IGAZ, inkrementálisan, vagyis, hogy hány lépéssel kell az eszközt elforgatni.
- **PRE bemenet:** ha a bemenet HAMIS az elérendő pozíció megegyezik a célpozícióval INC=0 állásnál, illetve INC=1 állásnál az elérendő pozíció a cél- és aktuális pozíció különbsége. A bemenet IGAZ állapotában az elérendő pozíció a célpozíció előtti első pozíció lesz.

A ROT utasítás

- **kimenete** azt mondja meg, hogy az eszközt a BID és a DIR bemenetek állapotának függvényében melyik irányban kell forgatni:
 ha a kimenet HAMIS pozitív,
 ha a kimenet IGAZ negatív

irányban.

Ha az utasítás nem hajtható végre az utasítás beállítja az FL_ER hibajelzőt és az utasítás kimenete és a Position to Go regiszter értéke változatlan marad!



A ROT utasítás bemenő paraméterei– **Position to Go:**

Értékhatár: 0...9999

A regiszter címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

*A programozónak kell definiálnia a felhasználói tárterületen.*Annak a regiszternek a címe, ahová az utasítás által kiszámolt **elérendő pozíció** értéke kerül, az utasítás kimenő regisztere.

Ha a BID bemenet IGAZ az elérendő pozíciót a rövidebb úton számítja. Ha az INC bemenete igaz az elérendő pozíciót inkrementálisan számítja, azaz hány lépést kell megtenni a pozícióig. Ha az PRE bemenete igaz az elérendő pozíció a célpozíció előtti pozícióra mutat.

– **Current Position:**

Értékhatár: 0...9999

A regiszter címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

*A programozónak kell definiálnia a felhasználói tárterületen.*Annak a regiszternek a címe, amely a forgó eszköz **pillanatnyi pozícióját** tartalmazza.– **Goal Position:**

Értékhatár: 0...9999

A regiszter címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

*A programozónak kell definiálnia a felhasználói tárterületen.*Annak a regiszternek a címe, amelyik a **célpozícióra** mutat. Ez általában egy programozott érték, például az NC-től átvett T szerszámszám értéke. A Position to Go kimenő regiszter számított értéke meg fog egyezik a Goal Position értékével, ha az INC és a PRE bemenetek hamisak.– **Starting Value:**Értékhatár: 0...2³¹

Fixpontos szám, vagy regiszterhivatkozás. Indexelt megadás lehetséges.

A forgó eszköz kezdőpozíciója, legkisebb pozíciója. Pl. 1.

– **Ending Value:**Értékhatár: 0...2³¹

Fixpontos szám, vagy regiszterhivatkozás. Indexelt megadás lehetséges.

A forgó eszköz végpozíciója, legnagyobb pozíciója. Pl. 12.

 Figyelem!*Ha a megadott címek kívül esnek az értelmezhető tartományon az utasítás beállítja az FL_ER hibajelzőt és az utasítás kimenete és a Position to Go regiszter értéke változatlan marad.*Példa a ROT utasítás használatára

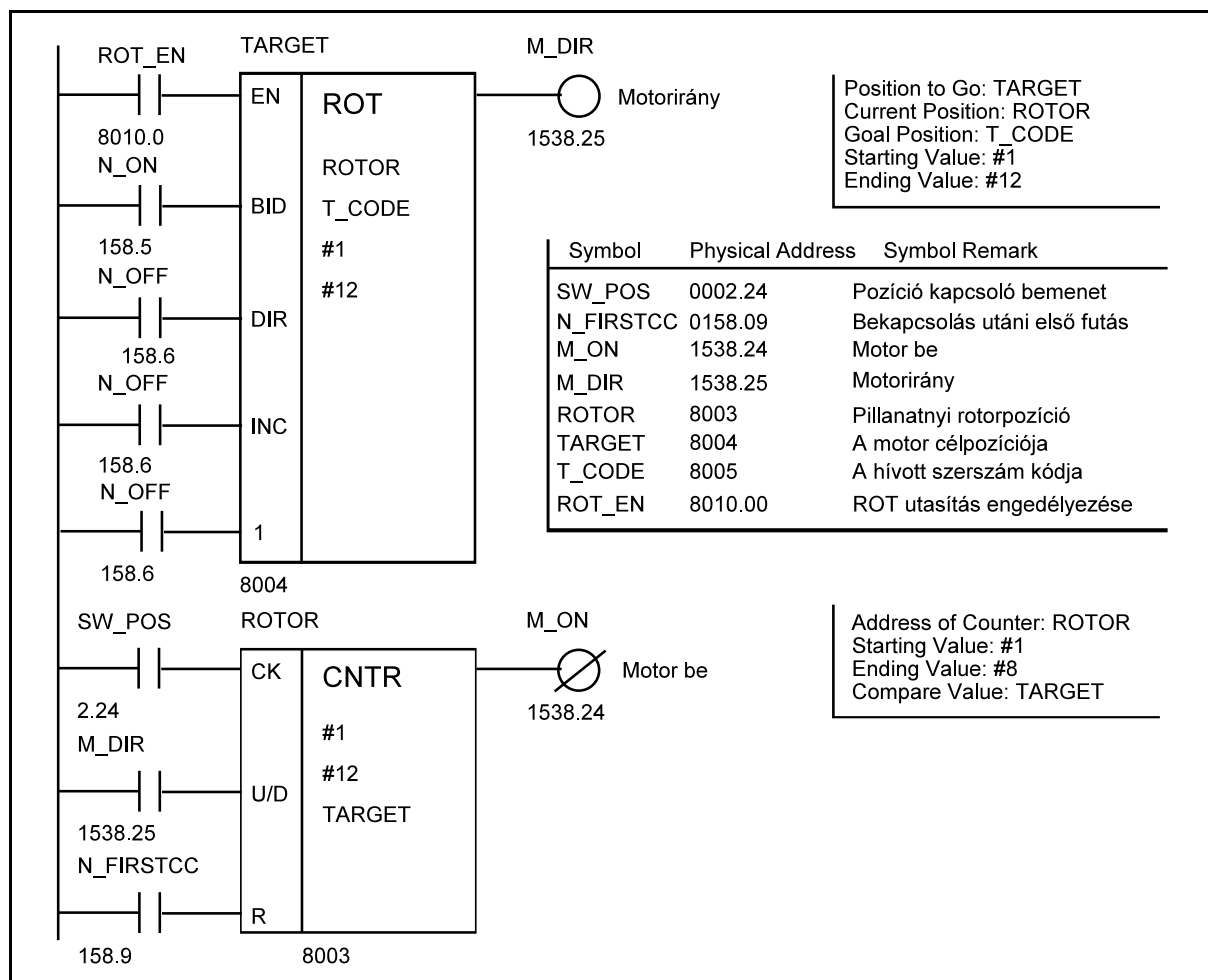
Legyen egy 12 állású revolverfej, amely mindkét irányban forgatható. Az NC-től jövő T kód a T_CODE regiszterbe kerül.

A ROT utasítás által kiszámolt elérendő pozíció a TARGET regiszterbe kerül. A pillanatnyi pozíciót a ROTOR nevű számláló értéke adja. A célpozíció értéke a T_CODE regiszter. A számítás-hoz még megadtuk a forgó eszköz kezdeti és végértékét: #1 és #12.

Az eszköz mindkét irányban forog, tehát a BID bemenete IGAZ. Az elérendő pozíciót abszolútban kérjük, ezért az INC bemenet HAMIS. Nem akarunk az elérendő pozíció előtt egygel megállni, ezért az PRE bemenet is HAMIS.

A revolverforgatás a ROT_EN jelző 1-be írásával indul. A ROT utasítás kiszámolja és a TAR-

GET regiszterbe írja az elérendő pozíció értékét és beállítja a megfelelő motorirányt M_DIR. Mivel a ROTOR nevű számláló Compare Value értéke (TARGET) megváltozott, kimenete HAMIS lesz, ami bekapcsolja az M_ON motor be kimenetet. A motor az M_DIR kimenet meghatározta irányban forogni kezd, és addig forog, amíg a ROTOR számláló tartalma el nem éri a TARGET értéket. Ekkor a számláló kimenete IGAZ állapotot vesz fel, kikapcsolja az M_ON kimenetet, ami leállítja a motort.



6.7 Eltoló és forgató utasítások

6.7.1 Shift regiszter SHTR

A shift regiszter segítségével bitmintát tudunk mozgatni jobbra, vagy balra a PLC programban.

Az SHTR utasítás bemenetei és kimenete

A shift regiszternek van egy

órjel (CK),
egy irány (L/R) és egy
adat (D)

bemenete.

- **CK bemenet:** A shift regiszter a CK bemenetére érkező impulzus *felfutó élére* a regiszterben található bitmintát eltolja.
- **L/R bemenet:** A shift regiszter a CK bemenetére érkező impulzus a regiszterben található bitmintát:
eggyel *balra* tolja, ha az *L/R* bemenete *HAMIS*, illetve
eggyel *jobbra* tolja, ha az *L/R* bemenet *IGAZ*.
- **D bemenet:** Az adat bemenet (D) állapota határozza meg a *belépő bit* értékét. Ha a bemenet *HAMIS* a belépő bit 0, ha *IGAZ* a belépő bit 1.
Balra tolásnál a legelső 0-ás bit helyére,
jobbra tolásnál a legfelső 31-es bit helyére

kerül a D adatbemenet állapota.

A shift regiszter

- **kimenete** a regiszterből kilépő bit:
balra tolásnál a legfelső 31-es állapotát,
jobbra tolásnál a legelső 0-ás bit állapotát
veszi fel. Értéke *HAMIS*, ha a kilépő bit 0, illetve *IGAZ*, ha a kilépő bit 1.

A shift regiszter bemenő paramétere

- **Address of Shift Register:**

Értékhatár: 0...9999

A regiszter címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

A shift regisztert a programozónak kell definiálnia a felhasználói tárterületen.

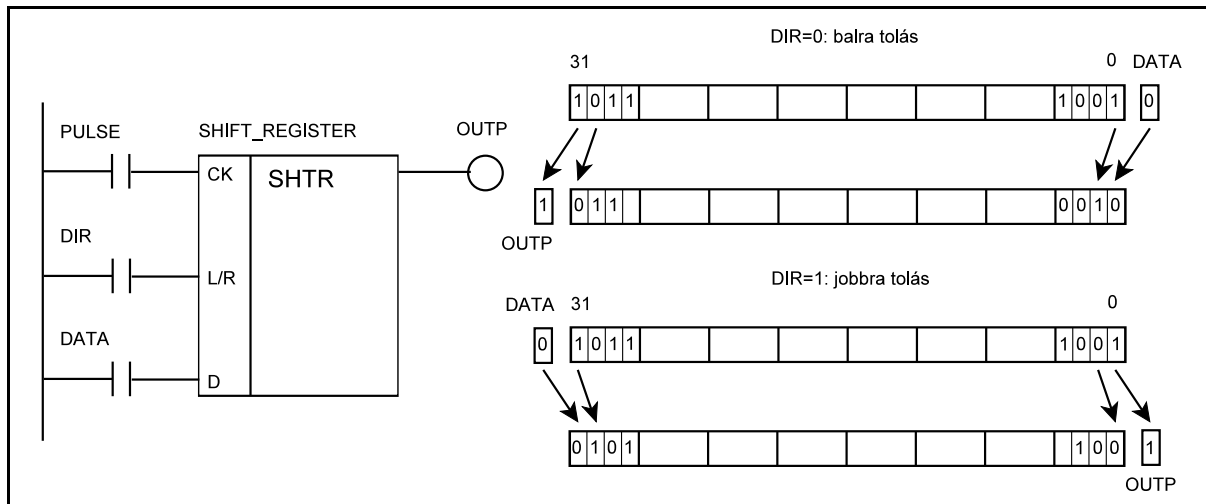
A shift regiszter címe a regiszter pillanatnyi értékét tartalmazó címre mutat, ahol a regiszter működik, és amely címről a regiszter pillanatnyi értékét kiolvashatjuk. Ugyanez vonatkozik a regiszter bármely bitjére is. Bármikor, akár a regiszter működése közben is átirható.

Figyelem!

Ha az Address of Shift Register paraméteren megadott cím kívül esik az értelmezhető tartományon az utasítás beállítja az FL_ER hibajelzőt.

- **Remark:**

Megjegyzés.



6.7.2 Aritmetikai eltolások: ASHL, ASHR

Az aritmetikai eltolást végző utasítások közül az *ASHL balra*, az *ASHR jobbra* tolja el a kijelölt bitmintát.

Mindkét utasításnak egy

statikus engedélyező bemenete, és egy

hibátlan végrehajtást jelző kimenete van.

Az engedélyező bemenet IGAZ állapotában a kijelölt eltolás elvégzésre kerül.

Ha az **engedélyező bemenete IGAZ** és a **műveletet hiba nélkül végrehajtotta** az utasítás **kimenete IGAZ** állapotot vesz fel, ellenkező esetben kimenete HAMIS lesz. A kimenetet felhasználhatjuk egy következő, pl aritmetikai művelet engedélyezésére.

Az utasítások segítségével kijelölhetünk a tárban egy n darab duplaszóból álló, összefüggő tömböt, a tömb kezdő és végcímének megadásával. Ezen kívül megadhatjuk, hogy egyszerre hány bittel tolja el az utasítás a bitmintát. Az eltolás során a kilépő bitek elvesznek, a felszabaduló helyekre 0-ák lépnek be.

Balra tolásnál a kilépő bitek a tömb utolsó duplaszavának felső bitjei, amelyek elvesznek, a belépő 0-ák a tömb első duplaszavának alsó bitjeire kerülnek.

Jobbra tolásnál a kilépő bitek a tömb első duplaszavának alsó bitjei, amelyek elvesznek, a belépő 0-ák a tömb utolsó duplaszavának alsó bitjeire kerülnek.

Az aritmetikai eltolások bemenő paraméterei

– Starting Address:

Értékhatár: 0...9999

A kijelölendő tömb kezdőcíme számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

– Ending Address:

Értékhatár: 0...9999

A kijelölendő tömb végcíme számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

👉 **Figyelem!**

A kezdő és végcímre igaznak kell lennie az alábbi feltételnek:

$$\text{Starting Address} \leq \text{Ending Address}$$

Ha a fenti feltétel nem teljesül, vagy a megadott címek kívül esnek az értelmezhető tartományon az utasítás beállítja az FL_ER hibajelzőt és az utasítás kimenete HAMIS állapotot vesz fel.

Ha a kezdő és a végcím értéke megegyezik csak egy duplaszón végzi el a műveletet.

– **Shift Value:**

Értékhatar: $0 \dots 2^{31}$

Fixpontos szám, vagy regiszterhivatkozás. Indexelt megadás lehetséges.

Ezen a paraméteren kell megadni, hogy hány bittel kell eltolni a tömb által kijelölt bitmin-tát.

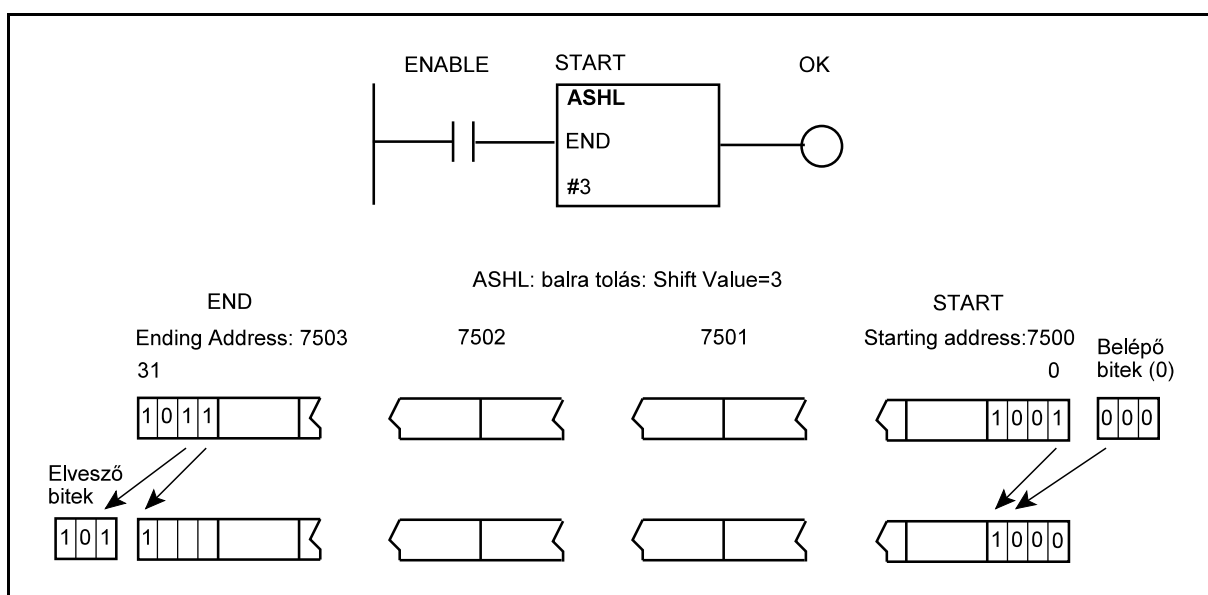
– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

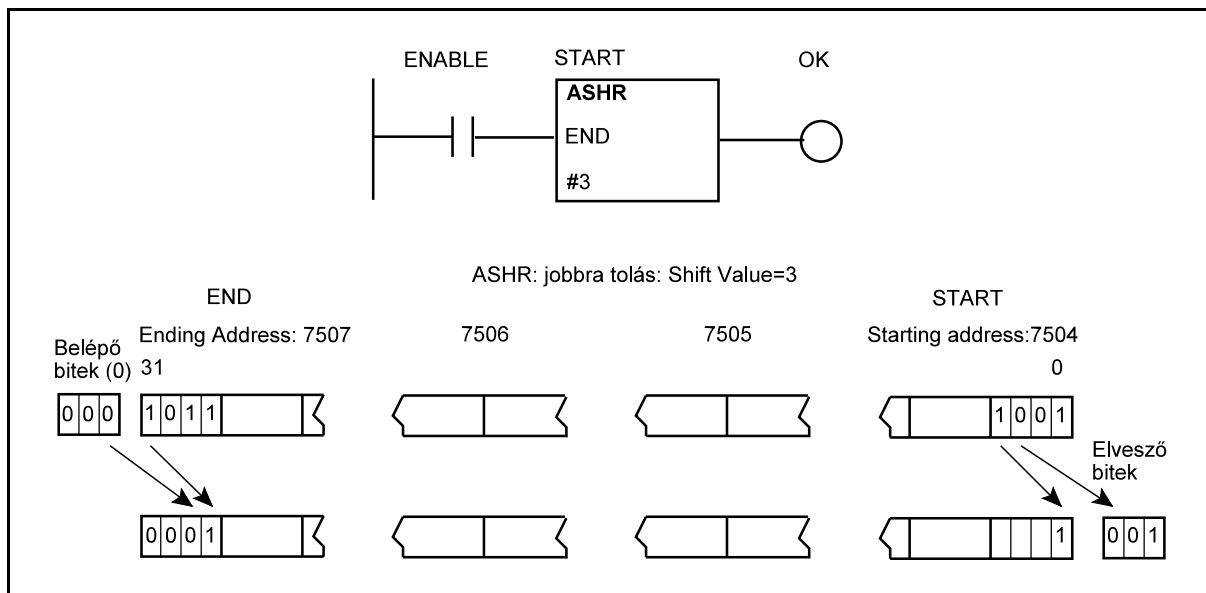
– **Remark:**

Megjegyzés.

Az ASHL, balra tolás tetszőleges számú bittel, utasítás működése:



Az ASHR, jobbra tolás tetszőleges számú bittel, utasítás működése:



6.7.3 Aritmetikai forgatások: ARTL, ARTR

Az aritmetikai forgatást végző utasítások közül az **ARTL balra**, az **ARTR jobbra** forgatja el a kijelölt bitmintát.

Mindkét utasításnak egy

statikus engedélyező bemenete, és egy

hibátlan végrehajtást jelző kimenete van.

Az engedélyező bemenet IGAZ állapotában a kijelölt forgatás elvégzésre kerül.

Ha az **engedélyező bemenete IGAZ** és a **műveletet hiba nélkül végrehajtotta** az utasítás **kimenete IGAZ** állapotot vesz fel, ellenkező esetben kimenete HAMIS lesz. A kimenetet felhasználhatjuk egy következő, pl aritmetikai művelet engedélyezésére.

Az utasítások segítségével kijelölhetünk a tárban egy n darab duplaszóból álló összefüggő tömböt, a tömb kezdő és végcímének megadásával. Ezen kívül megadhatjuk, hogy egyszerre hány bittel forgassa el az utasítás a bitmintát. A forgatás során a kilépő bitek a másik oldalon belépnek.

Balra tolásnál a kilépő bitek a tömb utolsó duplaszavának felső bitjei, amelyek a tömb első duplaszavának alsó bitjeire kerülnek.

Jobbra tolásnál a kilépő bitek a tömb első duplaszavának alsó bitjei, amelyek a tömb utolsó duplaszavának alsó bitjeire kerülnek.

Az aritmetikai forgatások bemenő paramétere

– **Starting Address:**

Értékhatár: 0...9999

A kijelölendő tömb kezdőcíme számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

– **Ending Address:**

Értékhatár: 0...9999

A kijelölendő tömb végcíme számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

Figyelem!

A kezdő és végcímre igaznak kell lennie az alábbi feltételnek:

$$\text{Starting Address} \leq \text{Ending Address}$$

Ha a fenti feltétel nem teljesül, vagy a megadott címek kívül esnek az értelmezhető tartományon az utasítás beállítja az FL_ER hibajelzőt és az utasítás kimenete HAMIS állapotot vesz fel.

Ha a kezdő és a végcím értéke megegyezik csak egy duplaszón végzi el a műveletet.

– **Rotation Value:**

Értékhatar: $0 \dots 2^{31}$

Fixpontos szám, vagy regiszterhivatkozás. Indexelt megadás lehetséges.

Ezen a paraméteren kell megadni, hogy hány bittel kell elforgatni a tömb által kijelölt bitmintát.

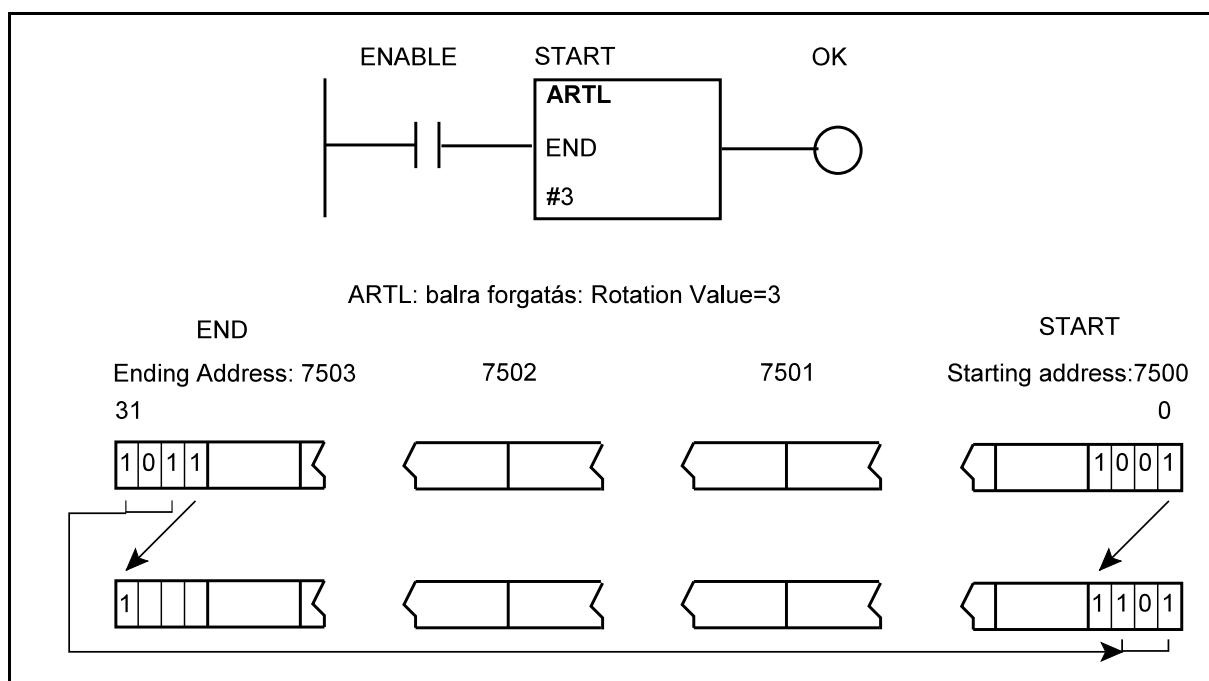
– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

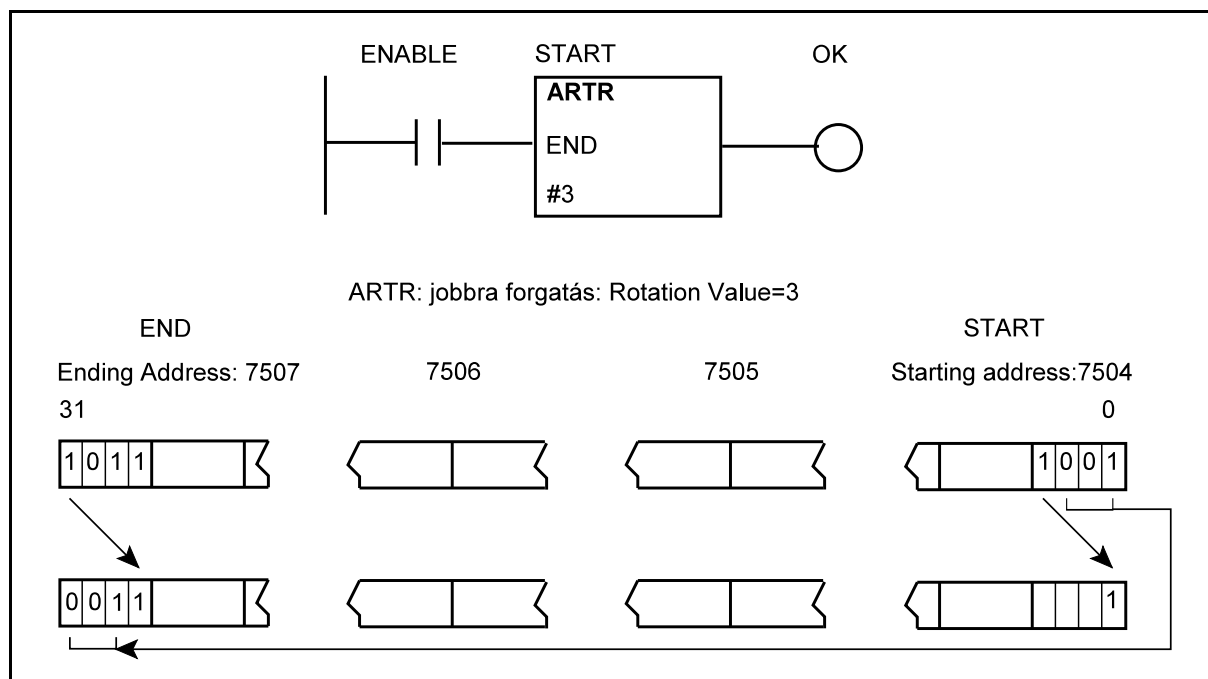
– **Remark:**

Megjegyzés.

Az ARTL, balra forgatás tetszőleges számú bittel, utasítás működése:



Az ARTR, jobbra forgatás tetszőleges számú bittel, utasítás működése:



6.8 Logikai utasítások

A logikai utasításokkal 32 bites adatok között lehet logikai és, vagy, kizáró vagy és negáló műveletet végezni.

6.8.1 Egyoperandusú utasítás: NEG

A **NEG** utasítás egy 32 bites regiszter bitenkénti negálását végzi. Azt a bitet, ahol 0 van 1-be, ahol 1-et talál 0-ba írja.

Az utasításnak egy

statikus engedélyező bemenete, és egy,
a művelet sikeres végrehajtását jelző **kimenete** van.

Az utasítás kimenete HAMIS lesz, ha a bemenete HAMIS, vagy ha az utasítás nem hajtható végre, azaz ha az utasítás beállítja az FL_ER hibajelzőt!

A NEG utasítás bemenő paraméterei

– **Address of Result:**

Értékhatár: 0...9999

A művelet eredményének címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

– **Operand:**

A negálandó adat

#\$ operátorral megadott hexadecimális számot, vagy

a DWORD hosszúságú regiszter tartalmát

bitenként negálja és beírja a célregiszterbe.

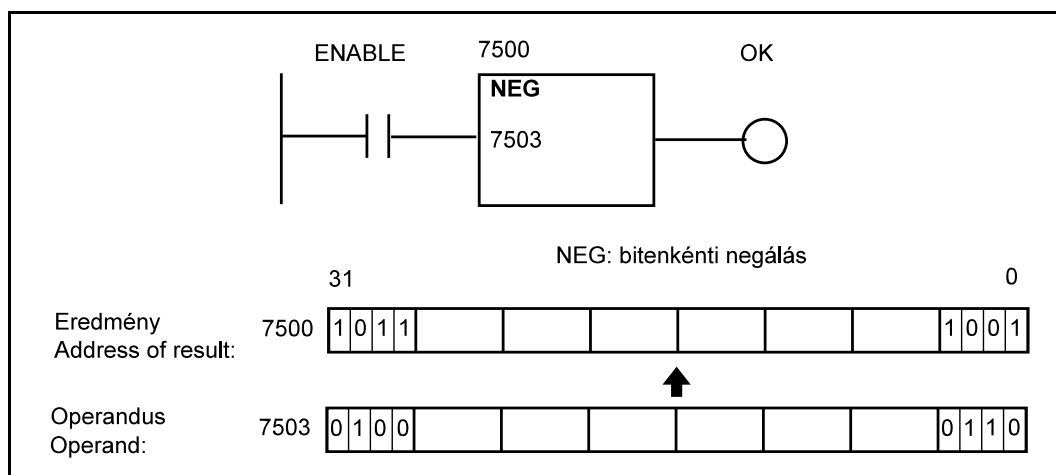
Az operandusra hivatkozhatunk közvetlen értékadással, vagy regiszterhivatkozással. Indexelt megadás lehetséges.

☞ **Figyelem!**

Ha a megadott címek kívül esnek az értelmezhető tartományon az utasítás kimenete HAMIS lesz, és beállítja az FL_ER hibajelzőt.

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.



6.8.2 Kétooperandusú utasítások: AND, OR, XOR

Az

AND utasítás két 32 bites adat bitenkénti ÉS kapcsolatát, az
OR utasítás két 32 bites adat bitenkénti VAGY kapcsolatát, a
XOR utasítás két 32 bites adat bitenkénti KIZÁRÓ VAGY kapcsolatát

valósítja meg.

Az utasításoknak egy

statikus engedélyező bemenete, és egy,
a művelet sikeres végrehajtását jelző **kimenete** van.

Az utasítás kimenete HAMIS lesz, ha a bemenete HAMIS, vagy, ha az utasítás nem hajtható végre, azaz ha az utasítás beállítja az FL_ER hibajelzőt!

Az utasítások bemenő paraméterei

– **Address of Result:**

Értékhatar: 0...9999

A művelet eredményének címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

– **1st Operand:**

A művelet első operandusa.

Az első operandusra *csak regiszterként* hivatkozhatunk. Indexelt megadás lehetséges.

– **2nd Operand:**

A művelet második operandusa.

Az operandusra hivatkozhatunk közvetlen értékadással, vagy regiszterhivatkozással. Indexelt megadás lehetséges.

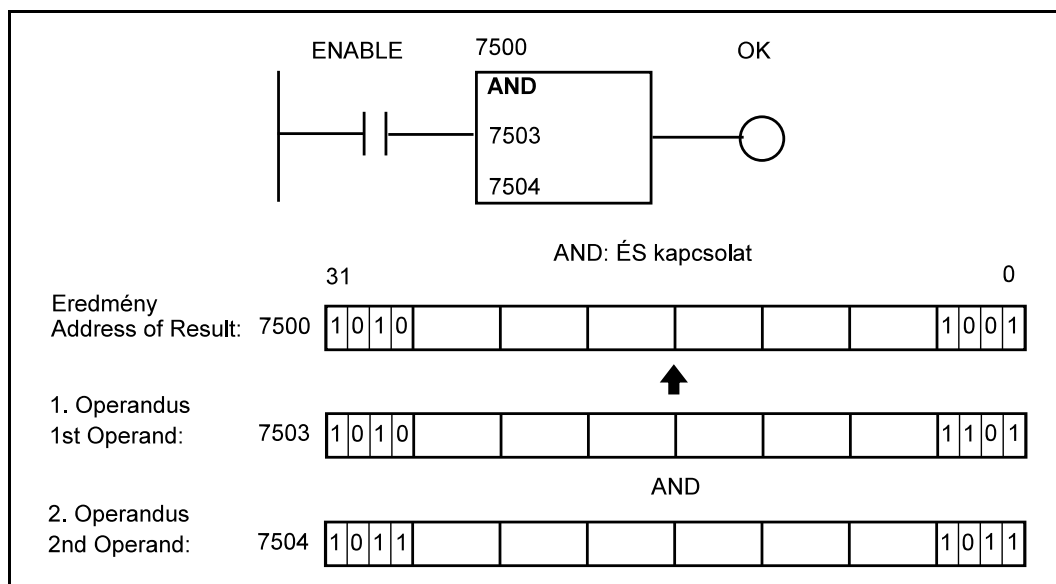
☞ **Figyelem!**

Ha a megadott címek kívül esnek az értelmezhető tartományon az utasítás kimenete HAMIS lesz, és beállítja az FL_ER hibajelzőt.

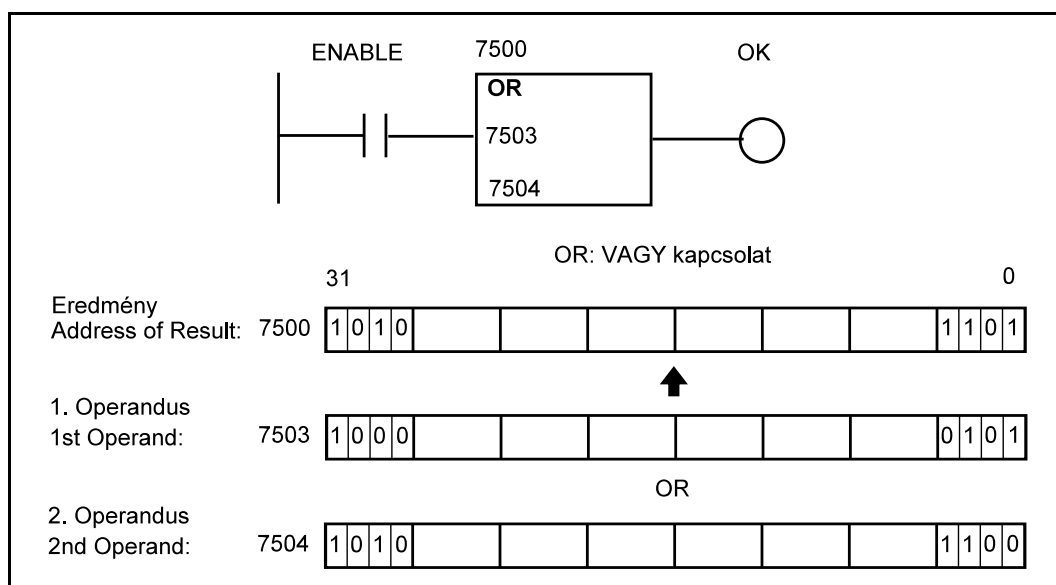
– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

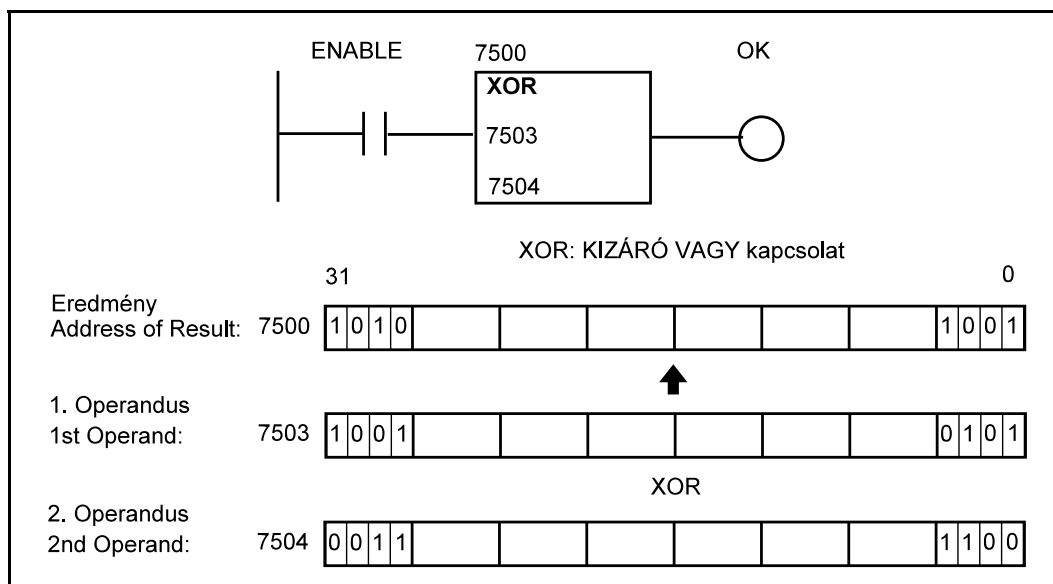
Az **AND** utasítás bitenkénti eredménye akkor 1, ha mindkét operandusának megfelelő bitje 1:



Az **OR** utasítás bitenkénti eredménye akkor 1, ha valamelyik operandusának megfelelő bitje 1:



A **XOR** utasítás bitenkénti eredménye akkor 1, ha a két operandus megfelelő bitjei különböznek.



6.9 Fixpontos aritmetikai utasítások

Az aritmetikai utasítások az összeadás, kivonás, szorzás és osztás.

Az utasításoknak egy

statikus engedélyező bemenete, és egy,
a művelet sikeres végrehajtását jelző **kimenete** van.

Az utasítás kimenete HAMIS lesz, ha a bemenete HAMIS, vagy, ha az utasítás nem hajtható végre, azaz ha az utasítás beállítja az FL_ER hibajelzőt!

Ezen kívül a fixpontos utasítások kezelik az alábbi jelzőket:

FL_UF: alulcsordulás (underflow)

FL_OF: túlcsordulás (overflow)

FL_CY: átvitel (carry)

Az utasítások bemenő paraméterei

– **Address of Result:**

Értékhatar: 0...9999

A művelet eredményének címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

– **1st Operand:**

A művelet első operandusa.

Az első operandusra *csak regiszterként* hivatkozhatunk.. Indexelt megadás lehetséges.

– **2nd Operand:**

A művelet második operandusa.

Az operandusra hivatkozhatunk közvetlen értékadással, vagy regiszterhivatkozással. Indexelt megadás lehetséges.

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

 Figyelem!

Ha a megadott címek kívül esnek az értelmezhető tartományon az utasítás kimenete HAMIS lesz, és beállítja az FL_ER hibajelzőt.

6.9.1 Előjeles, fixpontos összeadás, átvitel nélkül: ADD

Átvitel nélküli összeadáson azt értjük, hogy ha az ADD utasítás végrehajtása előtt egy előző művelet során keletkezett átvitel, azaz az **FL_CY** jelző értéke 1, az **nem adódik hozzá** az ADD utasítás által képzett összeghez.

Az

összeg az Address of Result címen megadott regiszterbe kerül, az

1. összeadandót a 1st Operand, a
2. összeadandót a 2nd Operand regiszterből veszi.

FL_CY átvitel jelző beírásra kerül, ha az összeadás során **átvitel** képződik, vagyis az összeg nem fér el 32 biten, egyébként 0.

```

      7000000F
    +FC000012
  -----
FL  CY=1    6C000021

```

FL_OF túlsordulás jelző beírásra kerül, ha két pozitív szám (31-es bit mindkét számon 0) összeadása után keletkező összeg a 80000000...FFFFFFF negatív számok tartományába esik, egyébként 0.

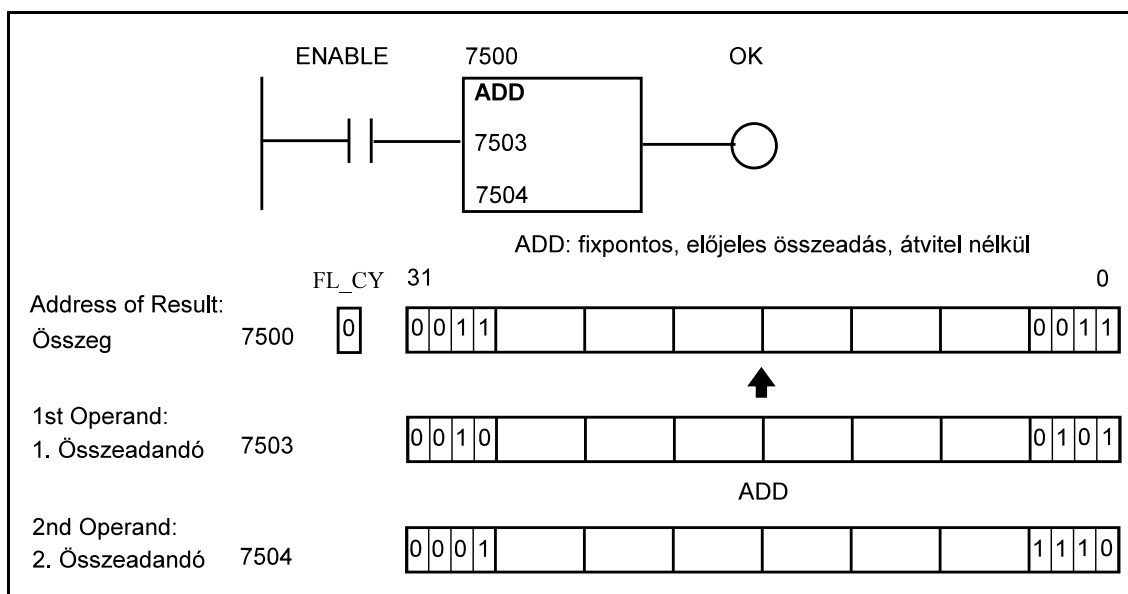
[illegible]

FL_UF alulcsordulás jelző beírásra kerül, ha két negatív szám (31-es bit mindkét számon 1) összeadása után keletkező összeg a 00000000...07FFFFFF pozitív számok tartományába esik, egyébként 0.

```

      8000FC22
      +80F032A1
-----
FL  CY=1,  FL  UF=1  00F12EC3

```



6.9.3 Előjeles, fixpontos szorzás: MUL

A *MUL* előjeles, fixpontos szorzás akkor használható, ha a szorzás eredménye (a szorzat) 32 biten (DWORD) ábrázolható. Ellenkező esetben az utasítás az *FL_OF* túlszordulás jelzőt 1-be állítja.

A

szorzat az Address of Result címen megadott regiszterbe, a

szorzandót a 1st Operand, a

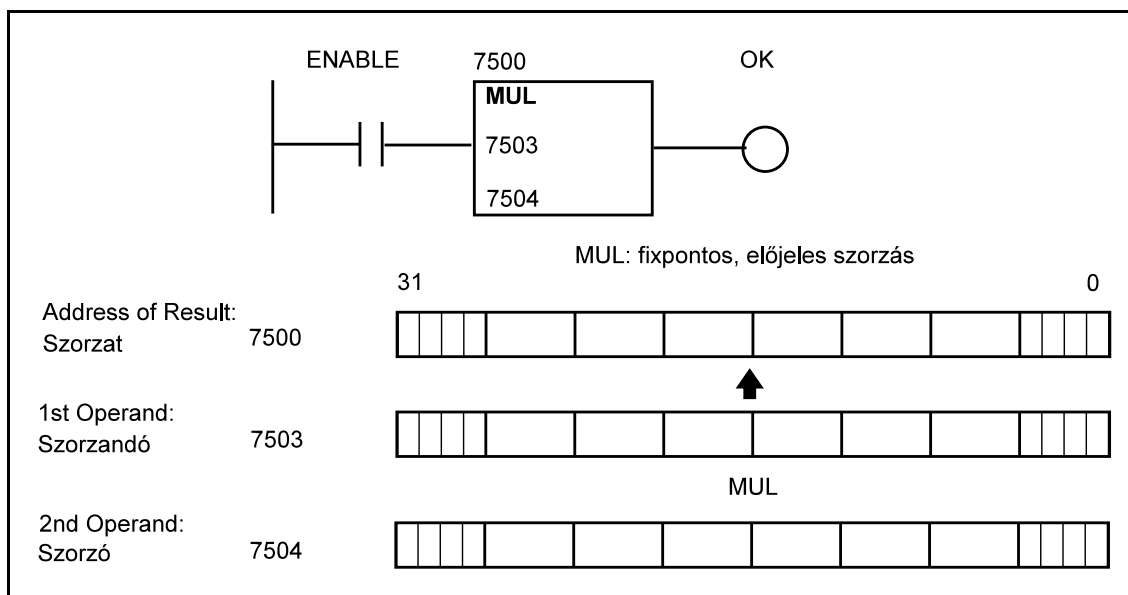
szorzót a 2nd Operand regiszterből veszi.

Az *FL_OF* túlszordulás jelző beírásra kerül, ha az eredmény nem ábrázolható 32 biten:

$$\begin{array}{r} 7000000F \\ \times 0C000012 \\ \hline FL_OF=1 \quad 9400010E \end{array}$$

Mivel a szorzás előjeles, az alábbi esetben nem képződik túlszordulás:

$$\begin{array}{r} FFFFFFFF \\ \times 00000002 \\ \hline FL_OF=0 \quad FFFFFFFE \end{array}$$



6.9.4 Előjeles, fixpontos osztás: DIV

A *DIV* előjeles, fixpontos osztás akkor használható, ha az osztás maradékára nincs szükség.
A

hányados az Address of Result címen megadott regiszterbe kerül, az
osztandót a 1st Operand, az
osztót a 2nd Operand regiszterből veszi.

Az osztás eredménye 0 lesz, ha az osztó nagyobb, mint az osztandó:

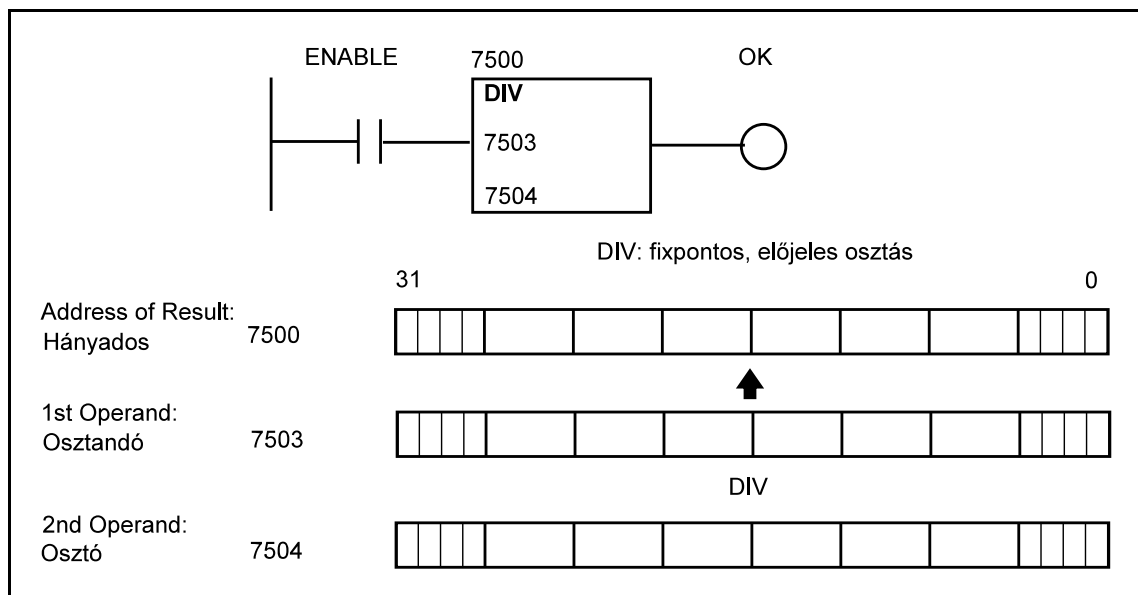
$$\begin{array}{r} 0000002A \\ \div 000000FF \\ \hline 00000000 \end{array}$$

Az osztás eredménye negatív szám lesz, ha vagy az osztó, vagy az osztandó negatív:

$$\begin{array}{r} 00000002 \\ \div FFFFFFFF \\ \hline FFFFFFFE \end{array}$$

Figyelem!

Ha az osztó értéke 0, az osztás eredménye 0 lesz és az **FL_ER** jelzőt 1-be írja!



6.10 Lebegőpontos matematikai műveletek

Az utasításoknak egy

statikus engedélyező bemenete, és egy,
a művelet sikeres végrehajtását jelző **kimenete** van.

Az utasítás kimenete HAMIS lesz, ha a bemenete HAMIS, vagy, ha az utasítás nem hajtható végre, azaz ha az utasítás beállítja az

FL_ER

hibajelzőt!

Ezen kívül a fixpontos utasítások kezelik az alábbi jelzőket:

FL_UF: alulcsordulás (underflow)

FL_OF: túlcsordulás (overflow)

Az utasítások bemenő paraméterei

– **Address of Result:**

Értékhatar: 0...9998

A művelet eredményének címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

– **Operand:**

Az utasítás típusától függően egy vagy két operandus lehetséges. Kétooperandusú művelet pl az összeadás, egyoperandusú a négyzetgyökvonás.

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

☞ **Figyelem!**

Ha a megadott címek kívül esnek az értelmezhető tartományon az utasítás kimenete HAMIS lesz, és beállítja az FL_ER hibajelzőt.

Az utasítások által kezelt jelzők

FL_OF túlcsordulás jelző beírásra kerül, ha az eredmény abszolút értéke olyan nagy, hogy nem ábrázolható dupla pontosságú lebegőpontos számábrázolással.

FL_UF alulcsordulás jelző beírásra kerül, ha az eredmény abszolút értéke olyan kicsi, hogy nem ábrázolható dupla pontosságú lebegőpontos számábrázolással.

6.10.1 Lebegőpontos összeadás: +F

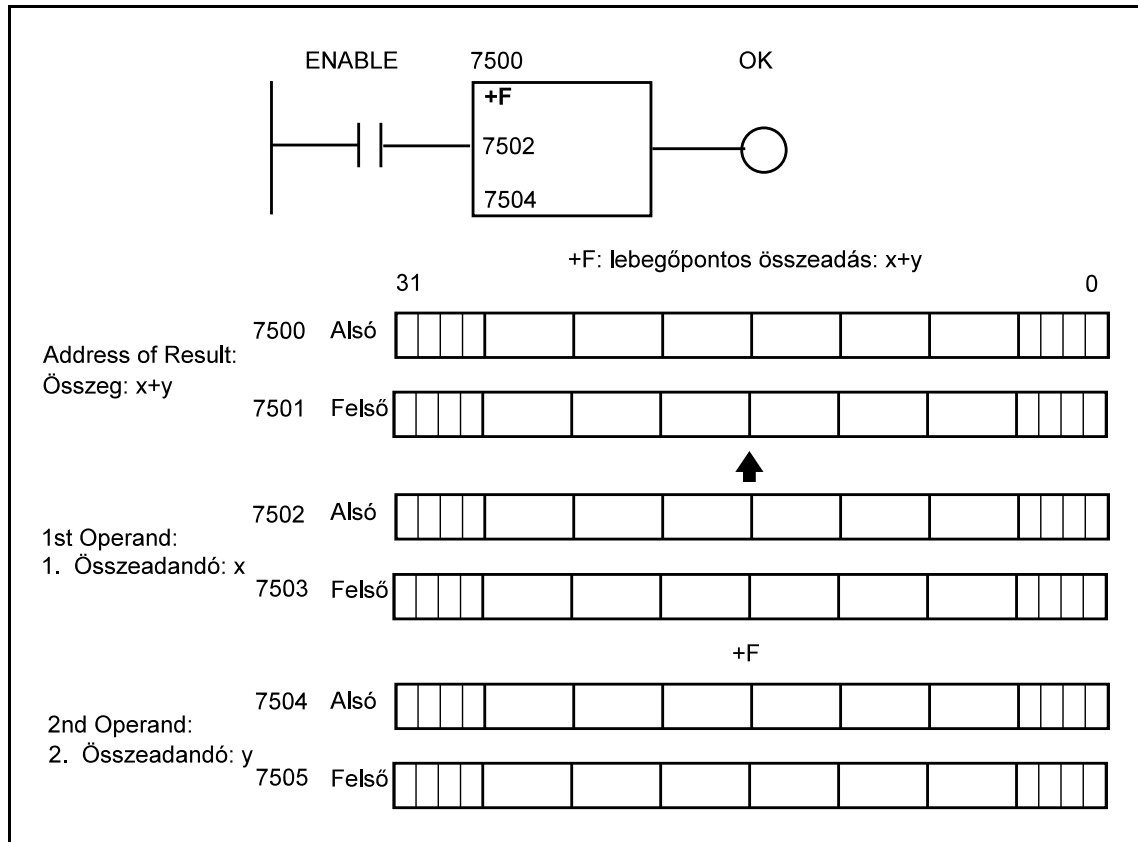
Az összeg az Address of Result címen megadott regiszterbe kerül.

– **1st Operand**

az 1. összeadandó értéke

– **2nd Operand**

a 2. összeadandó értéke.



6.10.2 Lebegőpontos kivonás: -F

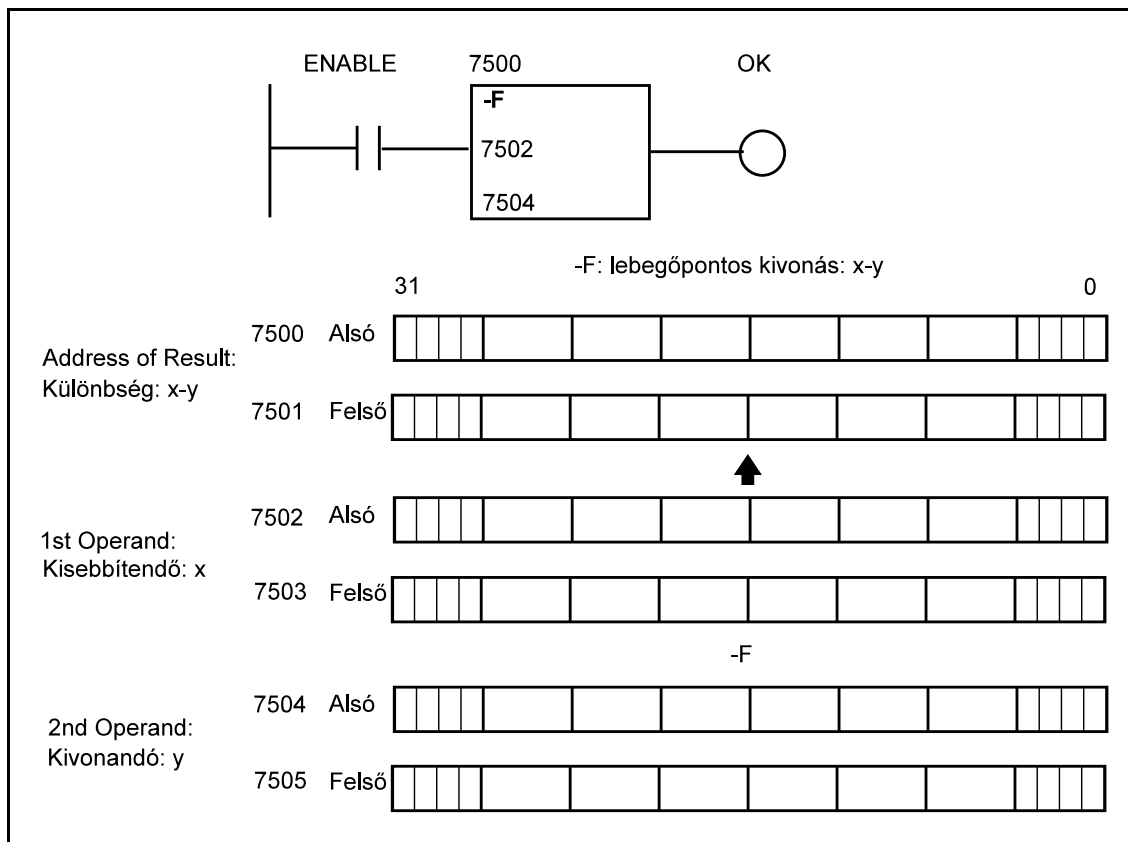
A különbség az Address of Result címen megadott regiszterbe kerül.

– **1st Operand**

a kisebbbítendő értéke

– **2nd Operand**

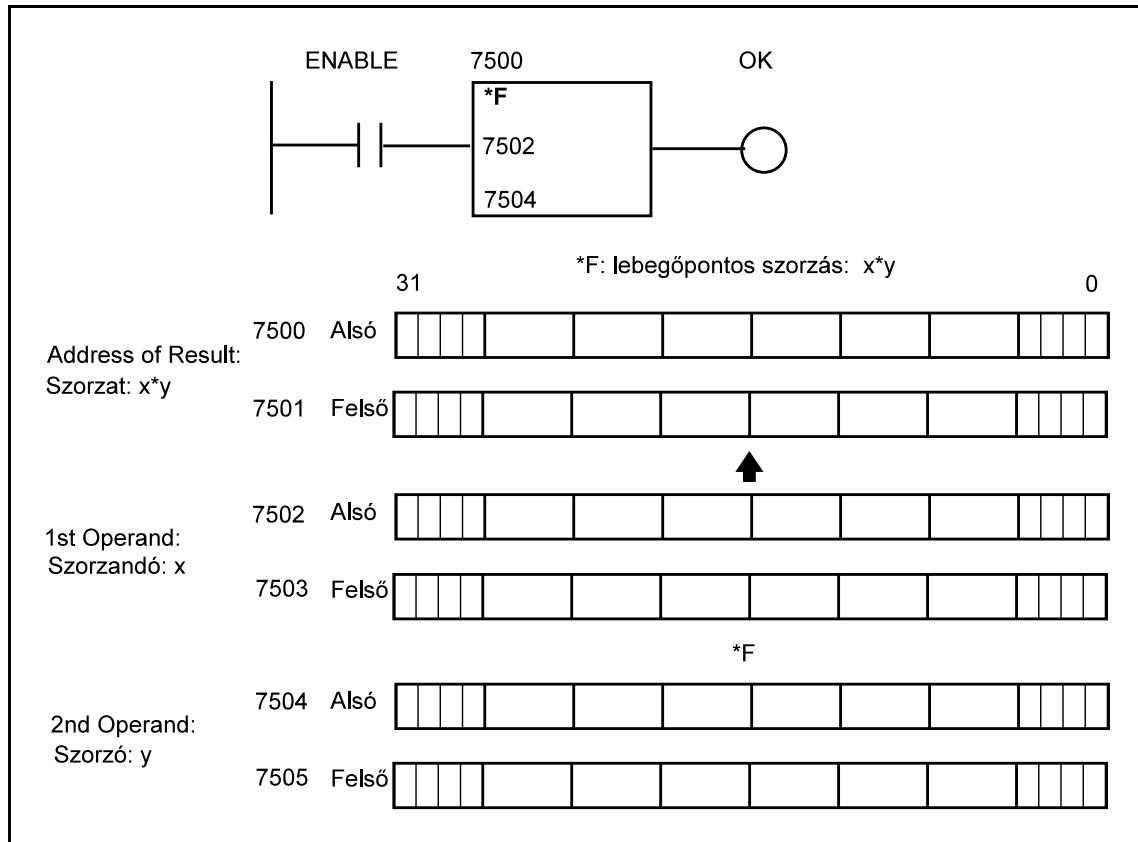
a kivonandó értéke.



6.10.3 Lebegőpontos szorzás: *F

A szorzat az Address of Result címen megadott regiszterbe kerül.

- **1st Operand**
a szorzandó értéke
- **2nd Operand**
a szorzó értéke.



6.10.4 Lebegőpontos osztás: /F

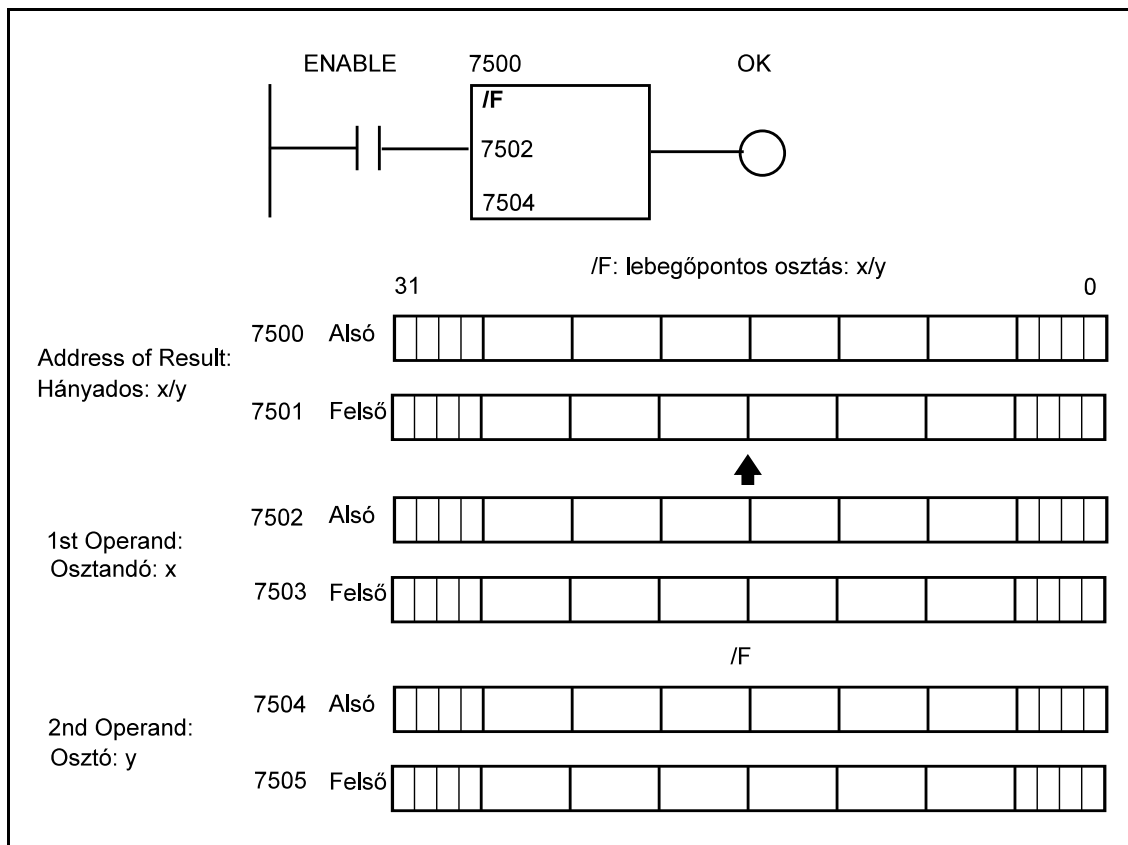
A hányados az Address of Result címen megadott regiszterbe kerül.

– **1st Operand**

az osztandó értéke

– **2nd Operand**

az osztó értéke.



6.10.5 Hatványozás: PWR

A hatvány az Address of Result címen megadott regiszterbe kerül.

– **Base**

az alap értéke

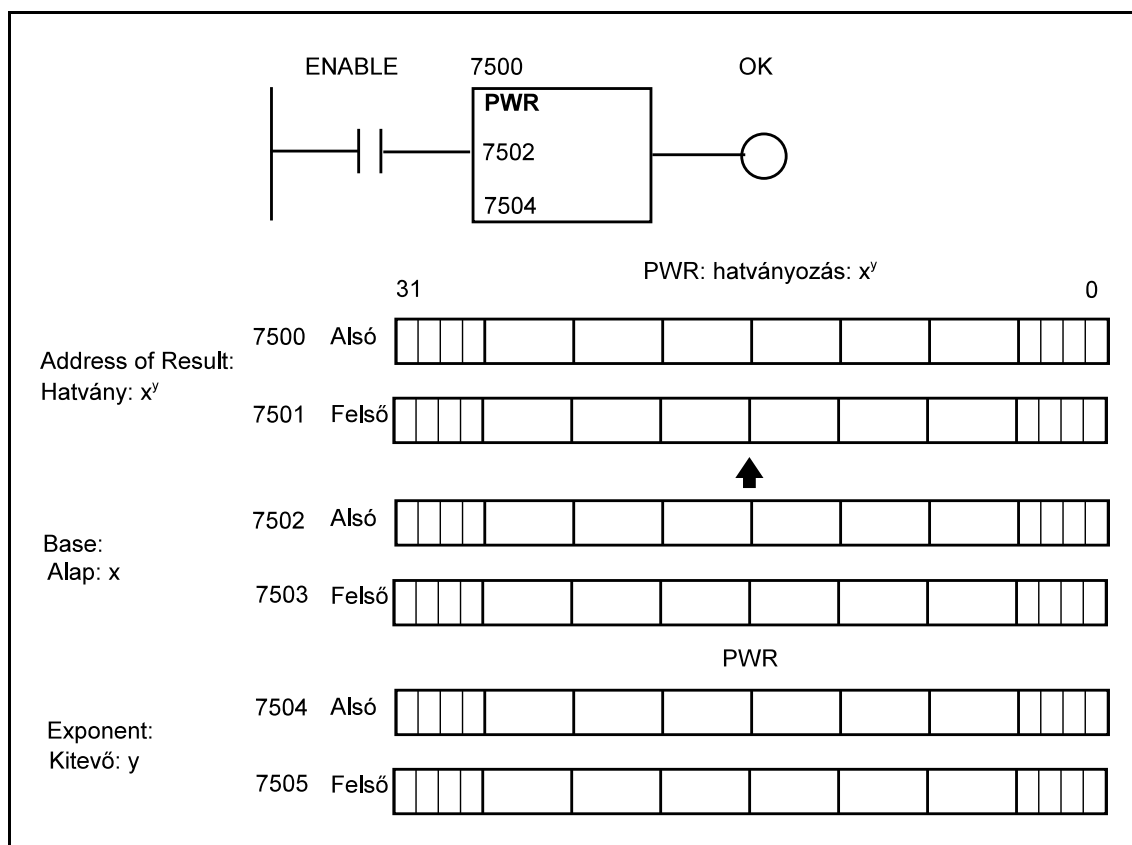
– **Exponent**

a kitevő értéke.

☞ **Figyelem:** Az alap értékére igaznak kell lennie a következőknek:

Base ≥ 0

Ha a fenti feltétel nem teljesül, az utasítást nem hajtja végre és a hibajelzőt FL_ER=1 IGAZra állítja.



6.10.6 Négyzetgyök: SQRT

A négyzetgyök az Address of Result címen megadott regiszterbe kerül.

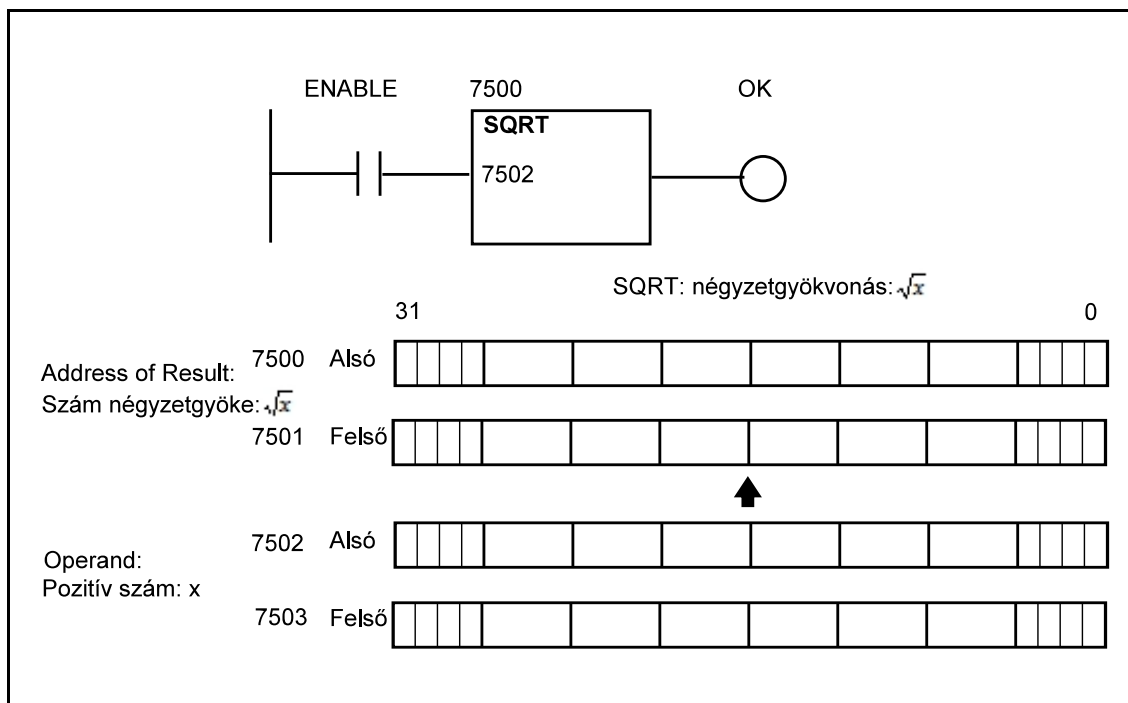
– **Operand**

az a szám amiből négyzetgyököt kell vonni.

☞ **Figyelem:** Az Operand értékére igaznak kell lennie a következőknek:

Operand ≥ 0

Ha a fenti feltétel nem teljesül, az utasítást nem hajtja végre és a hibajelzőt FL_ER=1 IGAZra állítja.

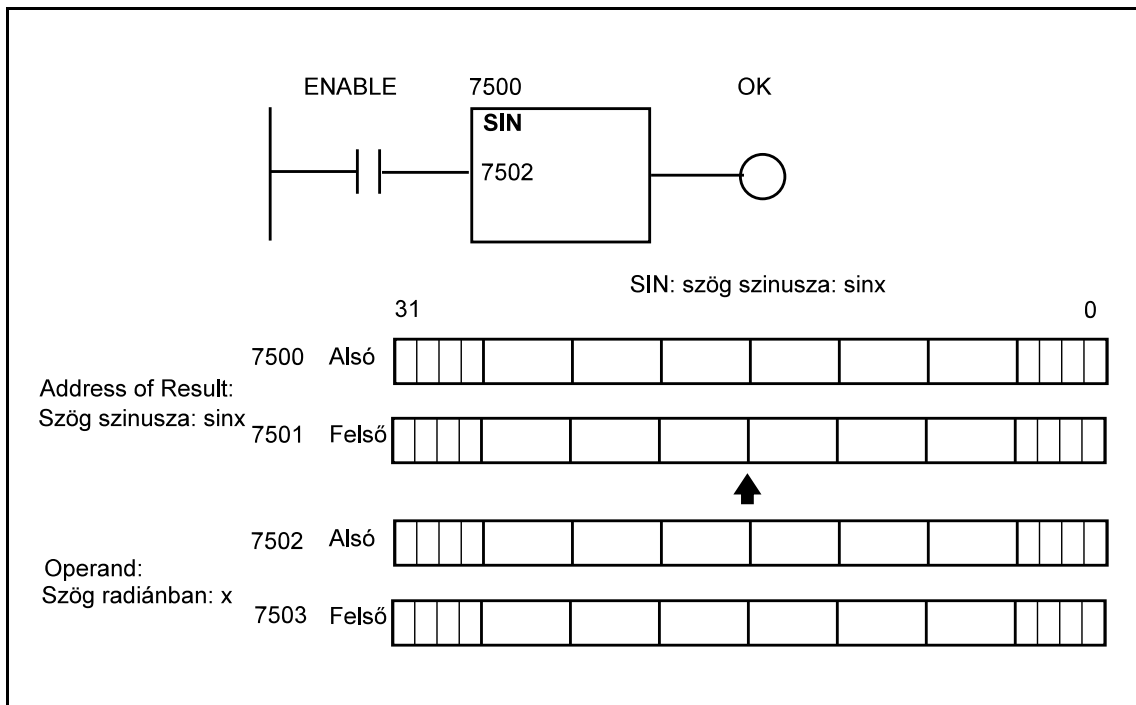


6.10.7 Szinusz: SIN

A szög szinusa az Address of Result címen megadott regiszterbe kerül.

– **Operand**

az a szögérték *radiánban*, aminek a szinuszt vesszük.

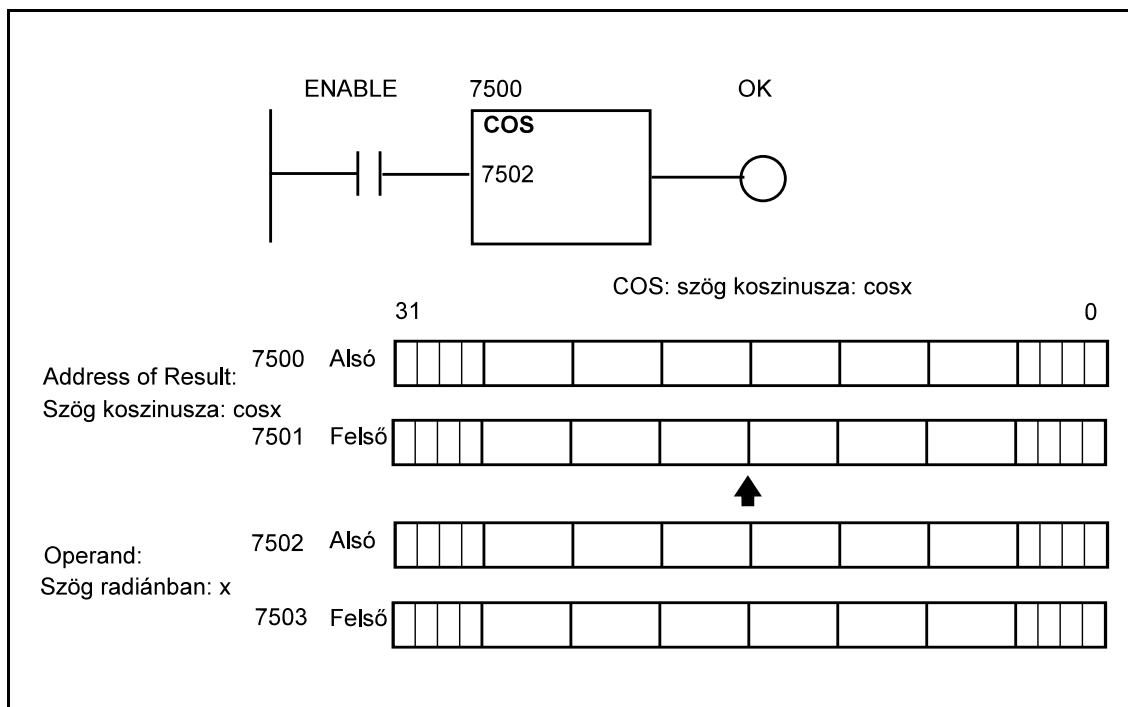


6.10.8 Koszinusz: COS

A szög koszinusza az Address of Result címen megadott regiszterbe kerül.

– **Operand**

az a szögérték *radiánban*, aminek a koszinuszát vesszük.



6.10.9 Tangens: TAN

A szög tangense az Address of Result címen megadott regiszterbe kerül.

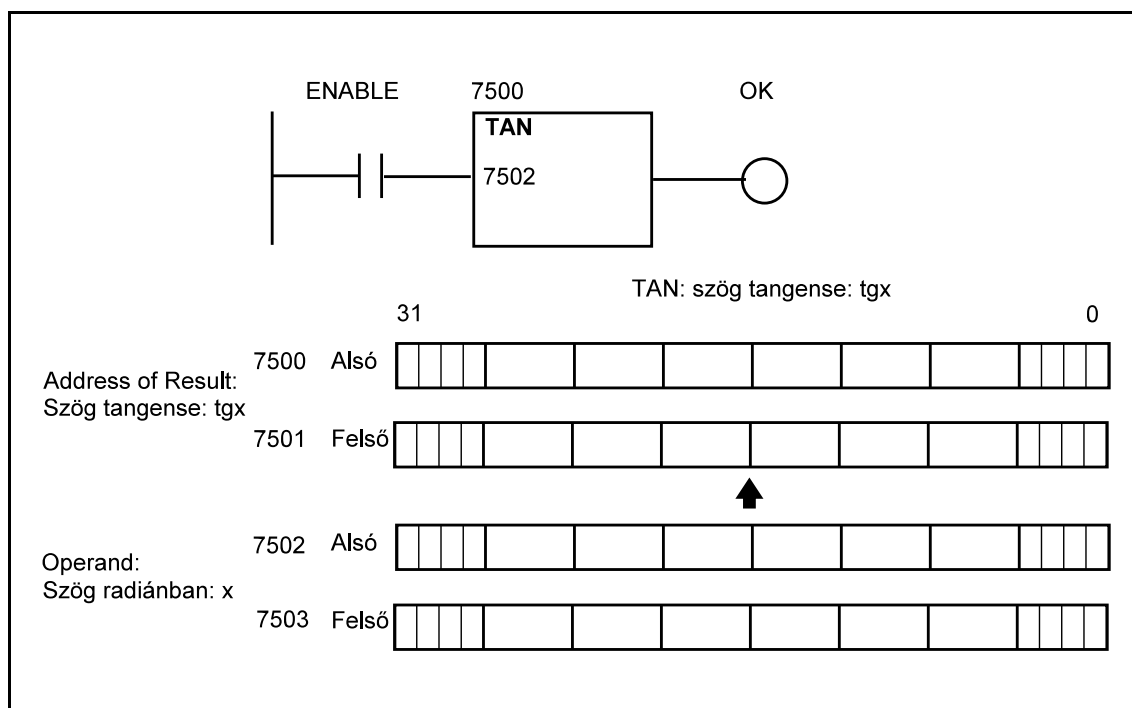
– **Operand**

az a szögérték *radiánban*, aminek a tangensét vesszük.

☞ **Figyelem:** Ha az Operand értéke:

$$\text{Operand} = \pi/2$$

az utasítást nem hajtja végre és a hibajelzőt FL_ER=1 IGAZra állítja.



6.10.10 Arkusz szinusz: ASIN

A szám arkusz szinusa az Address of Result címen megadott regiszterbe kerül *radiánban*.

– Operand

az a szám, aminek az arkusz szinuszát vesszük.

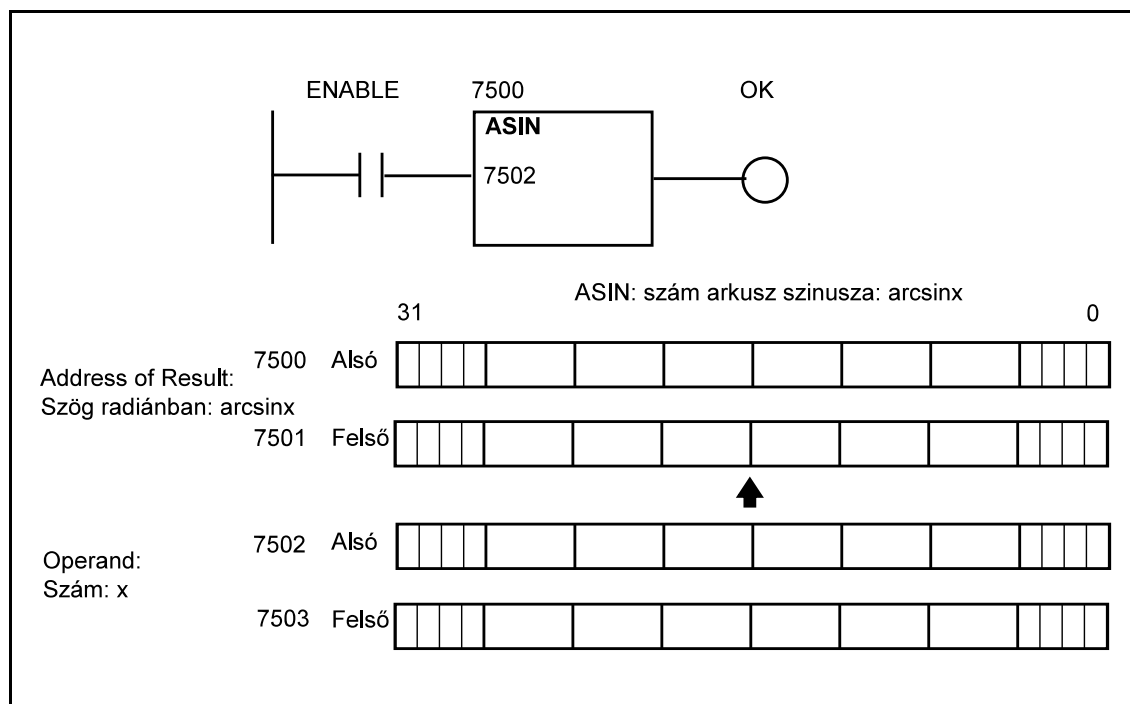
☞ **Figyelem:** Az Operand értékére igaznak kell lennie a következőknek:

$$-1 \leq \text{Operand} \leq 1$$

Ha a fenti feltétel nem teljesül, az utasítást nem hajtja végre és a hibajelzőt FL_ER=1 IGAZra állítja.

Az eredményt az alábbi szögtartományban kapjuk:

$$-\pi/2 \leq \text{Result} \leq \pi/2$$



6.10.11 Arkusz koszinusz: ACOS

A szám arkusz koszinusza az Address of Result címen megadott regiszterbe kerül **radiánban**.

– **Operand**

az a szám, aminek az arkusz koszinuszát vesszük.

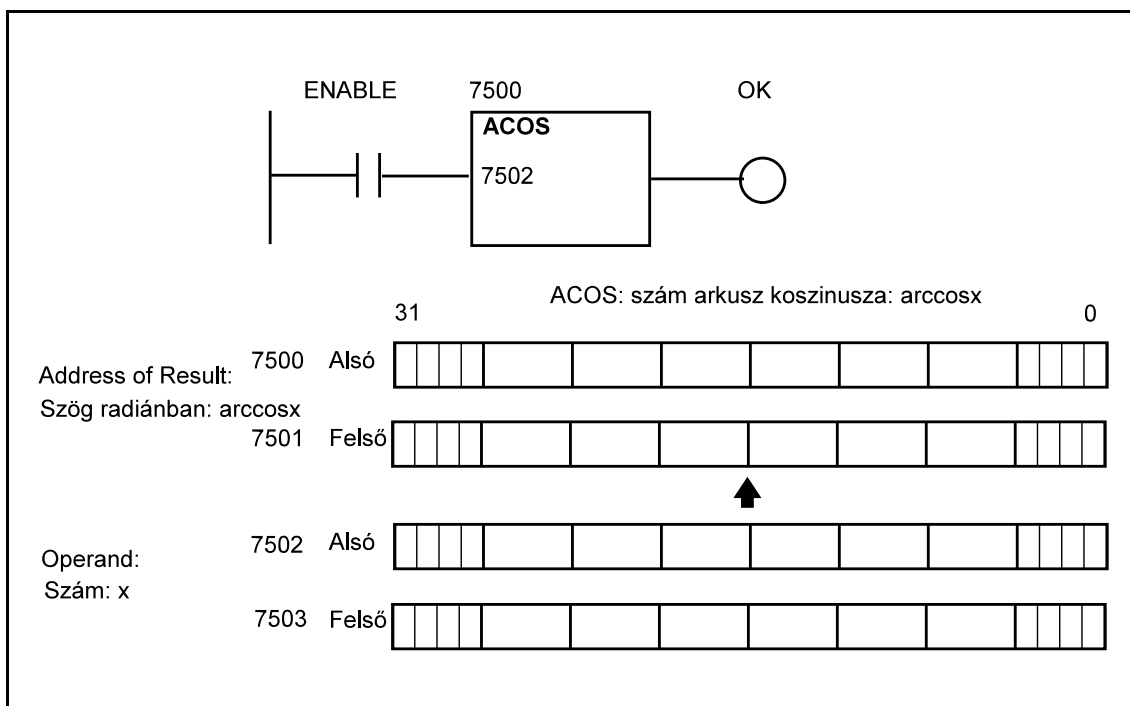
☞ **Figyelem:** Az Operand értékére igaznak kell lennie a következőknek:

$$-1 \leq \text{Operand} \leq 1$$

Ha a fenti feltétel nem teljesül, az utasítást nem hajtja végre és a hibajelzőt FL_ER=1 IGAZra állítja.

Az eredményt az alábbi szögtartományban kapjuk:

$$0 \leq \text{Result} \leq \pi$$



6.10.12 Arkusz tangens: ATAN

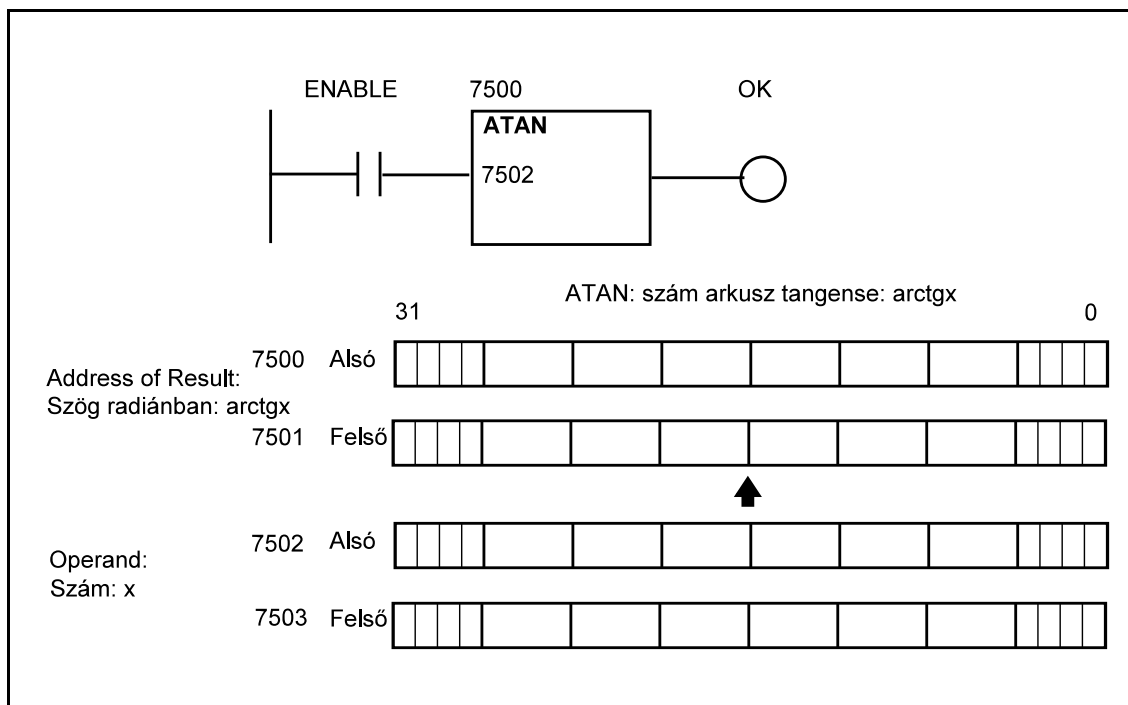
A szám arkusz tangense az Address of Result címen megadott regiszterbe kerül *radiánban*.

– **Operand**

az a szám, aminek az arkusz tangensét vesszük.

Az eredményt az alábbi szögtartományban kapjuk:

$$-\pi/2 \leq \text{Result} \leq \pi/2$$



6.10.13 Természetes alapú (e) hatvány: EXP

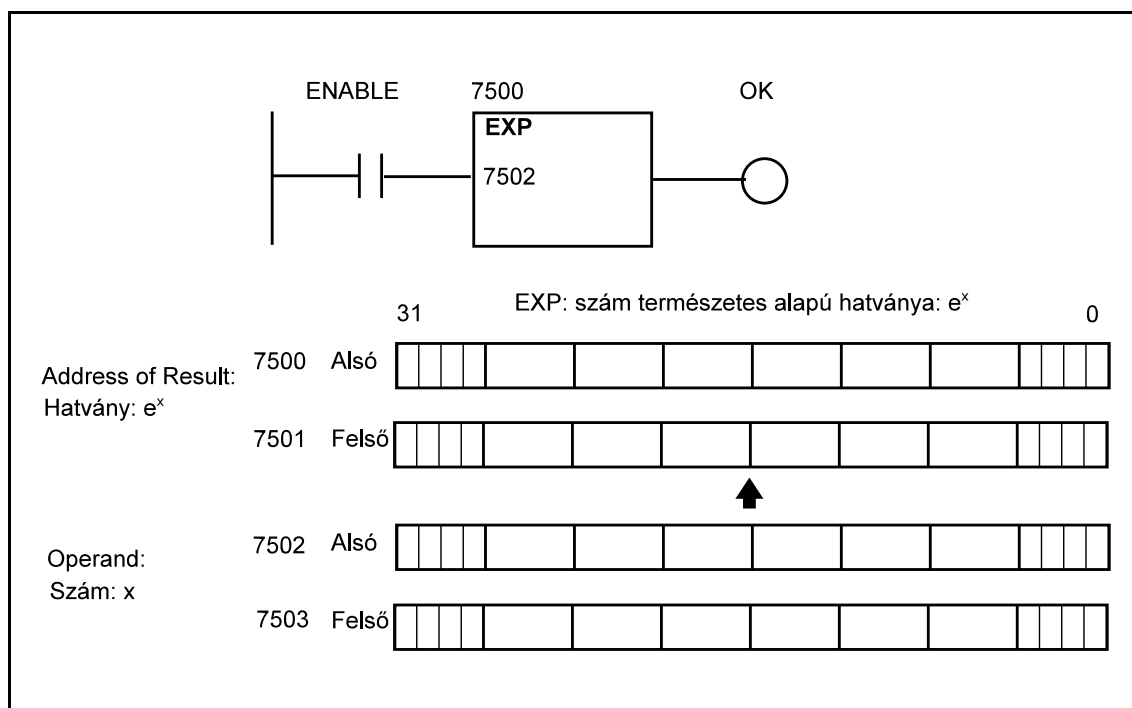
A szám természetes alapú ($e=2.718282\dots$) hatványa az Address of Result címen megadott regiszterbe kerül.

– **Operand**

az a szám, aminek a természetes alapú hatványát vesszük.

Az eredményt az alábbi tartományban kapjuk:

$$0 < \text{Result}$$



6.10.14 Természetes alapú (e) logaritmus: LOG

A szám természetes alapú ($e=2.718282\dots$) logaritmusa az Address of Result címen megadott regiszterbe kerül.

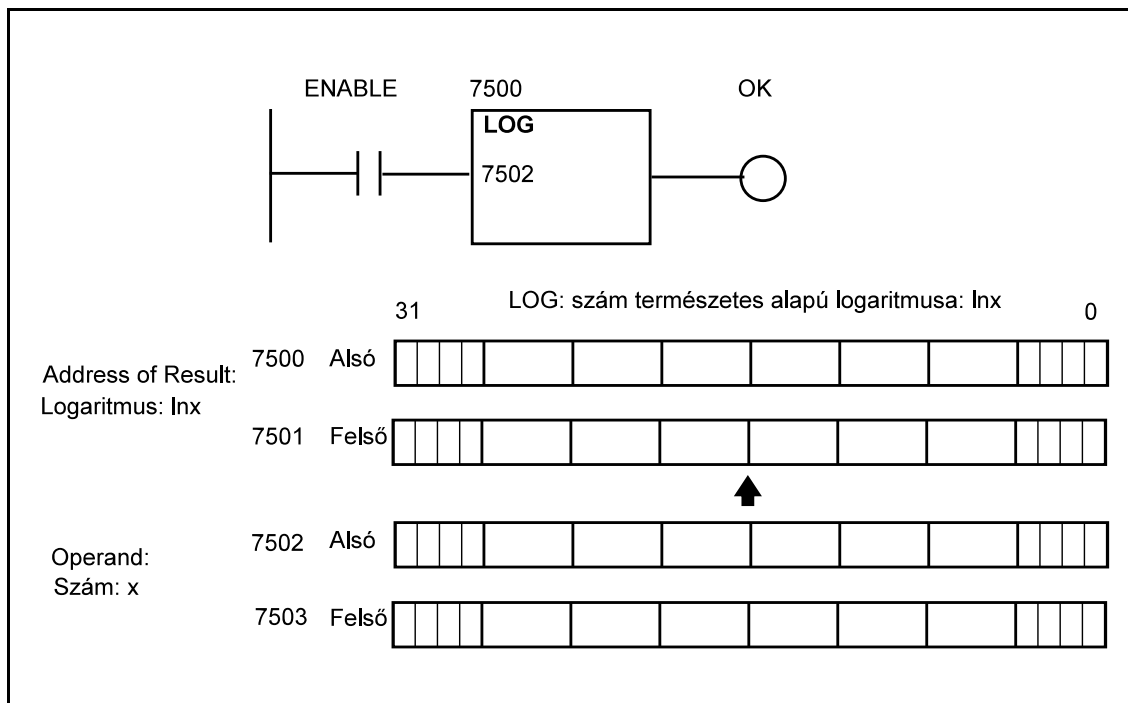
– **Operand**

az a szám, aminek a természetes alapú logaritmusát vesszük.

☞ **Figyelem:** Az Operand értékére igaznak kell lennie a következőknek:

$$0 < \text{Operand}$$

Ha a fenti feltétel nem teljesül, az utasítást nem hajtja végre és a hibajelzőt FL_ER=1 IGAZra állítja.



6.11 Konverziós utasítások

Az konverziós utasításokkal adatokat konvertálhatunk át egyik formátumból egy másikba.

A konverziós utasítások ki-, és bemenete

Minden konverziós utasításnak van egy

engedélyező bemenete.

Az engedélyező bemenet igaz állapotában végzi el a konverziót.

A konverziós utasításoknak *konfigurálhatunk*

kimenetet.

A kimenet IGAZ állapotot vesz fel, ha az utasítás bemenete IGAZ és a konverziós műveletet végrehajtotta. A kimenet HAMIS állapotot vesz fel, ha a bemenete HAMIS, vagy, ha az utasítás nem hajtható végre, azaz ha az utasítás beállítja az FL_ER hibajelzőt!

A kimenetre akár újabb utasítást is köthetünk.

Az összes konverziós utasításnak közösek a bemenő paraméterei.

A konverziós utasítások bemenő paraméterei

– **Address of Result:**

Értékhatár: 0...9999

A célregiszter címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

A konvertálandó regiszter tartalma erre a címre kerül.

– **Operand:**

A konvertálandó regiszter címe.

A regisztercím lehet szimbólikus, szám, vagy indexelt.

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

– **Remark:**

Ha a kimenet nincs engedélyezve az ide írt megjegyzés kommentként kerül megjelenítésre, illetve, ha nem írunk ide semmit, az Address of Result mezőben megadott szimbólum kommentjét írja ide.

6.11.1 BCD szám bináris számmá alakítás: BIN

Az utasítás operandusának kötelezően BCD számnak kell lennie.

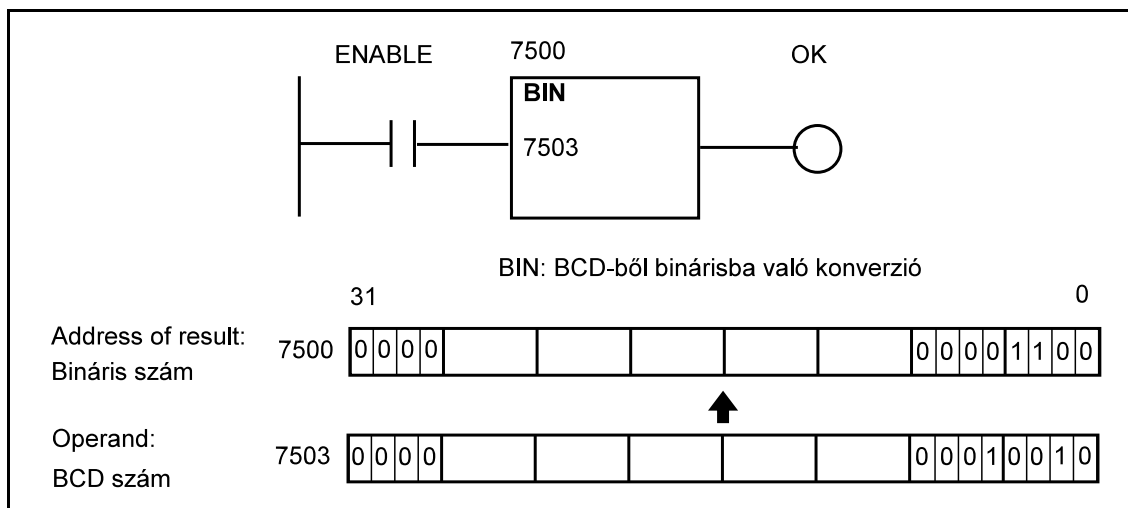
A legnagyobb átalakítható BCD szám: #99999999, azaz 8 decimális számjegyen ábrázolható.

👉 **Figyelem:**

Az utasítás nem hajtódik végre és az FL_ER hibajelzőt IGAZra állítja, ha a konvertálandó szám nem BCD. Például a

#\$000002AF

szám nem BCD!



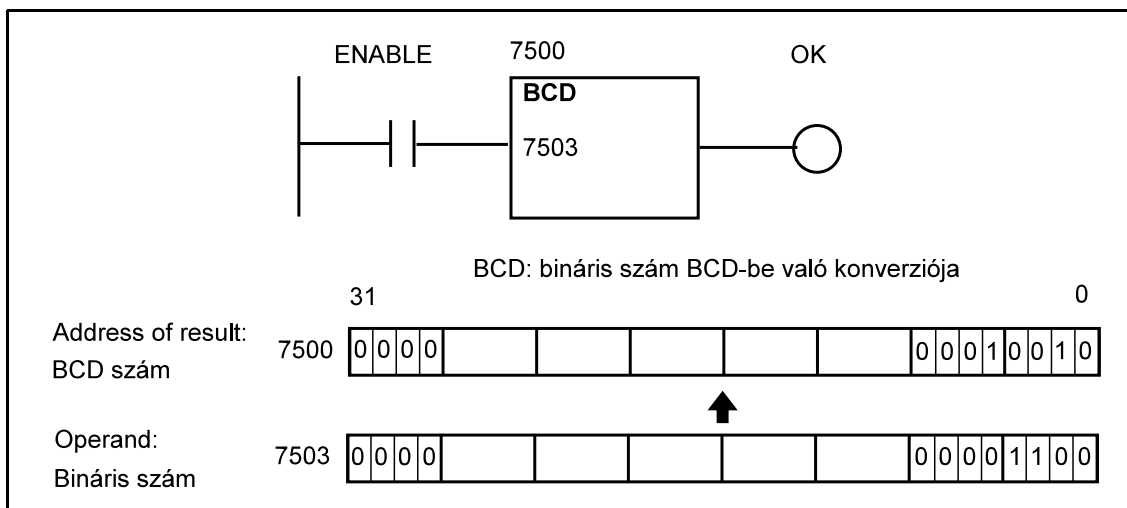
6.11.2 Bináris szám BCD számmá alakítása: BCD

A legnagyobb bináris szám, amely BCD számmá konvertálható: #05F5E0FF.

Ez a szám megfelel a #99999999 8 decimális számjegyű számnak.

Figyelem:

Az utasítás nem hajtódik végre és az FL_ER hibajelzőt IGAZra állítja, ha a konvertálandó bináris szám nem alakítható át 8 számjegyű BCD számmá. Például a #2FFFFFFF szám nem alakítható át!



6.11.3 Fixpontos szám lebegőpontos számmá alakítása: FLT

Figyelem!

Az eredmény az
Address of Result
Address of Result+1
 címeken kerül letárolásra!

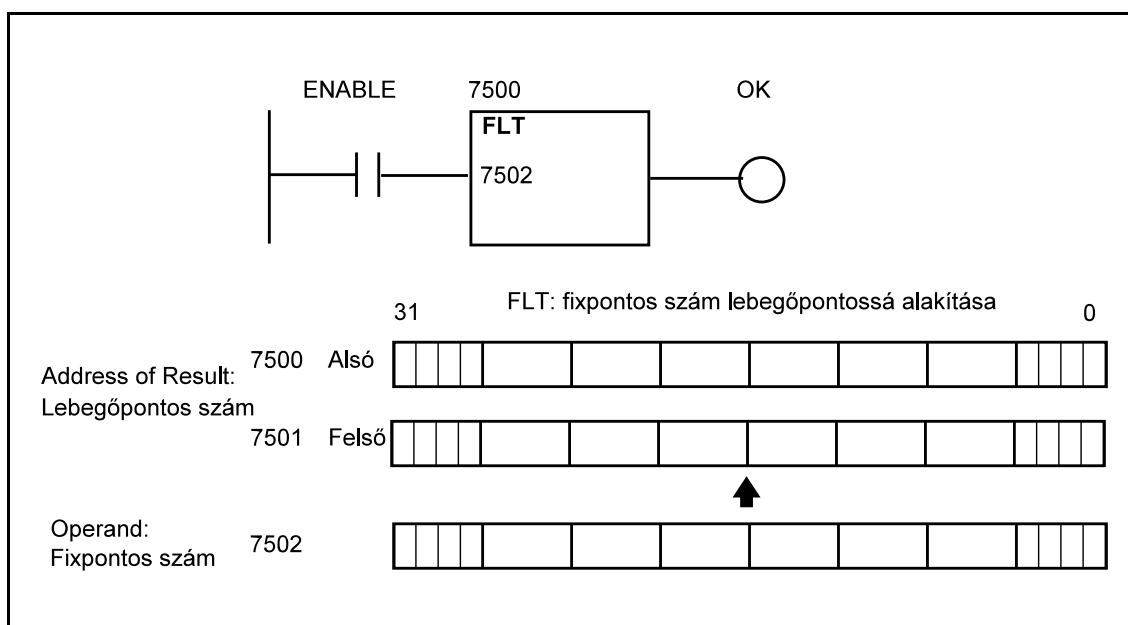
Az Operand bemenő paraméteren megadott adatot előjeles egész számként kezeli, ezért konverzió után is előjelhelyesen kapjuk az eredményt. Például a

#\$FFFFFFFF

szám lebegőpontoszá konvertálása után az eredmény:

-1.000

lesz.



6.11.4 Lebegőpontos szám fixpontos számmá alakítása: FIX

Figyelem!

*A konvertálandó lebegőpontos számot az utasítás az
Operand
Operand+1
címeiről veszi!*

A konvertálható lebegőpontos számok halmaza a következő:

$-2147483648.000... \leq \text{Operand} \leq 2147483647.000...$

Ha a konvertálandó lebegőpontos szám kívül esik a fent megadott tartományon, az utasítást nem hajtja végre és az FL_ER jelzőt IGAZra állítja.

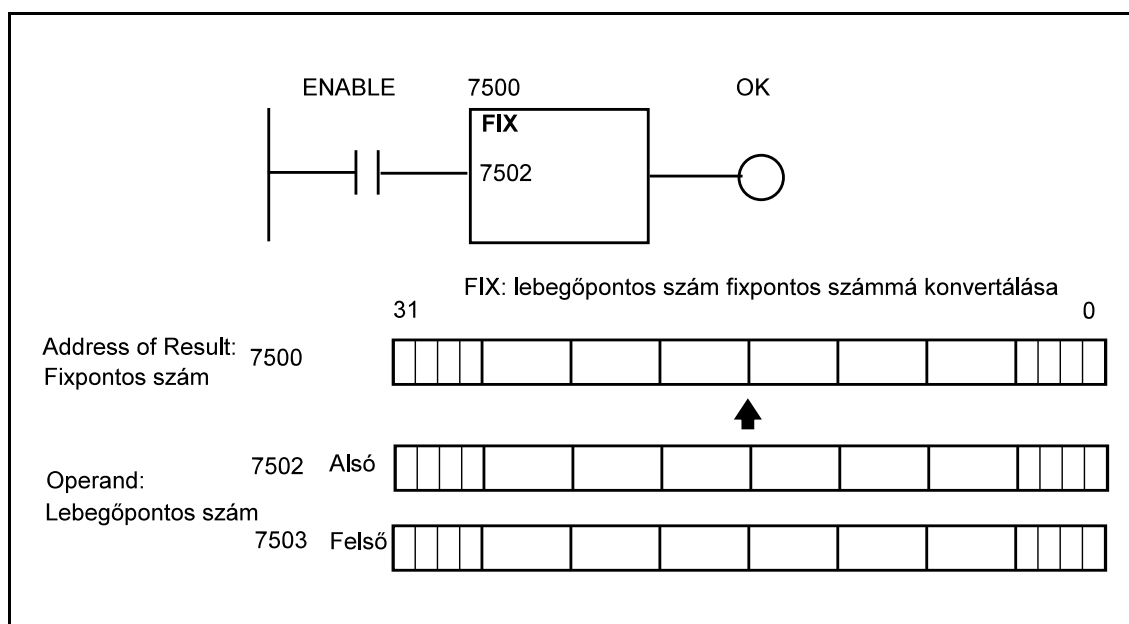
Az Address of result címen kapott fixpontos szám előjelhelyes lesz. Például a

-1.000

lebegőpontos szám fixpontossá konvertálása után az eredmény:

#\$FFFFFFF

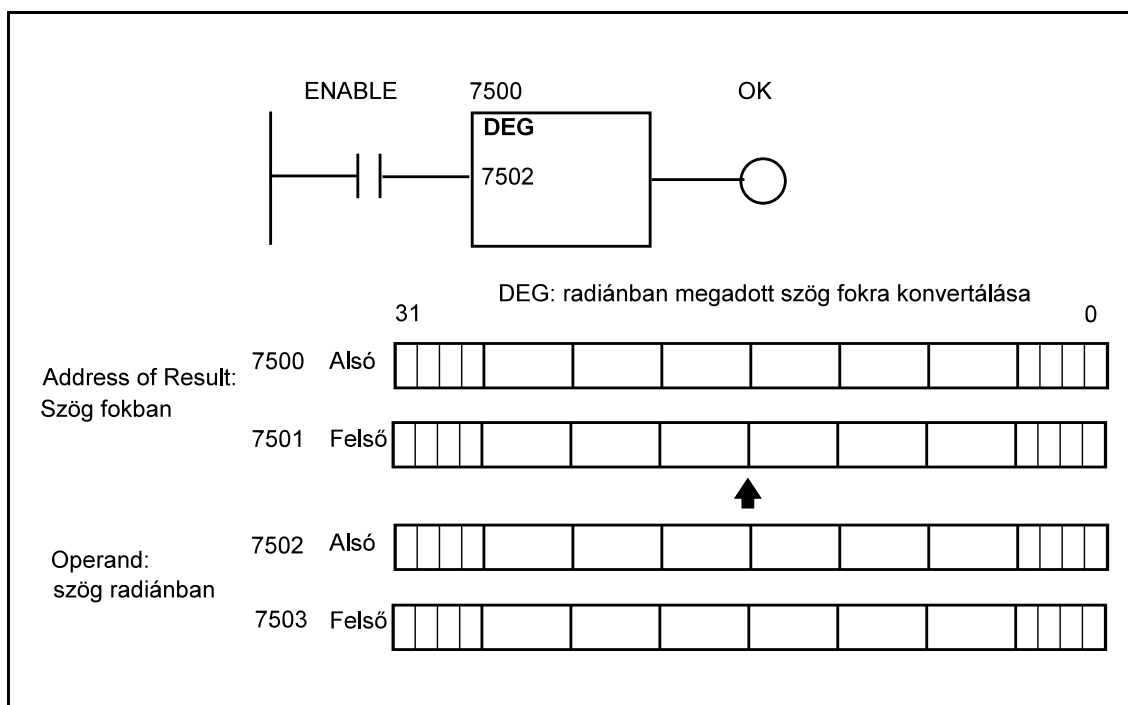
lesz.



6.11.5 Radiánban megadott szög fokra konvertálása: DEG

Figyelem!

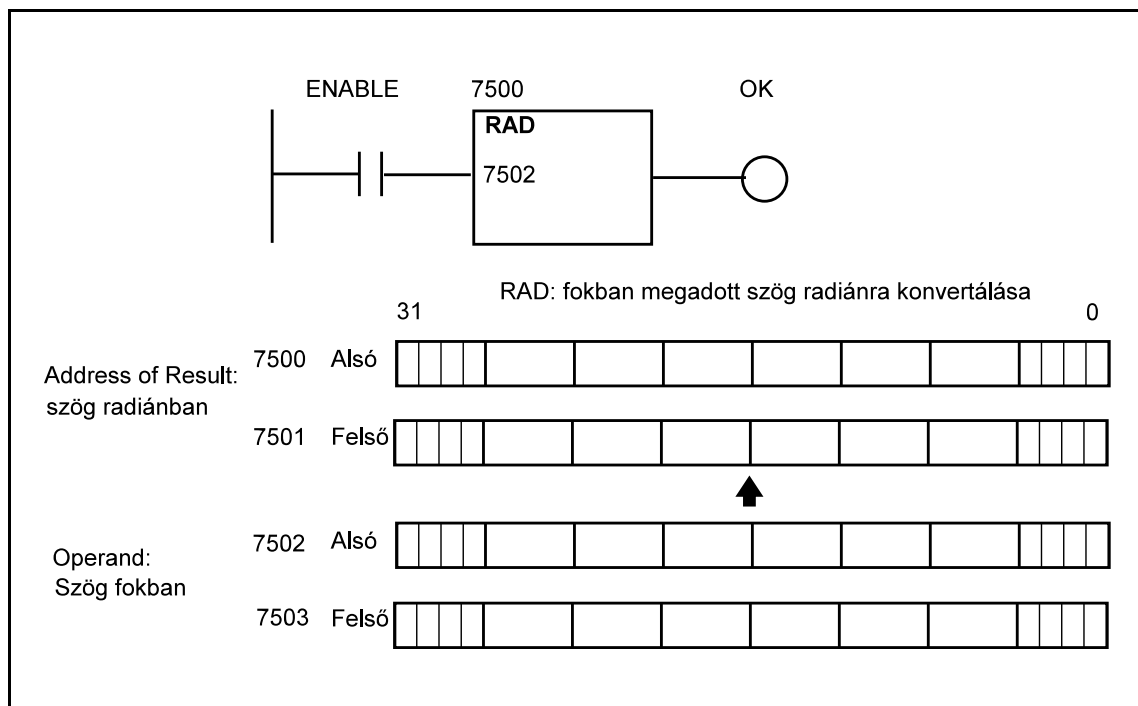
*A radiánban megadott szöget lebegőpontos számként az utasítás az
Operand
Operand+1
címekekről veszi!
Az fokban kapott eredmény az
Address of Result
Address of Result+1
címekeken kerül letárolásra!*



6.11.6 Fokban megadott szög radiánra konvertálása: RAD

Figyelem!

*A fokban megadott szöget lebegőpontos számként az utasítás az
 Operand
 Operand+1
 címekről veszi!
 Az radiánban kapott eredmény az
 Address of Result
 Address of Result+1
 címeken kerül letárolásra!*



6.12 Összehasonlító utasítások

Az összehasonlító utasításokkal két 32 bites, előjeles, fixpontos, vagy két 64 bites lebegőpontos értéket tudunk összehasonlítani.

6.12.1 A CMP és FCMP utasítások

A

CMP utasítás 32 bites, **előjeles, fixpontos** számok, az
FCMP utasítás pedig 64 bites **lebegőpontos** számok
összehasonlítását végzi.

Mindkét utasításnak egy

statikus engedélyező bemenete van.

Egyik utasításnak **sincs kimenete**.

Az összehasonlítás eredményét az

FL_GT nagyobb, mint (>), az

FL_EQ egyenlő (=), és az

FL_LT kisebb, mint (<)

jelzők kiértékelésével állapíthatjuk meg.

A CMP és FCMP utasítások bemenő paraméterei

– **Address of Value:**

Értékhatár: 0...9999

Az összehasonlítandó bal oldali érték címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

– **Data:**

Az összehasonlítandó jobb oldali érték.

CMP utasítás esetén előjeles, fixpontos szám,

FCMP utasítás esetén lebegőpontos szám.

Mindkét esetben hivatkozhatunk az adatra regiszterhivatkozással. Indexelt megadás lehetséges.

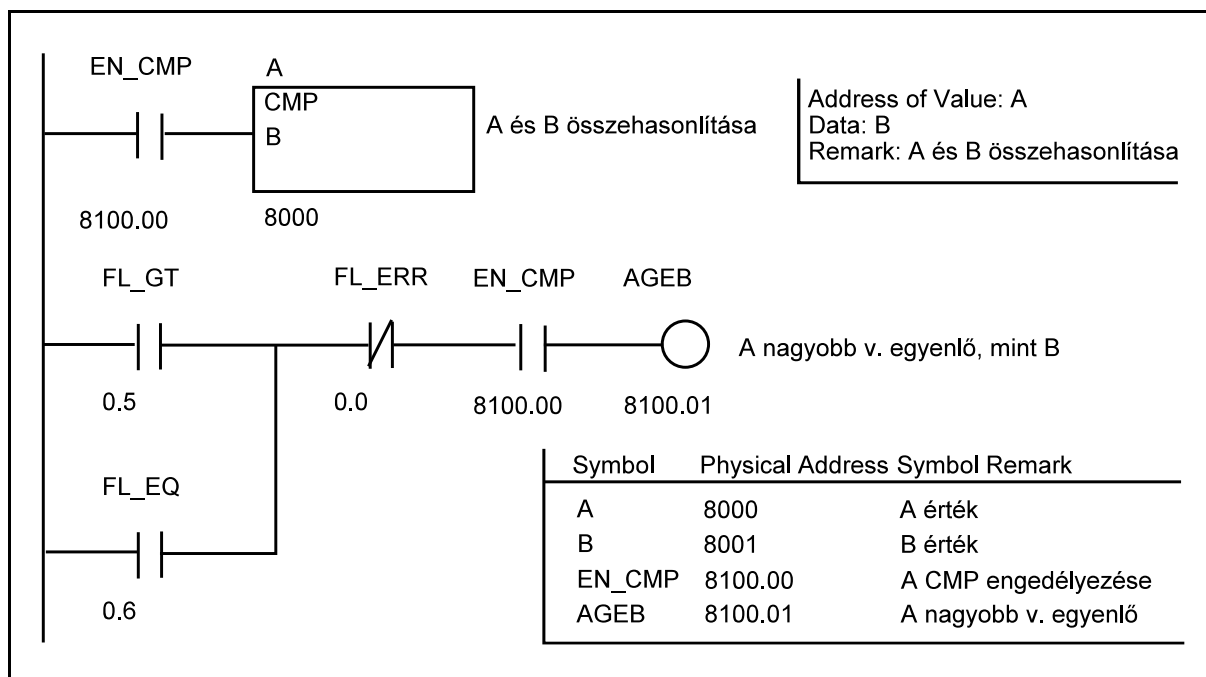
☞ **Figyelem:**

Az FCMP utasítás esetén az Address of Value és az Address of Value+1 címről veszi a lebegőpontos adatot. Ugyanez vonatkozik a Data paraméterre, ha regiszterhivatkozást írunk bele.

☞ **Figyelem!**

Ha a megadott címek kívül esnek az értelmezhető tartományon az utasítás beállítja az FL_ER hibajelzőt.

Az alábbi ábra a CMP utasítás használatát mutatja be az $A \geq B$ feltétel kiértékelése kapcsán.



6.12.2 Az érintkező típusú összehasonlító utasítások

Az utasításokat reléérintkezőkhöz hasonlóan használhatjuk a PLC programban. Ha az adatokra a feltétel teljesül, a kontaktus zárt, ha nem szakadt. Bármelyik utasítás lehet a létrafok első eleme is.

A 32 bites, **előjeles, fixpontos** számokon végezhető összehasonlító utasítások:

- CLT (<) kisebb, mint
- CLE (<=) kisebb, vagy egyenlő
- CEQ (=) egyenlő
- CNE (<>) nem egyenlő
- CGE (>=) nagyobb, vagy egyenlő
- CGT (>) nagyobb, mint

A 64 bites **lebegőpontos** számokon végezhető összehasonlító utasítások:

- FLLT (<) kisebb, mint
- FLLE (<=) kisebb, vagy egyenlő
- FLEQ (=) egyenlő
- FLNE (<>) nem egyenlő
- FLGE (>=) nagyobb, vagy egyenlő
- FLGT (>) nagyobb, mint

Mindegyik utasításnak egy

- statikus engedélyező bemenete**, és egy,
- a feltétel teljesülését (IGAZ állapotát) jelző **kimenete** van.

Az utasítás kimenete akkor is HAMIS lesz, ha a bemenete HAMIS, vagy, ha az utasítás nem hajtható végre, azaz ha az utasítás beállítja az FL_ER hibajelzőt!

Az utasítások bemenő paraméterei– **Address of Value:**

Értékhatár: 0...9999

Az összehasonlítandó bal oldali érték címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

– **Data:**

Az összehasonlítandó jobb oldali érték.

Az CLT, CLE, CEQ, CNE, CGE, CGT utasítások esetén előjeles, fixpontos szám,

Az FLLT, FLLE, FLEQ, FLNE, FLGE, FLGT utasítások esetén lebegőpontos szám.

Mindkét esetben hivatkozhatunk az adatra regiszterhivatkozással. Indexelt megadás lehetséges.

☞ **Figyelem:**

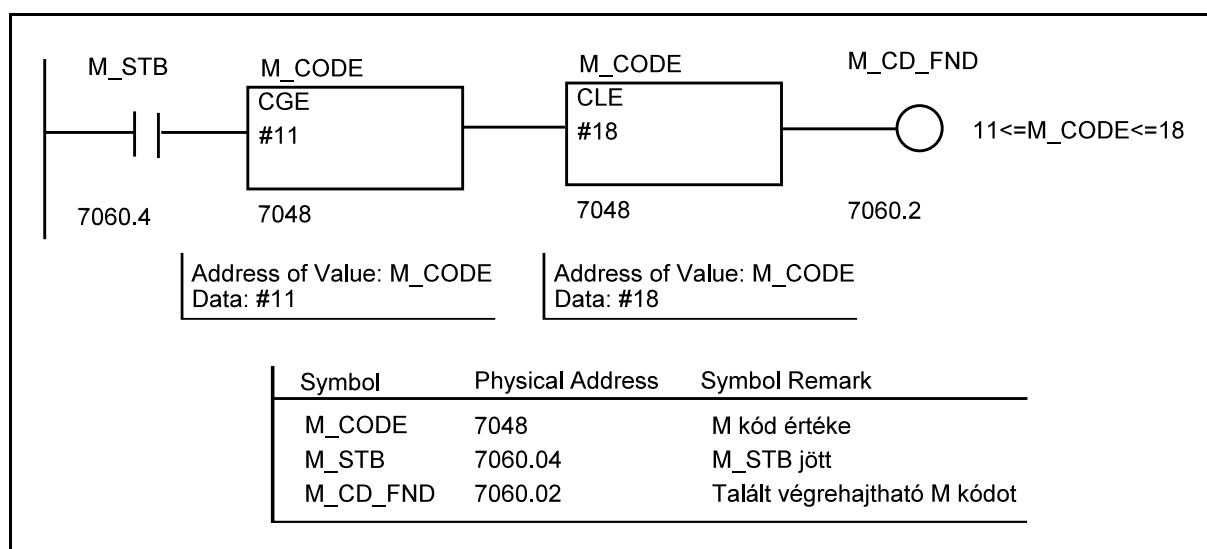
Az FLLT, FLLE, FLEQ, FLNE, FLGE, FLGT utasítások esetén az Address of Value és az Address of Value+1 címről veszi a lebegőpontos adatot. Ugyanez vonatkozik a Data paraméterre, ha regiszterhivatkozást írunk bele.

☞ **Figyelem!**

Ha a megadott címek kívül esnek az értelmezhető tartományon az utasítás kimenete HAMIS lesz, és beállítja az FL_ER hibajelzőt.

Az alábbiakban egy példa látható, amely az M_CD_FND tekercset akkor kapcsolja be, ha az alábbi feltétel teljesül az M_CODE változóra:

$$11 \leq M_CODE \leq 18$$



6.13 A PLC programból küldhető üzenetek

A PLC programból **üzenetek**et lehet küldeni a **képernyőre** és a **naplófájlba**. Vannak olyan utasítások, amelyek az üzeneteket csak a képernyőre küldik, vannak utasítások, amelyek csak a naplófájlba, és vannak olyanok, amelyek mind a képernyőre, mind a naplófájlba küldenek üzeneteket. A képernyőre kiírandó üzenetek egy üzenetpufferben gyűlnek és arra várnak, hogy valamilyen beavatkozás hatására onnan kitörlődjenek. Ez a beavatkozás lehet pl. a törlőgomb, start gomb, vagy egyszerűen csak az üzenet okának megszűnte is.

Az NC a legfontosabb eseményeket és hibákat egy ú.n. naplófájlban gyűjti. A naplófájlba küldött üzenetek és eseménybejegyzések kb. 1 hónapra visszamenően megőrződnek. Ebbe a naplófájlba írhat be a PLC program is üzeneteket.

A PLC **üzeneteknek** van **kódja** és **szövege**. Összesen 999 üzenetet lehet a PLC programból küldeni, amit az üzenet kódján belül az **üzenet száma** azonosít. A PLC üzenetek az NC által küldött globális üzenetek részét képezik.

Az üzenetek megjelenési sorrendje a képernyőn

Mindig az időben legutoljára küldött üzenet fog a képernyőn elsőként megjelenni: Last In First Out. Ha az összes üzenetet kiíratjuk az utolsó üzenet a lista tetején, az első a lista alján lesz.

Az üzenetek kódja

Minden globális üzenetet egy 8 jegyű, decimális szám azonosít. Ezt a számot nevezzük az üzenet kódjának.

Az **üzenetek kódja** a következőképp alakul ki:

AABCCDD

ahol:

AA: csatornaindex. Csatornafüggetlen üzeneteknél értéke 0, míg csatornafüggőeknél annak a csatornának a száma kerül bele, amelyikből az üzenetet küldték.

PLC üzenetnél a PLC programozó dönti el, hogy az üzenet csatornafüggetlen, vagy az üzenetet egy adott csatornához rendeli. Pl: a "Vészállapot" üzenet csatornafüggetlen, tehát az indexe AA=0, míg a "Főorsó forgást kérek" üzenet egy adott csatornából érkezik, tehát az indexe AA>0. Ez azért lényeges, mert ugyanazt az üzenetszöveget küldhetjük több csatornából is, viszont az üzenet kódjának egyértelműen azonosítania kell az üzenet küldőjét.

B: modulindex: annak a rendszer modulnak (mérőrendszer, interpolátor, mondatelőkészítő, stb.) az indexe, amely az üzenetet küldi.

PLC üzenetnél értéke B=0.

CCC: üzenet száma. Az üzenet száma 1-től 999-ig terjedhet.

DD: tetszőleges index: NC üzeneteknél ez lehet pl. tengely-, orsó- stb. index.

PLC üzenetnél a programozó határozza meg: pl. orsó-, vagy revolverindex, azaz, azt kódolja, hogy az üzenet hányadik orsóra, vagy hányadik revolverfejre vonatkozik. Ez azért lényeges, mert ugyanaz az üzenetszöveg vonatkozhat több orsóra, revolverre is, viszont az üzenet kódjának egyértelműen azonosítania kell az üzenet küldőjét.

A PLC program üzeneteküldő utasításainak osztályozása

PLC utasítás szintjén az üzeneteket osztályozhatjuk a következőképp:

- Kiíródnak a képernyőre, vagy nem:
kiíródnak a képernyőre: **MSG, MSGF, ALR, ALRF**
nem íródnak ki a képernyőre: **REM, REMF**
- Bekerülnek a naplófájlba, vagy nem:
bekerülnek a naplófájlba: **REM, REMF, ALR, ALRF**
nem kerülnek be a naplófájlba: **MSG, MSGF**

A PLC program írásánál ügyelni kell, hogy melyik üzenetet melyik utasítással küldjük.

Pl. egy “Ajtó nyitva” üzenetet ki kell írni a képernyőre, hogy tájékoztassa a kezelőt, miért nem megy a megmunkálás. Mivel az üzenet sokszor előfordulhat egy nap folyamán, de nem hibát, vagy működési problémát jelez, csak a kezelőt figyelmezteti, főleg a naplófájlba letenni. Tehát ebben az esetben az MSG utasítás használata indokolt.

Ha viszont az üzenet egy hiba, pl. egy motorvédő kikapcsolt állapota miatt következik be, az ALR utasítás használata célszerű, mert ez az üzenet kikerül a képernyőre és a naplófájlba is bejegyzésre kerül.

Ha viszont semmit sem akarunk a képernyőre kiírni, nem várunk semmilyen kezelői beavatkozást, de a naplófájlba be akarunk valamit jegyezni, a REM utasítást kell használni.

A **PLC üzenetekbe**, az üzenet szövegén belül tetszőleges helyre, opcionálisan **számértéket** is írhatunk. A szám lehet fix-, és lebegőpontos is. Az üzeneteket osztályozhatjuk úgy is, hogy az üzenetekbe fix-, vagy lebegőpontos adatot kívánunk-e közölni:

- **fixpontos** adatot tartalmazó üzenetek: **MSG, ALR, REM,**
- **lebegőpontos** adatot tartalmazó üzenetek: **MSGF, ALRF, REMF.**

A PLC üzenetek szövege

A PLC szerkesztő program támogatja az üzenet szövegek szerkesztését, azok a *.plc fájlban kerülnek tárolásra a PLC programmal együtt.

Az üzeneteket külön is lehet menteni *.mes néven a PLC szerkesztő segítségével, az Üzenetek Exportálása gombbal. Ugyanúgy már meglévő *.mes üzenetfájlt importálni lehet a PLC programba.

A *.mes szövegfájl kódolása UTF8. Ez azt jelenti, hogy a jelenleg ismert összes írásjel megadható benne.

PLC változó tartalmának szöveges kiírása az üzeneten belül

Az üzenet szövegében, tetszőleges helyen, ki lehet írni egy regiszter tartalmát. Azt, hogy a kiírásra kerülő szám fix-, vagy lebegőpontos-e a PLC utasítás dönti el. A szerkesztőben meg lehet adni, hogy hány tizedes pontossággal legyen a szám kiírva, ha a kiírandó szám lebegőpontos.

Pl. kiírathatjuk a szöveget, hogy “Tegye be a n. szerszámot az orsóba”, ahol “n” egy PLC regiszter értéke.

Az üzenet exportálása során a következő formátumot kapjuk:

Tegye be a {#0:F0}. szerszámot az orsóba

ahol a szövegben a {#0:F0} karaktorsor jelentése, hogy az MSG, ALR, vagy REM utasításban megadott fixpontos bináris adatot konvertálja decimális számmá és írja be a szövegbe az üzenetfeldolgozó. # jelentése: fixpontos szám és az F0 jelentése: ne használjon tizedespontot a kiíratásnál.

Ha lebegőpontos számot tartalmazó üzenetet exportálunk:

Az orsó hőmérséklete {*0:F1} °C

szöveget kapjuk, ahol a szövegben a {*0:F1} karaktorsor jelentése, hogy az MSGF, ALRF, vagy

REMF utasításban megadott lebegőpontos bináris adatot konvertálja decimális számmá és írja be a szövegbe az üzenetfeldolgozó 1 tizedesjegy pontossággal. * a lebegőpontos ábrázolást jelenti, F1 pedig azt jelenti, hogy a számot 1 tizedes pontossággal jelenítse meg.

A formázás szabályai:

{	(formázás kezdete)		
#	(fixpontos adat), vagy	*	(lebegőpontos adat)
0	(kötelezően)		
:	(elválasztó)		
F			
i	(megjeleníteni kívánt tizedesjegyek száma lebegőpontos ábrázolás esetén, fixpontosnál 0)		
}	(formázás vége)		

6.13.1 Az üzenetküldő utasítások: MSG, MSGF, ALR, ALRF, REM, REMF

Az **MSG, MSGF, ALR, ALRF** utasításokkal küldött PLC üzenetek **megjelennek a képernyőn**, az üzenetmezőben. A PLC programozó feladata annak eldöntése, hogy a kezelő milyen beavatkozásra törlődnek le ezek az üzenetek a képernyőről.

Az **MSG, MSGF** utasítások üzenetei **nem kerülnek be a naplófájlba**.

Az **ALR, ALRF** utasítások üzenetei **mindig bekerülnek a naplófájlba**.

A **REM, REMF** utasításokkal küldött PLC üzenetek **nem jelennek meg a képernyőn**, az üzenetmezőben, viszont mindig **bekerülnek a naplófájlba**.

Az **MSG, ALR, REM** utasítások opcionális bemenő változói **fixpontosak**, míg az **MSGF, ALRF, REMF** utasítások opcionális bemenő változói **lebegőpontosak**.

Az utasítások be-, kimenetei

Az összes üzenetküldő utasítás az **engedélyező bemenetén**

- felfutó élre bejegyzi az üzenet adatait a képernyőre vonatkozó üzenetpufferbe és a naplófájlba,
- lefutó élre pedig törli az üzenetet a képernyőre vonatkozó üzenetpufferből.

Ezeknek az utasításoknak **van kimenetük**: a kimenete IGAZ értéket vesz fel, ha az üzenet megjelenik a képernyőn, vagy REM, REMF utasítás esetén bekerült a naplóba.

Az MSG, MSGF, ALR, ALRF, REM, REMF utasítások bemenő paraméterei:

– **Message Number:**

értékhatár: 1,...999

az üzenet sorszáma

Fixpontos szám, vagy regiszter cím. Indexelt megadás lehetséges.

– **Channel Index:**

Értékhatár: 0...99

az üzenet csatornaindex.

Fixpontos szám, vagy regiszter cím. Indexelt megadás lehetséges.

– **Arbitrary Index:**

értékhatár: 0...99

az üzenet tetszőleges indexe.

Fixpontos szám, vagy regiszter cím. Indexelt megadás lehetséges. Megadhatunk pl. tengely-, orsó-, revolverfejszámot.

– **Optional Variable:**

értékhatár: MSG, ALR, REM utasítások esetén 32 bites szám, vagy regiszter címe,
MSGF, ALRF, REMF utasítások esetén lebegőpontos szám, vagy regiszter címe

az üzenet opcionális változója

Indexelt megadás lehetséges.

– **Remark:**

Megjegyzés

Az MSG, MSGF, ALR, ALRF, utasítások működése és kimenete:

Az utasítás akkor teszi be az üzenet szövegét az üzenetpufferbe, ha a bemenetén felfutó élet detektál.

Az utasítás akkor törli ki az üzenet szövegét az üzenetpufferből, ha a bemenetén lefutó élet detektál.

Az ALR, ALRF utasítás a bemenetén megjelenő felfutó élre a naplófájlba is bejegyzi az üzenet kódját és szövegét.

Az utasítások kimenete akkor lesz IGAZ, ha az utasításban szereplő üzenet kódja jelenik meg a képernyőn:

AA0CCCDD

ahol:

AA: a Channel Index utasításparaméter értéke,

0: az üzenetet a PLC küldte,

CCC: a Message Number utasításparaméter értéke,

DD: az Arbitrary Index utasításparaméter értéke.

A REM, REMF utasítások működése és kimenete:

A REM, REMF utasítás a bemenetén megjelenő felfutó élre a naplófájlba bejegyzi az üzenet kódját és szövegét.

Az utasítások kimenete akkor lesz IGAZ, ha az utasításban szereplő üzenet bekerült a naplófájlba, gyakorlatilag azonnal.

Példák:

Az S1 indexű orsóhoz tartozó megállási hiba jelző:

S_STOP_ERR

Az üzenet száma: #20

Az üzenet az S1_C csatornából érkezik, ahol az S1_C regiszter mondja meg, hogy az S1 orsó melyik csatornában van éppen.

Az üzenet az _S1 számú orsóra vonatkozik, ahol _S1 az S1 orsó száma.

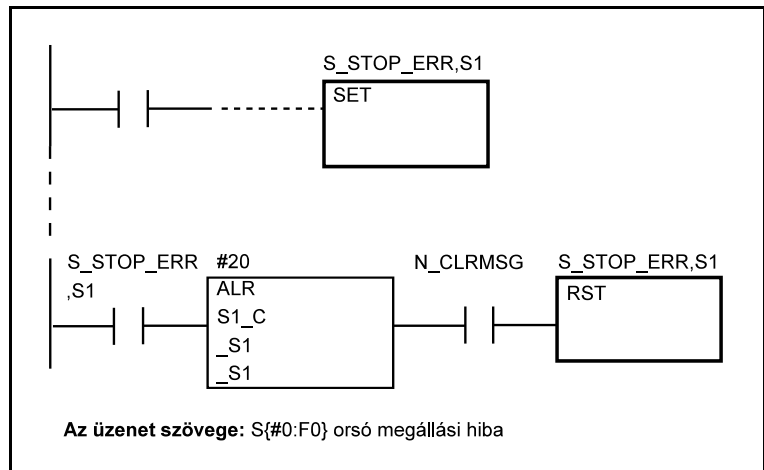
(Esetünkben az orsó indexe S1=0, az orsó száma _S1=1)

Az üzenet szövegébe is beírjuk az orsó számát: az opcionális változó értéke _S1.

Az üzenet teljesen ki van kódolva: a csatorna és az orsón számával. Az üzenet szövegébe is belevettük az orsó számát a jobb áttekinthetőség érdekében.

Az üzenet szövegének azonosítója 20. Ugyanezt az üzenetszöveget lehet használni a 2. 3. stb orsókra is, tehát új szöveget nem kell orsónként újra felvenni, csak az orsó számát kell változtatni. Ha az üzenetdoboz bemenetén IGAZ állapot jelenik meg az üzenet bekerül a naplófájlba és az üzenetpufferbe is.

Ha az üzenet a képernyőre kerül, az ALR utasítás kimenete IGAZ állapotba kerül. Ekkor a CANCEL gomb hatására működő N_CLRMSG jelző segítségével az üzenetet töröljük az S_STOP_ERR,S1 jelző törlésével.



A CH1 indexű csatornában start gomb nyomást kérünk az

ST_MB_REQ

jelzőn.

Az üzenet száma: #06

Az üzenetet az _CH1 számú csatornából kérjük.

(Esetünkben a csatorna indexe CH1=0, a csatorna száma _CH1=1.)

A tetszőleges index értéke #0.

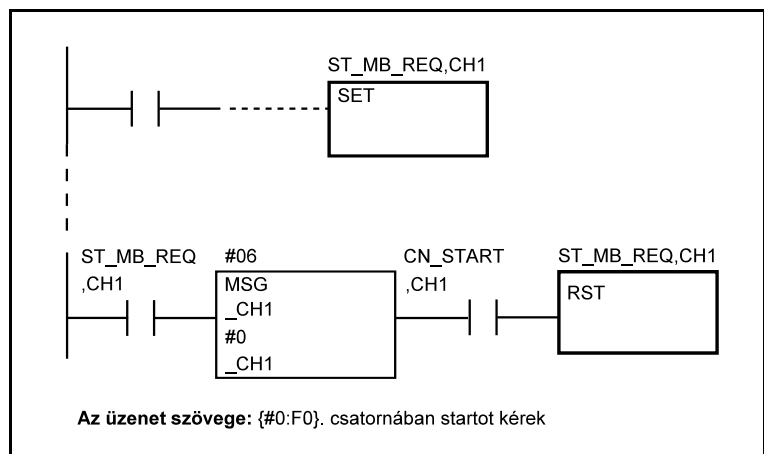
Az üzenet szövegébe is beírjuk a csatorna számát az opcionális változó használatával: _CH1.

Az üzenet teljesen ki van kódolva a csatorna számával. Az üzenet szövegébe is belevettük a csatorna számát a jobb áttekinthetőség érdekében.

Az üzenet szövegének azonosítója 06. Ugyanezt az üzenetszöveget lehet használni a többi csatornában is, tehát új szöveget nem kell felvenni, csak a csatorna számát kell változtatni.

Ha az üzenetdoboz bemenetén IGAZ állapot jelenik meg az üzenet bekerül az üzenetpufferbe, de a naplófájlba nem.

Ha az üzenet a képernyőre kerül, az MSG utasítás kimenete IGAZ állapotba kerül. Ekkor a



START gomb hatására az üzenetet töröljük az ST_MB_REQ,CH1 jelző törlésével.

A PLC előre meghatározott idő letelte után naplózza az orsó hőmérsékletét: beállítja a

LOG_TEMP

jelzőt.

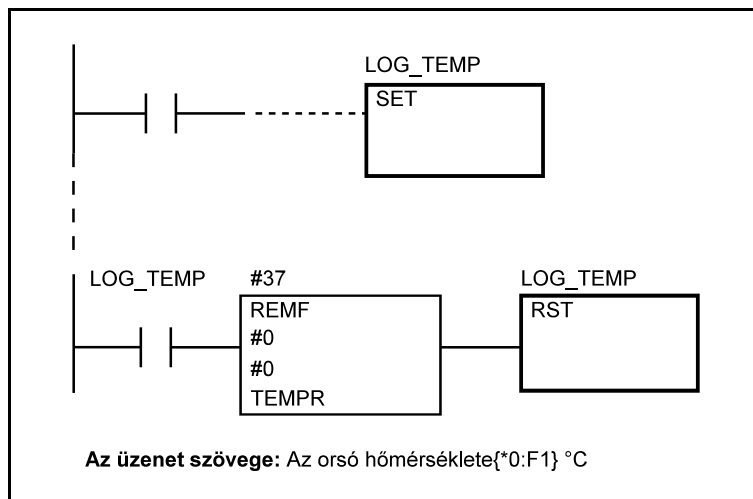
Az üzenet száma #37.

A gépen csak egy csatorna és orsó van, ezért a csatorna és orsóindex is #0.

A TEMPR címkejű regiszterben tároljuk az orsó hőmérsékletét lebegőpontos formában.

Az üzenet szövegébe beleírtuk, hogy a lebegőpontos adatot egy tizedesjegy pontossággal jelezze ki {*0:F1}.

Ha az üzenetdoboz bemenete IGAZ állapotúra vált, az üzenet tartalma bekerül a naplófájlba, a doboz kimenete pedig egyidejűleg IGAZ állapotot vesz fel. Ez visszatörli a naplózáskérés jelzőt.



6.14 Programvezérlő utasítások

6.14.1 Modul vége utasítás: END

A ciklikus PLC modul végét jelző utasítás. Úgy a Főprogramban, mint az Int0 modulban használható

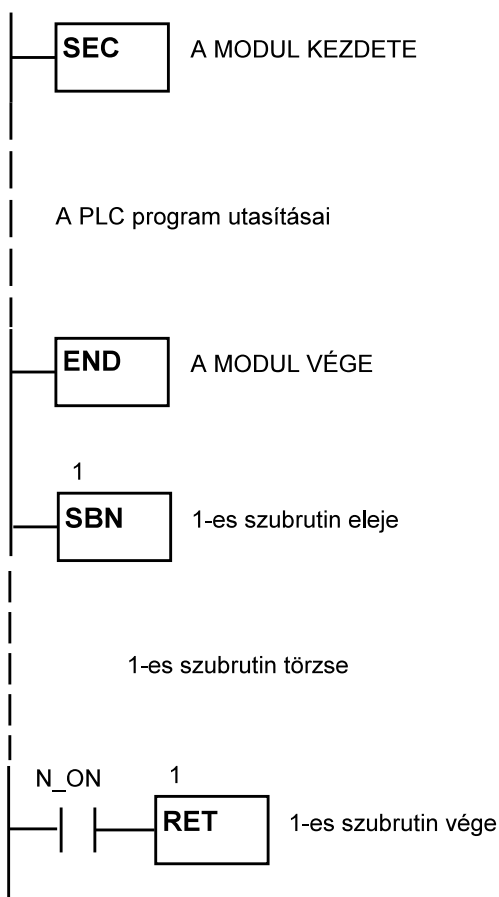
A PLC programmodul a létradiagram elejétől kezdve végrehajtja a programban lévő utasításokat majd, ha a program végrehajtása az END utasításra fut, a PLC programmodul futása befejeződik, és csak a ciklusidő (Főprogram esetén T_{PLC} , Int0 esetén TimeSlice) idő eltelte után indul újra. Az END utasításnak elválasztó szerepe van. A PLC programmodulban használt szubrutinok törzsét az END utasítás után kell leírni szintén létradiagram formátumban. Ha a modulban nem használunk szubrutinokat az utasítás használata nem kötelező.

Figyelem! Az END utasítás, bár záróelem, kizárólag az 1. oszlopba írható!

Az END utasítás bemenő paramétere:

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a logikai egység kommentje.



6.14.2 Ugrás a PLC programmodulban: JMP és JME utasítások

A JMP és JME utasítások segítségével átugorhatjuk a PLC programmodul egy adott szakaszát. A JMP ugró utasítás érvényesülése feltételhez kötött: ha a bemenetén teljesül a feltétel, akkor ugrik. Az ugrás a JMP utasításban megadott azonosítójú címkeire ugrik.

A JME utasítással jelölhetjük ki azt a címkét, ahova a JMP utasítás ugrik.

☞ **Figyelem!** A JMP és JME utasításokban használt címkék 0-tól 99-ig terjedhetnek. Ügyelni kell arra, hogy minden címke egyedi legyen és ne legyen átfedés a szubrutinok SBS, SBN és RET utasításainak címkéivel!

☞ **Figyelem!** A JMP utasítással csak modulon belül lehet ugrani, másik modulba nem!

A JMP utasítás bemenete

Van egy

engedélyező bemenete.

Az engedélyező bemenet igaz állapotában végzi el az ugrást.

A JMP utasítás bemenő paraméterei

– **Jump Number:**

Értékhatár: 0...99

A címke azonosítója, ahová az ugrás történik.

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a logikai egység kommentje.

A JME utasítás bemenete

☞ **Figyelem!** Az JME utasítás, bár záróelem, kizárólag az 1. oszlopba írható, nincs bemenete!

A JME utasítás bemenő paraméterei

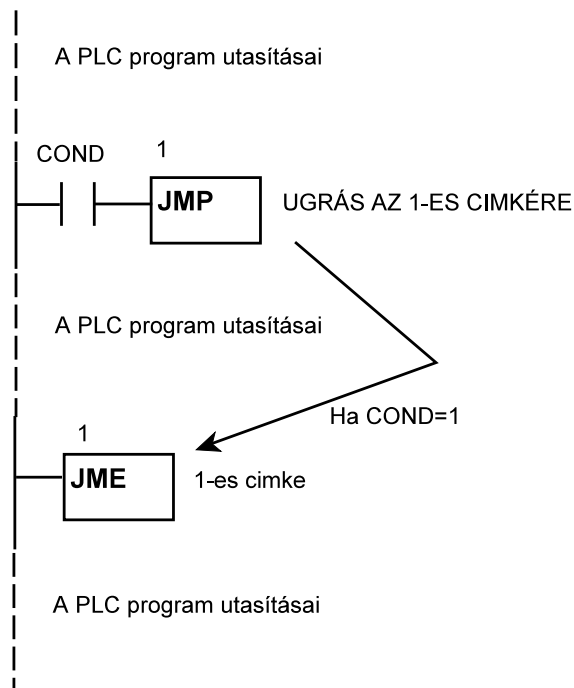
– **Jump Number:**

Értékhatár: 0...99

A címke azonosítója, a JMP utasítás ide ugrik.

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a logikai egység kommentje.



6.14.3 Szubrutin hívás utasításai: SBS, SBN és RET utasítások

A PLC program modulból SBS utasítással szubrutin hívást kezdeményezhetünk. Az utasítás bemenetén feltételt kell megadni, hogy mikor hívódjék meg a szubrutin. Az utasításban meg kell adni a szubrutin azonosítóját, azaz címkéjét, ahova a vezérlés átadódik.

A megfelelő címkével ellátott SBN utasítással kezdődik a szubrutin.

A szubrutin végét a RET utasítással adhatjuk meg. Az utasítás bemenetén feltételt kell megadni, hogy mikor térhet vissza.

A szubrutinok törzsét, tehát az SBN és RET utasítások által bezárt programrészt mindig a modul END utasítása után kell írni!

Többszörös, egymásba ágyazott szubrutinhívás lehetséges. A hívások mélységét csak a rendelkezésre álló címkék száma korlátozza.

☞ **Figyelem!** Az SBS, SBN és RET utasításokban egy modulon belül használt címkék 0-tól 99-ig terjedhetnek. Ügyelni kell arra, hogy minden címke egyedi legyen és ne legyen átfedés a JMP, JME utasítások címkéivel!

☞ **Figyelem!** Az egyik modulban leírt szubrutint nem lehet a másik modulból hívni!

Az SBS utasítás bemenete

Van egy

engedélyező bemenete.

Az engedélyező bemenet igaz állapotában végzi el a szubrutin hívást.

Az SBS utasítás bemenő paraméterei

– **Subroutine Number:**

Értékhatár: 0...99

A szubrutin azonosítója, címkéje, ami meghívásra kerül.

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a logikai egység kommentje.

Az SBN utasítás bemenete

☞ **Figyelem!** Az SBN utasítás, bár záróelem, kizárólag az 1. oszlopba írható, nincs bemenete!

Az SBN utasítás bemenő paraméterei

– **Subroutine Number:**

Értékhatár: 0...99

A címke azonosítója, az SBS utasítás ezt hívja.

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a logikai egység kommentje.

A RET utasítás bemenete

Van egy

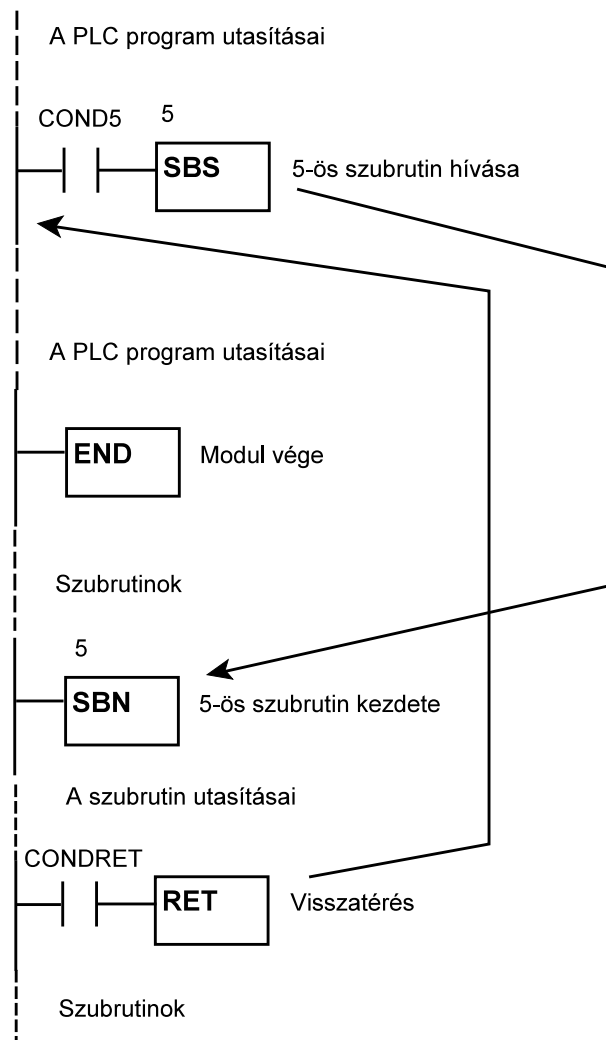
engedélyező bemenete.

Az engedélyező bemenet igaz állapotában tér vissza a szubrutinból.

A RET utasítás bemenő paraméterei

– **Remark:**

Megjegyzés: a Remarkba írt szöveg lesz a logikai egység kommentje.



6.15 A tengelymozgató utasítás: MOVCMMD

A tengelyek mozgatása alapvetően az NC feladata.

Szükség lehet azonban arra, hogy a **PLC** időnként **elkérjen** egy, vagy több **tengelyt az NC-től**, saját parancsaival mozgassa azokat, majd a művelet végeztével visszaadja a tengelyt, vagy tengelyeket az NC-nek. Ilyen alkalom lehet például, hogy szerszám-, vagy palettacsere közben bizonyos tengelyeket cserepozícióra kell küldeni. A tengelyek elkérése, visszaadása, indítása, stb. az erre a célra szolgáló jelzőkön keresztül történik.

☞ **Figyelem!** Az ilyen jellegű mozgásokat a PLC-ben szigorúan pufferürítő (összevárós) funkcióban kell kezelni, amely funkciókat paraméteren lehet kijelölni. Ennek az oka, hogy a funkció végeztével az NC-nek vissza kell olvasnia a PLC-ből elmozgatott tengelyek pozícióit és ebből az új pozícóból kiindulva kell a megmunkálást folytatnia.

Másik felhasználási területe a tengelymozgatásoknak az, amikor egy adott tengely mozgatása kizárólag a PLC hatáskörébe tartozik. Ilyen lehet például egy revolverfej, vagy magazin szervomotorral történő forgatása. Ilyenkor a PLC már a vezérlő bekapcsolásakor elkéri az NC-től a tengelyt és vissza se kell, hogy adja azt.

A tengelymozgató utasítás nem közvetlen az interpolátorba írja a parancsot, hanem a tengelymozgató utasítás **egymondatos pufferébe**. A pufferből azután veszi ki az interpolátor az utasítást, miután az előző parancsot végrehajtotta. A tengelymozgató utasítás pufferének folyamatos írással folyamatosan mozgathatjuk az interpolátort.

A tengelymozgató utasítás ki-, és bemenete

A tengelymozgató utasításnak van egy

engedélyező bemenete.

Az engedélyező bemenet igaz állapotában megvárja, hogy a mondatpuffer kiürüljön és az utasítást beírja a pufferbe.

A tengelymozgató utasításnak *konfigurálhatunk*

kimenetet.

A kimenet IGAZ állapotot vesz fel, miután sikerült a parancsot beírni a pufferbe. A kimenet HAMIS állapotot vesz fel, ha a bemenete HAMIS, vagy, ha az utasítás nem hajtható végre, azaz ha az utasítás beállítja az FL_ER hibajelzőt!

A kimenetre akár újabb utasítást is köthetünk.

A tengelymozgató utasítás bemenő paraméterei

– **Axis Address:**

A mozgatandó tengely indexe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges. Egész szám, értékhatár 0...31.

– **Command Code:**

A parancskód lehet szimbólikus, szám, vagy indexelt. Egész szám. A parancskód értelmezésére és lehetséges értékeire lásd az alábbi táblázatot.

– **Speed:**

A mozgásparancs sebessége. Lehet szimbólikus, szám, vagy indexelt. Az adatot a kimenő mértékrendszerben értelmezi az N0104 Unit of Measure paraméter IND bitje alapján: mm/min, fok/min, vagy inch/min-ben. Lebegőpontos szám.

– **Distance:**

A mozgás úthossza. Lehet szimbólikus, szám, vagy indexelt. Az adatot a kimenő mértékrendszerben értelmezi az N0104 Unit of Measure paraméter IND bitje alapján: mm, fok,

inch. Lebegőpontos szám.

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

– **Remark:**

Ha a kimenet nincs engedélyezve az ide írt megjegyzés kommentként kerül megjelenítésre, illetve, ha nem írunk ide semmit, az Axis Address mezőben megadott szimbólum kommentjét írja ide.

Az FL_ER hibajelzőt a következő esetekben állítja be:

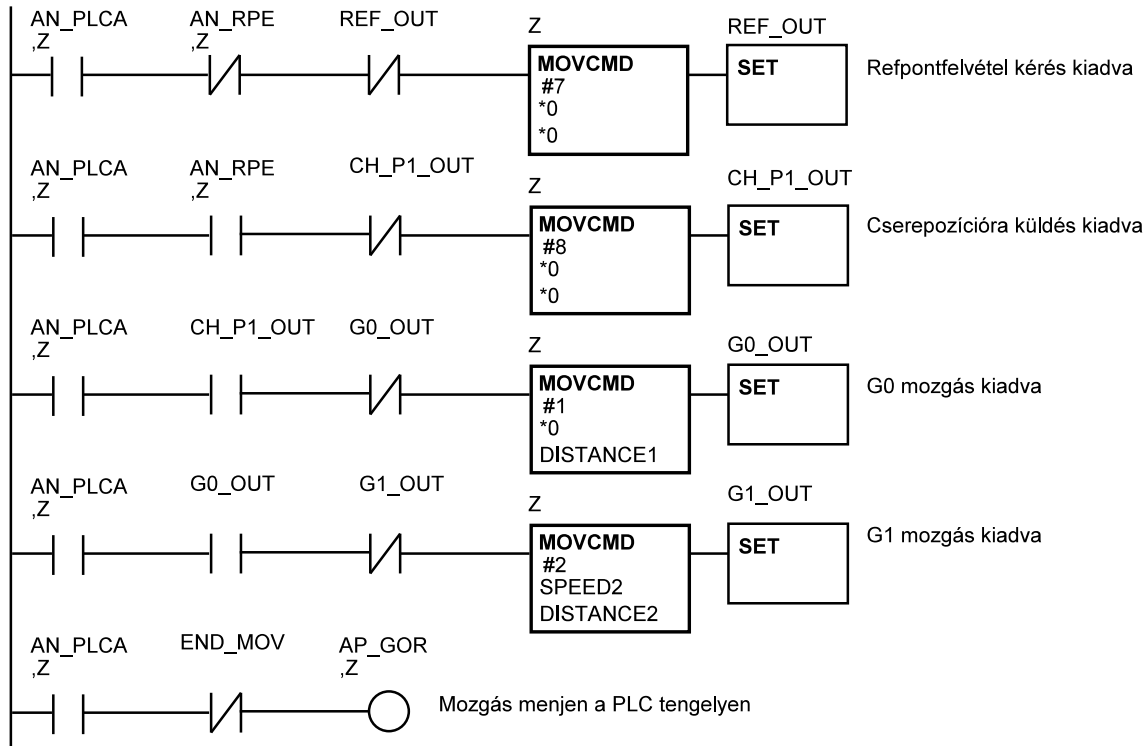
a mozgató tengely nincs PLC tengelynek átkapcsolva,
az utasítás kódja helytelen (helyes kódok az alábbi táblázatban),
azoknál a mozgásoknál, ahol a Speed paraméterre szükség van és a paraméter 0.

A **Command Code** értelmezése:

Com. Code	Ekvivalens NC kód	Leírás	Speed	Distance
0	G00 G90	Gyorsmeneti mozgás	nem számít	az aktuális koordináta-rendszerben megadott abszolút pozíció
1	G00 G91	Gyorsmeneti mozgás	nem számít	Inkrementális elmozdulás
2	G01 G90 G94	Egyenes interpoláció	percenkénti előtolás	az aktuális koordináta-rendszerben megadott abszolút pozíció
3	G01 G91 G94	Egyenes interpoláció	percenkénti előtolás	Inkrementális elmozdulás
4	G01 G90 G95	Egyenes interpoláció	fordulatonkénti előtolás	az aktuális koordináta-rendszerben megadott abszolút pozíció
5	G01 G91 G95	Egyenes interpoláció	fordulatonkénti előtolás	Inkrementális elmozdulás
6	G53	Pozicionálás a gépi koordináta-rendszerben	nem számít	a gépi koordinátarendszerben megadott abszolút pozíció
7	G28	Refpontfelvétel	nem számít	nem számít
8	G30 P2	Pozicionálás a 2. referenciapont pozícióra	nem számít	nem számít

Com. Code	Ekvivalens NC kód	Leírás	Speed	Distance
9	G30 P3	Pozicionálás a 3. referenciapont pozícióra	nem számít	nem számít
10	G30 P4	Pozicionálás a 4. referenciapont pozícióra	nem számít	nem számít
11	G4 G94	Várakozás	nem számít	Várakozási idő másodpercben
12	G4 G95	Várakozás	nem számít	Várakozási idő orsó fordulatban
13	Jog mozgató. A parancsot egy újabb parancs, vagy az AP_RES törli	AP_JOGP=1: pozitív irányba, AP_JOGN=1: negatív irányba mozog	percenkénti előtolás: ha értéke=0: a 0316 Jog F Contr paraméter alapján mozog, ha értéke >0 a megadott előtolással.	nem számít
14	Kézikerék mozgató. A parancsot egy újabb parancs, vagy az AP_RES törli	Ha a P_HnAS regiszterek valamelyike a PLC tengelyhez van kijelölve, a kézikerék szerint mozog	nem számít	nem számít

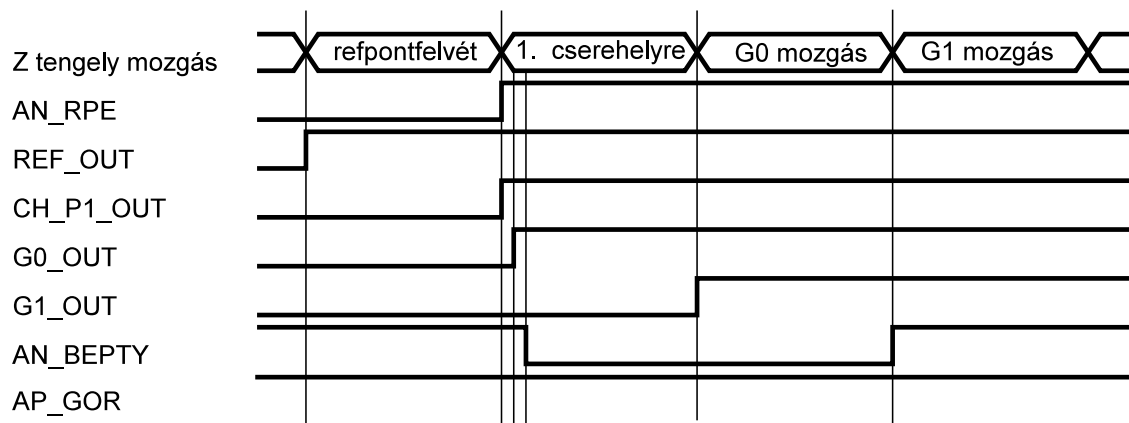
Az alábbi ábrán egy mintapélda látható PLC tengely mozgására. A MOVCMMD utasítások egymás után következnek. Az utasítássor először megvizsgálja, hogy a Z tengelyen van-e referenciapont. Ha nincs, refpontfelfétel parancsot ad ki, ha van, mindjárt a 2-es referenciapontra (1-es cserehelyre) küldi a tengelyt. Ezután gyorsmenettel történő, inkrementális, DISTANCE1 elmozdulású mozgás következik, majd egyenes interpoláció a DISTANCE2 abszolút pozícióra, SPEED2 sebességgel.



A repontfelvétel parancsot azonnal kiveszi az interpolátor, ezért az AN_BEPTY (puffer üres) jelző nem változik, azaz azt jelzi, hogy a puffer üres.

A következő parancs (1. cserehelyre való mozgás) a repont van (AN_RPE=1) jelző megjövedele után kerül kiadásra. Ezt azonnal kiszedi az interpolátor (AN_BEPTY még 1), majd a következő PLC ciklusban a G0 parancs beírásra kerül a pufferbe. Egy PLC ciklusnyi késleltetéssel az AN_BEPTY=0 állapota jelzi, hogy a puffer nem üres: az interpolátor az 1. cserehelyre mozog, és a G0 mozgásparancs a pufferben van. Amint cserehelyre futott a tengely, az interpolátor kiveszi a G0 parancsot a pufferből, de a G1 parancs azonnal beíródik, tehát az AN_BEPTY jelző marad 0. A G0 mozgás végeztével az interpolátor kiszedi a G1 parancsot a pufferből és, mivel több mozgásparancs nincs, az AN_BEPTY=1 állapotot vesz fel.

Ha folyamatos mozgásra van szükség, a PLC programnak nem kell figyelnie az AN_BEPTY jelzőt, mert a MOVCMDB parancs elvégzi azt.



6.16 A globális makróváltozók írása és olvasása

A #100...#499 és a #500...#599 globális makróváltozók értéke a PLC programból írható és kiolvasható.

Mivel ezek a makróváltozók csatornánként léteznek, az utasítás használatakor mindig meg kell adni annak a csatornának az indexét is, amelyikből olvasni, vagy amelyikbe írni akarunk.

6.16.1 Globális makróváltozó olvasása: a MACR utasítás

A MACR utasítással egy csatorna globális makróváltozói (#100...#499 és a #500...#599) kérdezhetők le. Ezek a változók mindig lebegőpontosak, tehát a PLC programban mindig 2 duplaszó helyet kell foglalni számukra.

A MACR utasítás ki-, és bemenete

A MACR utasításnak van egy

engedélyező bemenete.

Az engedélyező bemenet igaz állapotában végzi el a kiolvasást.

A MACR utasításnak *konfigurálhatunk*

kimenetet.

A kimenet IGAZ állapotot vesz fel, ha az utasítás bemenete IGAZ és az olvasás műveletet végrehajtotta. A kimenet HAMIS lesz, illetve az FL_ER jelző 1-be íródik, ha az utasítás nem hajtható végre.

A kimenetre újabb utasításokat is köthetünk.

A MACR utasítás bemenő paraméterei

– **Address of Result:**

Értékhatár: n...9998 (n: ahol a felhasználói címek kezdődnek)

A célregiszter címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

A lekérdezett érték erre a címre kerül. *Az itt megadott cím helyfoglalása mindig double kell legyen, mert az utasítás mindig lebegőpontos értéket ad vissza!*

– **Number of macro variable:**

Számszerű értékadásnál # előjelzött kell használni (decimális bevitel).

Értékhatár számszerűen: #100...#999.

Megadható szimbólíkusa is. Indexelt hivatkozás lehetséges.

– **Channel:**

Annak a csatornának az indexe számszerűen, vagy szimbolikusan, amelyik csatornából ki akarjuk olvasni a makróváltozót..

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

– **Remark:**

Ha a kimenet nincs engedélyezve az ide írt megjegyzés kommentként kerül megjelenítésre, illetve, ha nem írunk ide semmit, az Address of Result mezőben megadott szimbólum kommentjét írja ide.

Ha a megadott makróváltozó nem a #100...#999 tartományba esik, vagy a megadott csatorna indexe nem létező csatornára mutat, a kimenet HAMIS lesz, illetve az FL_ER jelző 1-be íródik.

6.16.2 Globális makróváltozó írása: a MACW utasítás

A MACW utasítással egy csatorna globális makróváltozóit (#100...#499 és a #500...#599) írhatjuk. Ezek a változók mindig lebegőpontosak, tehát a PLC memóriából mindig 2 duplaszót ír be a makróváltozóra.

A MACW utasítás ki-, és bemenete

A MACW utasításnak van egy

engedélyező bemenete.

Az engedélyező bemenet igaz állapotában végzi el az írást.

A MACW utasításnak *konfigurálhatunk*

kimenetet.

A kimenet IGAZ állapotot vesz fel, ha az utasítás bemenete IGAZ és az írás műveletet végrehajtotta. A kimenet HAMIS lesz, illetve az FL_ER jelző 1-be íródik, ha az utasítás nem hajtható végre.

A kimenetre újabb utasításokat is köthetünk.

A MACW utasítás bemenő paraméterei

– **Number of macro variable:**

Számszerű értékadásnál # előjelzöt kell használni (decimális bevitel).

Értékhatár számszerűen: #100...#999.

Megadható szimbólíkusa is. Indexelt hivatkozás lehetséges.

– **Operand:**

Értékhatár: n...9998 (n: ahol a felhasználói címek kezdődnek)

A forrás PLC regiszter címe számszerűen, vagy szimbólikusan. Indexelt megadás lehetséges.

Az itt megadott cím helyfoglalása mindig double kell legyen, mert az utasítás mindig lebegőpontos értéket ír!

– **Channel:**

Annak a csatornának az indexe számszerűen, vagy szimbólikusan, amelyik csatornából ki akarjuk olvasni a makróváltozót..

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

– **Remark:**

Ha a kimenet nincs engedélyezve az ide írt megjegyzés kommentként kerül megjelenítésre, illetve, ha nem írunk ide semmit, az Address of Result mezőben megadott szimbólum kommentjét írja ide.

Ha a megadott makróváltozó nem a #100...#999 tartományba esik, vagy a megadott csatorna indexe nem létező csatornára mutat, a kimenet HAMIS lesz, illetve az FL_ER jelző 1-be íródik.

6.17 Az NC belső változóinak lekérdezése: az SCP utasítás

Az SCP utasítással az NC azon belső regiszterei kérdezhetők le, amelyeket a vezérlőbe épített oszcilloszkóppal meg lehet jeleníteni. Ezek a változók különböző típusúak lehetnek: bitesek, duplaszavasak, vagy lebegőpontosak. A kapott eredmény azonban mindig lebegőpontos!

Az SCP utasítás ki-, és bemenete

Az SCP utasításnak van egy

engedélyező bemenete.

Az engedélyező bemenet igaz állapotában végzi el a kiolvasást.

Az SCP utasításnak *konfigurálhatunk*

kimenetet.

A kimenet IGAZ állapotot vesz fel, ha az utasítás bemenete IGAZ és az olvasás műveletet végrehajtotta.

A kimenetre újabb utasításokat is köthetünk.

Az SCP utasítás bemenő paramétere

– **Address of Result:**

Értékhatár: n...9998 (n: ahol a felhasználói címek kezdődnek)

A célregiszter címe számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges.

A lekérdezett érték erre a címre kerül. *Az itt megadott cím helyfoglalása mindig double kell legyen, mert az utasítás mindig lebegőpontos értéket ad vissza!*

– **Scope Symbol:**

Legördülő menüből választható ki a lekérdezendő adat szimbóluma. A szimbólumok jelentését a lenti táblázat adja meg.

– **Id:**

A legördülő menüből kiválasztott szimbólum azonosítószáma kerül ide. Beírni nem kell.

– **Param1:**

A kiválasztott szimbólumhoz tartozó 1. paraméter értéke: #<szám>. Megadását lásd a lenti táblázatban.

– **Param2:**

A kiválasztott szimbólumhoz tartozó 2. paraméter értéke: #<szám>. Megadását lásd a lenti táblázatban.

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

– **Remark:**

Ha a kimenet nincs engedélyezve az ide írt megjegyzés kommentként kerül megjelenítésre, illetve, ha nem írunk ide semmit, az Address of Result mezőben megadott szimbólum kommentjét írja ide.

Az alábbi szimbólumokkal megadott NC változók kérdezhetők le az SCP utasítás használatával:

Szimbólum	Leírás	Param1	Param2
PlcBit	PLC bites változó lekérdezése a PLC memóriából. Lebegőpontos adat.	Dupla szó címe: számszerűen, előtte # karakter Érték: #0...#9999	Bit címe: számszerűen, előtte # karakter Érték: #0...#31
PlcInt	PLC egész változó lekérdezése a PLC memóriából. Lebegőpontos adat.	Dupla szó címe: számszerűen, előtte # karakter Érték: #0...#9999	#0
PlcDouble	PLC lebegőpontos változó lekérdezése a PLC memóriából. Lebegőpontos adat.	Lebegőpontos változó címe: számszerűen, előtte # karakter Érték: #0...#9998	#0
DirectPlcBit	PLC bites változó lekérdezése a Time Slice memóriából. Lebegőpontos adat.	Dupla szó címe: számszerűen, előtte # karakter Érték: #0... #<PLCNVRAM-1>	Bit címe: számszerűen, előtte # karakter Érték: #0...#31
DirectPlcInt	PLC egész változó lekérdezése a Time Slice memóriából. Lebegőpontos adat.	Dupla szó címe: számszerűen, előtte # karakter Érték: #0... #<PLCNVRAM-1>	#0
DirectPlcDouble	PLC lebegőpontos változó lekérdezése a Time Slice memóriából. Lebegőpontos adat.	Lebegőpontos változó címe: számszerűen, előtte # karakter Érték: #0... #<PLCNVRAM-2>	#0
ComPosAx	Egy tengely mérőrendszerének kiadott parancspozíció. Kimeneti mértékrendszerben (mm, fok, inch) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0

Szimbólum	Leírás	Param1	Param2
ComVelAx	Egy tengely mérőrendszerének kiadott sebességparancs. Kimeneti mértékrendszerben (mm/sec, fok/sec, inch/sec) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0
FolErrAx	Egy tengelyen a követési hiba (lemaradás) mértéke. Kimeneti mértékrendszerben (mm, fok, inch) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0
CommandAx	Egy tengelynek a hajtás felé kimenő parancsjele. Kimeneti mértékrendszerben (mm/sec, fok/sec, inch/sec) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0
ActPosAx	Egy tengelynek a jeladóról mért aktuális pozíciója. Kimeneti mértékrendszerben (mm, fok, inch) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0
PosErrAx	Egy tengely parancspozíciójából jósolt aktuális pozíció és a jeladóról mért pozíció különbsége. Kimeneti mértékrendszerben (mm, fok, inch) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0
TachAx	Egy tengely jeladóról mért sebessége. Kimeneti mértékrendszerben (mm/sec, fok/sec, inch/sec) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0

Szimbólum	Leírás	Param1	Param2
TachRealAx	Egy tengely jeladóról mért sebessége. Time Slice idő alatt a jeladóról bejött impulzusok száma lebegőpontos adatként.	A tengely száma. Érték: #1...#32	#0
EGBvTarAx	EGB mester tengely sebessége. Kimeneti mértékrendszerben (mm/sec, fok/sec, inch/sec) megadott lebegőpontos adat.	A szolga tengely számát kell megadni! Érték: #1...#32	#0
EGBcCurrAx	EGB szolga tengely sebessége. Kimeneti mértékrendszerben (mm/sec, fok/sec, inch/sec) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0
SyncErrAx	A szolga tengely pozícióeltérése a mestertől, EGB, vagy gantry szinkron esetén. Kimeneti mértékrendszerben (mm, fok, inch) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0
PitchAx	Az adott tengelyen pillanatnyilag érvényes menetemelkedés kompenzáció értéke. Kimeneti mértékrendszerben (mm, fok, inch) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0
StraightnessAx	Az adott tengelyen pillanatnyilag érvényes egyenességi kompenzáció értéke. Kimeneti mértékrendszerben (mm, fok, inch) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0

Szimbólum	Leírás	Param1	Param2
CompenValAx	Az adott tengelyen pillanatnyilag érvényben lévő összes kompenzáció összege. Kimeneti mértékrendszerben (mm, fok, inch) megadott lebegőpontos adat.	A tengely száma. Érték: #1...#32	#0
ComPosSp	Zárt pozícióhurok esetén egy orsó mérőrendszerének kiadott parancspozíció. Orsó fordulatban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
ComVelSp	Zárt pozícióhurok esetén egy orsó mérőrendszerének kiadott sebességparancs. Orsó fordulat/sec-ban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
FolErrSp	Zárt pozícióhurok esetén egy orsón a követési hiba (lemaradás) mértéke. Orsó fordulatban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
CommandSp	Az orsóhajtás felé menő parancsjel. Orsó fordulat/sec-ban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
ActPosSp	Az orsó aktuális pozíciója. Orsó fordulatban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
PosErrSp	Zárt pozícióhurok esetén egy orsó parancspozíciójából jósolt aktuális pozíció és a jeladóról mért pozíció különbsége. Orsó fordulatban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
TachSp	Egy orsó jeladóról mért sebessége. Orsó fordulat/sec-ban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0

Szimbólum	Leírás	Param1	Param2
NActSp	Egy orsó jeladóról mért sebessége. Orsó fordulát/min-ban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
NSetSp	Az orsókezelő bemenetén az override-dal módosított fordulatszám parancs. Orsó fordulát/min-ban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
NCommandSp	Az orsóhajtás felé menő, az orsókezelő által módosított parancsjel. Orsó fordulát/min-ban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
SyncVTargetSp	Orsószingkron állapotban a mesterorsó sebessége. Orsó fordulát/sec-ban megadott lebegőpontos adat.	A szolga orsó számát kell megadni! Érték: #1...#16	#0
SyncVSlaveSp	Orsószingkron állapotban a szolga orsó sebessége. Orsó fordulát/sec-ban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
SyncErrSp	Orsószingkron állapotban a szolga orsó pozícióeltérése a mesterorsótól. Orsó fordulátban megadott lebegőpontos adat.	Az orsó száma. Érték: #1...#16	#0
Measured	Belső használatra.		
RealTime	Az összes valós idejű feladat időhasználat μ sec-ben. Lebegőpontos adat.	#0	#0
HardwareTime	Az I/O-k frissítési ideje μ sec-ben. Lebegőpontos adat.	#0	#0
ChannelsTime	A csatornafekezelő időhasználat μ sec-ben. Lebegőpontos adat.	#0	#0

Szimbólum	Leírás	Param1	Param2
AxesTime	A tengelykezelő időhaszná- lata μ sec-ben. Lebegőpon- tos adat.	#0	#0
PlcTime	Az Int0 PLC modul idő- használata μ sec-ben. Lebe- gőpontos adat.	#0	#0
PlcCycle	A PLC Main időhasználata időhasználata μ sec-ben. Le- begőpontos adat. A real time megszakításokat tartalmazza.	#0	#0
CANErr	Belső használatra.		
TSliceErr	Belső használatra.		
CNCBufferCount	A pufferben lévő mondatok száma. Lebegőpontos adat.	A csatorna száma. Érték: #1...#8	#0
RotaryAx	Forgó tengelyek esetén a ciklikus pozíció értéke. Lebegőpontos pozícióadat fokban megadva.	A tengely száma. Érték: #1...#32	#0
TachRealSp	Egy orsó jeladóról mért se- bessége. Time Slice idő alatt a jela- dóról bejött impulzusok száma lebegőpontos adat- ként.	Az orsó száma. Érték: #1...#16	#0
NCTDriveMess	Az NCT hajtás által feladott adat, lebegőpontosan.	A mellék-, vagy az orsóhajtás száma: #1...#48	#0: motor fordú- latszám [ford/min]
			#1: motor áram [A],
			#2: motor relatív áram I/I_n [%]
			#3: sínfeszültség [V]
			#4: motor hőmérséklet [$^{\circ}$ C]

Szimbólum	Leírás	Param1	Param2
			#5: motor teljesítmény [kW]
Velocity	A csatornában érvényes pillanatnyi előtolás értéke, amivel az interpolátor mozog. Kimeneti mértékrendszerben (mm/sec, fok/sec, inch/sec) megadott lebegőpontos adat.	A csatorna száma. Érték: #1...#8	#0

6.18 Az NC belső változóinak tömbös olvasása és írása

A tömbös olvasó utasítással az NC memória egy tömbjét olvashatjuk be a PLC memóriaterületre. A tömbös író utasítással a PLC memória egy tömbjét írhatjuk be NC memória egy adott területére.

6.18.1 NC memóriatömb olvasása: az MR utasítás

Az utasítással az NC-ben tárolt különböző adatokat olvashatunk ki.

Az MR utasítás ki-, és bemenete

Az MR utasításnak van egy

engedélyező bemenete.

Az engedélyező bemenet igaz állapotában végzi el a kiolvasást.

Az MR utasításnak *konfigurálhatunk*

kimenetet.

A kimenet IGAZ állapotot vesz fel, ha az utasítás bemenete IGAZ és az olvasás műveletet végrehajtotta.

A kimenetre újabb utasításokat is köthetünk.

☞ **Figyelem!** Az utasítás végrehajtása több PLC ciklust is igénybe vehet, ezért az engedélyező bemenetét addig igaz állapotban kell tartani amíg az utasítás kimenetén meg nem jelenik az igaz állapot. Ezután az engedélyező bemenetét ki kell kapcsolni.

Az MR utasítás bemenő paramétere

– **Data Address:**

Értékhatar: 0...9999

Az utasítás számára kijelölt PLC memóriatömb kezdőcíme számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges. Ebbe a tömbbe írónak az utasítás végrehajtásának üzenetei és ide olvassa be az NC-ből az adatokat.

– **Function Code:**

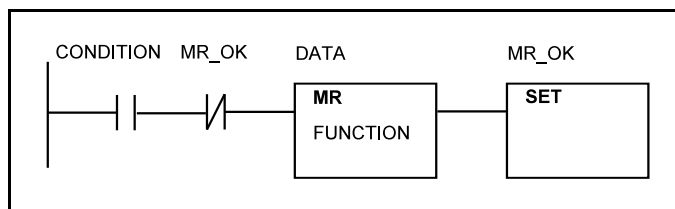
Azt határozza meg, hogy az olvasás az NC memória mely területéről történjék. A különböző funkciókódok jelentését a következő fejezetek tárgyalják.

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

– **Remark:**

Megjegyzés.



A PLC memóriában a beolvasandó adat(ok) hosszának függvényében tárterületet kell foglalni. Az utasításban Data Address változón megjelölt és utána következő címre az utasítás végrehajtási adatokat ír.

- **Cím:** a Data Address paraméteren megadott címtől kezdődik. Mindig erre a címre íródik be a teljesítés kódja.
- **Teljesítés kódja:** kimenő adat, az utasítás végrehajtása során képződik.
- **Adathossz:** bemenő adat. Az utasítás számára a *funkcióspecifikus adatterület hosszát* mondja meg duplaszóban (DWORD).
- **Funkcióspecifikus terület:** Ettől a címtől kezdődnek a Funkció kód által meghatározott speciális adatok. Ezek az adatok lehetnek:
 - az olvasást tovább konfiguráló bemenő adatok, és
 - az olvasás kimenő adatai.

Cím	Adat
0	Teljesítés kódja
1	Adathossz (bemenő adat)
2	Funkcióspecifikus terület
3	
...	...
n	...

☞ **Figyelem!** Az adathossz mindig a konfigurációs bemenő adatok és a beolvasni kívánt adatok együttes száma DWORD egységben!

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz
..	Funkcióspecifikus hibák
n	

6.18.2 NC memóriatömb írása: az MW utasítás

Az utasítással az NC-ben tárolt különböző adatokat írhatunk át.

Az MW utasítás ki-, és bemenete

Az MW utasításnak van egy

engedélyező bemenete.

Az engedélyező bemenet igaz állapotában végzi el az írást.

Az MW utasításnak *konfigurálhatunk*

kimenetet.

A kimenet IGAZ állapotot vesz fel, ha az utasítás bemenete IGAZ és az írás műveletet végrehajtotta.

A kimenetre újabb utasításokat is köthetünk.

☞ **Figyelem!** Az utasítás végrehajtása több PLC ciklust is igénybe vehet, ezért az engedélyező bemenetét addig igaz állapotban kell tartani amíg az utasítás kimenetén meg nem jelenik az igaz állapot. Ezután az engedélyező bemenetét ki kell kapcsolni.

Az MW utasítás bemenő paramétere– **Data Address:**

Értékhatár: 0...9999

Az utasítás számára kijelölt PLC memóriatömb kezdőcíme számszerűen, vagy szimbolikusan. Indexelt megadás lehetséges. Ebbe a tömbbe íródnak az utasítás végrehajtásának üzenetei és innen írja be az NC-be az adatokat.

– **Function Code:**

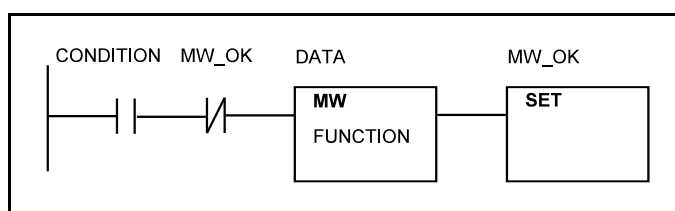
Azt határozza meg, hogy az írás az NC memória mely területére történjék. A különböző funkciókódok jelentését a következő fejezetek tárgyalják.

– **Output:**

Engedélyezett állapotban az utasításdoboz kimenetére további utasításokat lehet kötni.

– **Remark:**

Megjegyzés.



A PLC memóriában a kiírandó adat(ok) hosszának függvényében tárterületet kell foglalni. Az utasításban Data Address változón megjelölt címre az utasítás végrehajtási adatokat ír.

– **Cím:** a Data Address paraméteren megadott címtől kezdődik. Mindig erre a címre íródik be a teljesítés kódja.

– **Teljesítés kódja:** kimenő adat, az utasítás végrehajtása során képződik.

– **Adathossz:** bemenő adat. Az utasítás számára a *funkcióspecifikus adatterület hosszát* mondja meg duplaszóban (DWORD).

– **Funkcióspecifikus terület:** Ettől a címtől kezdődnek a Funkció kód által meghatározott speciális adatok. Ezek az adatok lehetnek:

- az írást tovább konfiguráló bemenő adatok, és
- a kiírandó adatok.

Cím	Adat
0	Teljesítés kódja
1	Adathossz (bemenő adat)
2	Funkcióspecifikus terület
3	
...	...
n	...

☞ **Figyelem!** Az adathossz mindig a konfigurációs bemenő adatok és a kiírni kívánt adatok együttes száma DWORD egységben!

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz
3	Írásvédett
...	Funkcióspecifikus hibák
n	

6.19 Adatok mozgatása a nem felejtő tár és a PLC memória között

Szükség lehet a PLC változók egy tömbjének nem felejtő memóriában való eltárolására, illetve onnan történő kiolvasására.

Egyes vezérlő típusok a PLCNVRAM címtől kezdődő 1024 duplaszó hosszúságú adattömböt automatikusan elmentik, a mentéssel a PLC programozónak nem kell törődnie.

Más vezérlő típusok *nem támogatják* a PLCNVRAM címtől induló 1024 duplaszó hosszúságú memóriatömb *automatikus megőrzését kikapcsolás után*, hanem a PLC programból megfelelő tömbös írás segítségével azonnal el kell a szükséges adatokat menteni, valahányszor megváltoztattuk azokat. Ugyanúgy, megfelelő alkalommal, pl. bekapcsolás után az elmentett memóriatömböt be kell olvasni. A menteni kívánt PLC változók számára a vezérlő ebben az esetben is 1024 duplaszó hosszúságú tárhelyet tart fenn.

Az első változó azonosítója a nem felejtő tárban: 0,
az utolsóé pedig 1023.

A PLC változók mentését és visszaolvasását végző MW, MR utasítások végrehajtása több PLC ciklusnyi időt vesznek igénybe, ezért mindig meg kell várni, hogy a kimenetük IGAZ állapotba kerüljön.

6.19.1 PLC változók adatainak kiolvasása nem felejtő tárból

Utasítás: MR

Funkció kód: 10

Kezdő azonosítószámával és az változók darabszámával megadott tömb változóinak beolvasása a nem felejtő tárból a PLC változók közé.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 3
2	Beolvasandó első változó azonosítója a nem felejtő tárban: 0...1023
3	Beolvasásra kerülő első változó kezdőcíme a PLC memóriában
4	A beolvasandó változók száma: 1...1024

*: az adatot nem kell kitölteni

Az adathosszra 3-at kell írni. Több változó beolvasása esetén folyamatosan tölti fel a kezdőcímtől a PLC memóriát.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat

Cím	Adat
1	Adathossz: 3
2	Beolvasandó első változó azonosítója a nem felejtő tárban 0...1023
3	Beolvasásra kerülő első változó kezdőcíme a PLC memóriában
4	A beolvasandó változók száma: 1...1024

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 3
4	Tár kontrol szumma hibás
10	A változó azonosítója nem 0...1023 közé esik
11	Első változó kezdőcíme nem nagyobb, v. egyenlő, mint PLCNVRAM
12	Adaszám hiba: (beolvasandó első változó azonosítója)+ (beolvasandó változók száma)>1024

6.19.2 PLC változók adatainak kiírása nem felejtő tárba

Utasítás: MW

Funkció kód: 11

Az első változó kezdőcímeivel és a változók darabszámával megadott tömb változóinak kiírása a PLC memóriából a nem felejtő tárba. Az utasítás kimenete csak azután tér vissza 0 teljesítési kóddal, miután a kiírás után a tár kontrol szummája is kiszámításra került.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 3
2	Kiírandó első változó azonosítója a nem felejtő tárban: 0...#1023
3	Kiírásra kerülő első változó kezdőcíme a PLC memóriában

Cím	Adat
4	A kiírandó változók száma: 1...1024

*: az adatot nem kell kitölteni

Az adathosszra 3-at kell írni. Több változó kiírása esetén folyamatosan tölti fel az első változó azonosítójától kezdve a nem felejtő tárat.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 3
2	Kiírandó első változó azonosítója a nem felejtő tárban: 0...#1023
3	Kiírásra kerülő első változó kezdőcíme a PLC memóriában
4	A kiírandó változók száma: 1...1024

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 3
10	A változó azonosítója nem 0...1023 közé esik
11	Első változó kezdőcíme nem nagyobb, v. egyenlő, mint PLCNVRAM
12	Adaszám hiba: (kiírandó első változó azonosítója)+ (kiírandó változók száma)>1024

6.20 Paraméterek olvasása és írása PLC programból

A PLC program az MR és MW utasítások segítségével elérheti olvasásra, vagy írásra az NC paramétereit.

Az olvasás és írás során az alábbi szempontokat kell figyelembe venni:

- a paraméter adatábrázolását,
- a paraméter globális-e, vagy indexált-e,

Adatábrázolás szempontjából megkülönböztetünk:

- bites,
- **DWORD: fixpontos, duplaszavas,**
- **double: lebegőpontos** ábrázolású

paramétereket. A bites paraméterek PLC-ből történő helyfoglalása DWORD. A különböző bit értékek a DWORD 0...7 bitjén kerülnek ábrázolásra.

A paraméter indexálható:

- **gépcsoportonként,**
- **csatornánként,**
- **tenegelyenként,** vagy
- **orsónként.**

Az indexált paramétereknek nincs külön azonosítószámuk.

Egy paraméterre a paraméter azonosítószámával és indexével hivatkozhatunk. Globális paraméterre történő hivatkozás esetén az index értéke 0. Tömbös paraméter írás, vagy olvasás (több paraméter egyszerre történő beolvasása, vagy írása) csak nem globális paraméterekre lehetséges. Például orsónként különböző paraméter beolvasása az összes orsóra egyszerre megtörténhet.

Az egyes paraméterek azonosítószámát (Nnnnn) formátumát (bit, DWORD, double), illetve, hogy mi alapján indexálódik, a paraméter leírás tartalmazza.

6.20.1 DWORD típusú paraméterek olvasása

Utasítás: MR

Funkció kód: 31

Azonosítószámával és kezdőindexével megadott DWORD formátumú paraméterek adatainak beolvasása a PLC változók közé. A beolvasásra kerülő adatok számát az utasítás adathossz paramétere adja meg.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: min. 3, max. n+2
2	Beolvasandó paraméter azonosítója:<szám>
3	A paraméter kezdőindexe: =0: globális paraméter >0 beolvasandó 1. paraméter indexe
4	1. paraméter értéke *

Cím	Adat
5	2. paraméter értéke *
...	...
n+3	n. paraméter értéke *

*: az adatot nem kell kitölteni

Az adathosszra minimum 3-at kell írni, ekkor csak 1 duplaszót olvas be. Több adat beolvasása esetén folyamatosan tölti fel a PLC memóriát.

Ha pl. egy tengelyenként indexált paramétert kívánunk beolvasni, és a 2. és 3. tengely paraméterére vagyunk kíváncsiak, az adathossz=4 és a kezdőindex=2.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: min. 3, max. n+2
2	Beolvasandó paraméter azonosítója:<szám>
3	A paraméter kezdőindexe: =0: globális paraméter >0 beolvasandó 1. paraméter indexe
4	1. paraméter értéke
5	2. paraméter értéke
...	...
n+3	n. paraméter értéke

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: < 3, vagy > (paramétertömb utolsó változójának indexe - megadott kezdőindex száma) + 2
4	Tár kontrol szumma hibás

Kód	Jelentés
30	Nem létező paraméterre történő hivatkozás
31	A paraméter nem globális (paraméter indexe = 0)
32	A paraméter globális (paraméter indexe > 0)
33	Rossz olvasási kód: a paraméter formátuma nem DWORD

6.20.2 Double típusú paraméterek olvasása

Utasítás: MR

Funkció kód: 32

Azonosítószámával és kezdőindexével megadott double formátumú paraméterek adatainak beolvasása a PLC változók közé. A beolvasásra kerülő adatok számát az utasítás adathossz paramétere adja meg.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: min. 4, max. $2n+2$
2	Beolvasandó paraméter azonosítója:<szám>
3	A paraméter kezdőindexe: =0: globális paraméter >0 beolvasandó 1. paraméter indexe
4	1. paraméter értéke *
5	
6	2. paraméter értéke *
7	
...	...
$2n+2$	n. paraméter értéke *
$2n+3$	

*: az adatot nem kell kitölteni

Az adathosszra minimum 4-et kell írni, ekkor csak 1 lebegőpontos adatot olvas be. Több adat beolvasása esetén folyamatosan tölti fel a PLC memóriát.

Ha pl. egy tengelyenként indexált paramétert kívánunk beolvasni, és a 2. és 3. tengely paraméterére vagyunk kíváncsiak, az adathossz=6 és a kezdőindex=2.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: min. 4, max. $2n+2$
2	Beolvasandó paraméter azonosítója:<szám>
3	A paraméter kezdőindexe: =0: globális paraméter >0 beolvasandó 1. paraméter indexe
4	1. paraméter értéke
5	
6	2. paraméter értéke
7	
...	...
$2n+2$	n. paraméter értéke
$2n+3$	

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: < 3, vagy > (paramétertömb utolsó változójának indexe - megadott kezdőindex száma) $\times 2 + 2$
4	Tár kontrol szumma hibás
30	Nem létező paraméterre történő hivatkozás
31	A paraméter nem globális (paraméter indexe = 0)
32	A paraméter globális (paraméter indexe > 0)
33	Rossz olvasási kód: a paraméter formátuma nem double

6.20.3 DWORD típusú paraméter írása

Utasítás: MW

Funkció kód: 34

Azonosítószámával és kezdőindexével megadott DWORD formátumú paraméterek adatainak írása a PLC-ből. A kiírásra kerülő adatok számát az utasítás adathossz paramétere adja meg.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: min. 3, max. n+2
2	Írandó paraméter azonosítója:<szám>
3	A paraméter kezdőindexe: =0: globális paraméter >0 írandó 1. paraméter indexe
4	1. paraméter értéke
5	2. paraméter értéke
...	...
n+3	n. paraméter értéke

*: az adatot nem kell kitölteni

Az adathosszra minimum 3-at kell írni, ekkor csak 1 duplaszót ír. Több adat írása esetén folyamatosan ír ki a PLC memóriából.

Ha pl. egy tengelyenként indexált paramétert kívánunk írni, és a 2. és 3. tengelyét, az adathossz=4 és a kezdőindex=2.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: min. 3, max. n+2
2	Írandó paraméter azonosítója:<szám>
3	A paraméter kezdőindexe: =0: globális paraméter >0 írandó 1. paraméter indexe
4	1. paraméter értéke
5	2. paraméter értéke

Cím	Adat
...	...
n+3	n. paraméter értéke

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: < 3, vagy > (paramétertömb utolsó változójának indexe - megadott kezdőindex száma) + 2
30	Nem létező paraméterre történő hivatkozás
31	A paraméter nem globális (paraméter indexe = 0)
32	A paraméter globális (paraméter indexe > 0)
33	Rossz írási kód: a paraméter formátuma nem DWORD

6.20.4 Double típusú paraméterek írása**Utasítás: MR****Funkció kód: 35**

Azonosítószámával és kezdőindexével megadott double formátumú paraméterek adatainak írása a PLC-ből. A kiírásra kerülő adatok számát az utasítás adathossz paramétere adja meg.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: min. 4, max. $2n+2$
2	Beolvasandó paraméter azonosítója:<szám>
3	A paraméter kezdőindexe: =0: globális paraméter >0 beolvasandó 1. paraméter indexe
4	1. paraméter értéke
5	
6	2. paraméter értéke
7	
...	...
$2n+2$	n. paraméter értéke
$2n+3$	

*: az adatot nem kell kitölteni

Az adathosszra minimum 4-et kell írni, ekkor csak 1 lebegőpontos adatot ír. Több adat írása esetén folyamatosan ír ki a PLC memóriából.

Ha pl. egy tengelyenként indexált paramétert kívánunk beolvasni, és a 2. és 3. tengely paraméterére vagyunk kíváncsiak, az adathossz=6 és a kezdőindex=2.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: min. 4, max. $2n+2$
2	Beolvasandó paraméter azonosítója:<szám>
3	A paraméter kezdőindexe: =0: globális paraméter >0 beolvasandó 1. paraméter indexe
4	1. paraméter értéke
5	
6	2. paraméter értéke
7	
...	...
$2n+2$	n. paraméter értéke
$2n+3$	

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: < 3, vagy > (paramétertömb utolsó változójának indexe - megadott kezdőindex száma) $\times 2 + 2$
4	Tár kontrol szumma hibás
30	Nem létező paraméterre történő hivatkozás
31	A paraméter nem globális (paraméter indexe = 0)
32	A paraméter globális (paraméter indexe > 0)
33	Rossz írási kód: a paraméter formátuma nem double

6.21 Program kijelölése végrehajtásra

A PLC programból tetszőleges, a vezérlő memóriájában lévő alkatrészprogramot ki lehet jelölni végrehajtásra.

6.21.1 Programszámával megadott program kijelölése automata végrehajtásra

Utasítás: MW

Funkció kód: 40

Négy-, vagy nyolcjegyű programszámával (Onnnn, vagy Onnnnnnnnn) megadott alkatrészprogramot jelöl ki automata üzemben történő végrehajtásra, a megadott csatornában. Az írási művelet csak akkor hajtodik végre, azaz az utasítás kimenete csak akkor lesz igaz, amikor az automata üzemben nem fut program.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 2
2	Programszám: <szám>
3	Csatorna száma: 1...8

*: az adatot nem kell kitölteni

Az adathosszra mindig 2-öt kell írni.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 2
2	Programszám: <szám>
3	Csatorna száma: 1...8

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 1
40	Nem létező program: a megadott program nincs a tárban

6.22 Hálózati kommunikációs utasítások

A PLC program adatokat fogadhat, illetve küldhet ki Ethernet hálózaton keresztül. A kommunikáció UDP protokollon keresztül történik. Egy Ethernet UDP/IP keret felépítése:

Ethernet fejléc UDP/IP	Küldött, vagy fogadott adatok			CRC
	...	...	...	

A PLC a küldött, vagy fogadott adatokkal dolgozik.

6.22.1 Hálózati kapcsolat megnyitása

Utasítás: MW

Funkció kód: 60

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 5
2	Kapcsolat azonosító: mindig 1
3	Windows hibakód *
4	Protokoll: 1: UDP 2: TCP Szerver 3: TCP kliens
5	Cél eszköz IP címe. Pl: 192.168.1.12 = #C0A8010C Értéke nem lehet: 0.0.0.0 és 255.255.255.255
6	A port száma, amin kommunikálni akarunk. Javasolt értékek: 49202...49205

*: az adatot nem kell kitölteni

Az adathosszra mindig 5-öt kell írni.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 5
2	Kapcsolat azonosító: mindig 1
3	Windows hibakód: ha a teljesítés kódja 61, ide kerül a Windows által küldött hiba tovább dekódolva (lásd Windows Sockets Error Codes) *

Cím	Adat
4	Protokoll: 1: UDP 2: TCP Szerver 3: TCP kliens
5	Cél eszköz IP címe. Pl: 192.168.1.12 = #C0A8010C Értéke nem lehet: 0.0.0.0 és 255.255.255.255
6	A port száma, amin kommunikálni akarunk. Javasolt értékek: 49202...49205

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz
60	Érvénytelen kapcsolat azonosító
61	Kapcsolat létrehozása sikertelen
66	Hibás bejövő paraméterek

6.22.2 Hálózati kapcsolat bezárása

Utasítás: MW

Funkció kód: 61

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 1
2	Kapcsolat azonosító: mindig 1

*: az adatot nem kell kitölteni

Az adathosszra mindig 2-öt kell írni.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 1

Cím	Adat
2	Kapcsolat azonosító: mindig 1

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz
60	Érvénytelen kapcsolat azonosító

6.22.3 Hálózati adatcsomag fogadása

Utasítás: MR

Funkció kód: 62

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 3-5
2	Kapcsolat azonosító: mindig 1
3	Adattömb kezdőcíme a PLC memóriában
4	Adattömb hossza: ennyi PLC regiszternyi adat lesz fogadva $0 < \text{érték} \leq 350$
5	Widows hibakód **
6	Forrás eszköz IP címe. Pl: 192.168.1.12 = #\$C0A8010C **
7	Forrás eszköz port száma **

*: az adatot nem kell kitölteni

**: opcionális adat, az adatot nem kell kitölteni

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 3-5
2	Kapcsolat azonosító: mindig 1

Cím	Adat
3	Adattömb kezdőcíme a PLC memóriában
4	Adattömb hossza: ennyi PLC regiszternyi adat lesz fogadva $0 < \text{érték} \leq 350$
5	Widows hibakód: ha a teljesítés kódja 62, ide kerül a Windows által küldött hiba tovább dekódolva (lásd Windows Sockets Error Codes)
6	Forrás eszköz IP címe. Pl: 192.168.1.12 = #C0A8010C

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz
60	Érvénytelen kapcsolat azonosító
62	Az adatfogadás során hiba történt
63	A megadott kapcsolat nincs nyitva
64	Az adattömb hossza hibás: =0, vagy >350
65	Nem érkezett adat

6.22.4 Hálózati adatsomag küldése

Utasítás: MW

Funkció kód: 63

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 3-5
2	Kapcsolat azonosító: mindig 1
3	Adattömb kezdőcíme a PLC memóriában
4	Adattömb hossza: ennyi PLC regiszternyi adat lesz elküldve $0 < \text{érték} \leq 350$
5	Widows hibakód **

Cím	Adat
6	Cél eszköz IP címe. Pl: 192.168.1.12 = #\$C0A8010C **
7	Cél eszköz port száma: Pl: 49202 **

*: az adatot nem kell kitölteni

**: opcionális adat, az adatot csak UDP használata esetén kell kitölteni

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 3-5
2	Kapcsolat azonosító: mindig 1
3	Adattömb kezdőcíme a PLC memóriában
4	Adattömb hossza: ennyi PLC regiszternyi adat lesz fogadva $0 < \text{érték} \leq 350$
5	Widows hibakód: ha a teljesítés kódja 62, ide kerül a Windows által kküldött hiba tovább dekódolva (lásd Windows Sockets Error Codes)
6	Forrás eszköz IP címe. Pl: 192.168.1.12 = #\$C0A8010C

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz
60	Érvénytelen kapcsolat azonosító
62	Az adatküldés során hiba történt
63	A megadott kapcsolat nincs nyitva
64	Az adattömb hossza hibás: =0, vagy >350

6.23 Ablaknyitás a vezérlő képernyőjén

A vezérlő menürendszeréből kezelői beavatkozással megjeleníthető ablakokat a PLC programból MW utasítással is ki lehet választani.

Utasítás: MW

Funkció kód: 70

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 2-4
2	Ablak azonosítója: lásd táblázat
3	Csatorna száma: =0: csatornától független ablak =1...8: csatornában érvényes adatokkal jelenítse meg az ablakot
4	Paraméter1: lásd táblázat
5	Paraméter2: lásd táblázat

*: az adatot nem kell kitölteni

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 2-4
2	Ablak azonosítója: lásd táblázat
3	Csatorna száma: =0: csatornától független ablak =1...8: csatornában érvényes adatokkal jelenítse meg az ablakot
4	Paraméter1: lásd táblázat
5	Paraméter2: lásd táblázat

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz

Kód	Jelentés
6	Érvénytelen csatornaszám
70	Érvénytelen ablakszám
71	Nincs kapcsolat a kijelzővel (kijelző nem fogadóképes)

A PLC programból megjeleníthető ablakok azonosítása

Ablak neve	Azonosító	Paraméter 1	Paraméter 2
Alfanumerikus billentyűzet	100	-	-
PLC gombok	110	-	-
Ablakválasztó gombok	120	-	-
Gépi kezelőpanel	130	-	-
Aktív G kódok a legutoljára beállított nézetrel		-1	
Aktív G kódok	1000	0	Csoportszám
Aktív G kódok szöveggel	1000	1	Csoportszám
Összes G kód	1000	2	Csoportszám
Összes G kód szöveggel	1000	3	Csoportszám
Aktív M kódok a legutoljára beállított nézetrel		-1	
Aktív M kódok	1010	0	Csoportszám
Aktív M kódok szöveggel	1010	1	Csoportszám
Összes M kód	1010	2	Csoportszám
Összes M kód szöveggel	1010	3	Csoportszám
FST	1020	-	-
Összes orsó	1030	-	-
Egyedi mondatbevitel	1040	-	-
Program lista	1050	-	-
Baloldali könyvtárlablak	1090	-	-
Jobboldali könyvtárlablak	1091	-	-
Futó programok (fő- és al-) listája	1110	-	-

Ablak neve	Azonosító	Paraméter 1	Paraméter 2
Összes üzenet	1130	-	-
Idő/munkadarab számláló	1150	-	-
Pozíció	3000	-	-
Munkadarab bemérés	3011	-	-
Szerszám bemérés	3012	-	-
Maró szerszámkorrekciós táblázat	3020	-	Korr. száma
Eszterga szerszámkorrekciós táblázat	3031	-	Korr. száma
Eszterga 2. szerszámkorrekciós táblázat	3032	-	Korr. száma
Nullponteltolás ablak	3040	-	-
Szerszámkezelő táblázat	4010	-	-
Szerszámhely táblázat	4020	-	-
Egyedi szerszámszerkesztő ablak	4030	Magazinszám	Zsebszám
Szerszámalak táblázat	4040	-	-
Összesített éltartam táblázat	4050	-	-
Összes makrováltozó	5000	-	Vált. száma
#1...#33 lokális makrováltozók	5010	-	Vált. száma
#100...#499 globális makrováltozók	5020	-	Vált. száma
#500...#999 globális makrováltozók	5030	-	Vált. száma
Számológép	6000	-	-
Vágósebességszámítás	6010	-	-
Osztás számláló	6020	-	-
DXF konvertáló	6030	-	-
Orsó orientálási pozíció beállítás	6040	-	-
Orsó fázistolás beállítás	6041	-	-
Munkadarab bemérő kalibrálás	6060	-	-
Munkadarab felületmérés	6061	-	-
Munkadarab sarokmérés	6062	-	-
Munkadarab zseb/váll, furat/csap mérés	6063	-	-

6.24 Hajtásadatok kiíratása

A motorok által felvett áram értékei megjeleníthetők a pozíciókijelzőn és az FST ablakban. Az orsó motorok fordulatszáma az FST ablakban. NCT gyártmányú hajtások adatait a vezérlő automatikusan átveszi a hajtásoktól.

Idegen hajtások esetén, ha a PLC program ki tudja olvasni a hajtásból a különböző adatokat, vagy például az áramadatot A-D átalakítón keresztül be tudja olvasni, a 81-es utasítás segítségével át tudja azokat adni az NC-nek kijelzésre.

Utasítás: MW

Funkció kód: 81

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 3
2	Hajtás cím: az N0501 Axis Input Address, vagy az N0601 Spindle Input Address paraméteren beállított érték
3	Az átadott adat típusa: 0 – Motor fordulatszáma [1/min] 1 – Motor pillanatnyi áram [10mA] (125 = 1,25A) 2 – Motor terhelés (relatív áram) [%] 3 – DC sínfeszültség [V] 4 – Motor hőmérséklet [°C] 5 – Motor pillanatnyi teljesítmény [kW]
4	Az átadott adat értéke (DWORD)

*: az adatot nem kell kitölteni

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 3
2	Hajtás cím: az N0501 Axis Input Address, vagy az N0601 Spindle Input Address paraméteren beállított érték

Cím	Adat
3	Az átadott adat típusa: 0 – Motor fordulatszáma [1/min] 1 – Motor pillanatnyi áram [10mA] (125 = 1,25A) 2 – Motor terhelés (relatív áram) [%] 3 – DC sínfeszültség [V] 4 – Motor hőmérséklet [°C] 5 – Motor pillanatnyi teljesítmény [kW]
4	Az átadott adat értéke (DWORD)

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz
80	Érvénytelen hajtás cím
81	Érvénytelen adattípus
82	Érvénytelen átadott adat érték

6.25 Pozícióadatok beírása

A PLC program felülírhatja a tengelykezelőben nyilvántartott pozíciót, ha

- a tengelyen nyitott a pozíciószabályozó hurok (AN_OPNA=1), és
- az útmérő eszköztől a pozícióátvétel tiltva van (AP_EFD=1, vagy az N0514 Servo Control paraméter #4 EFD=1).

Az **utasítás hatására** a tengelykezelő **bejegyzi az adott tengelyre a referenciapont felvéve** állapotot (AN_RPE=1).

Az utasítás például akkor használható, amikor egy forgó tengelyt Hirth tárcsára ültetünk le és a pozíciónyilvántartást, mentést a PLC program végzi. Így kiválthatjuk a bekapcsolás utáni referenciapontfelvételt. Különösen akkor hasznos alkalmazni, amikor egy jeladó több tengely pozícióját is méri.

Utasítás: MW

Funkció kód: 90

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 3
2	A tengely indexe (0, ..., 31)
3	Beírandó pozíció a gépi koordinátarendszerben (double)
4	

*: az adatot nem kell kitölteni

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 3
2	A tengely indexe (0, ..., 31)
3	Beírandó pozíció a gépi koordinátarendszerben (double)
4	

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz
30	Érvénytelen tengelyindex: <0, vagy >31
90	Belső hiba az utasítás nem használható
91	Tengelyindex nem létező tengelyre mutat, vagy pozíciószabályozó hurok nincs nyitva (AN_OPNA=0), vagy az útmérő eszköztől a pozícióátvétel nincs tiltva (AP_EFD=0)

6.26 A szerszámkezelő táblázat adatainak írása és olvasása

A szerszámkezelési funkciót az alábbi esetekben kell a vezérlésen élesíteni:

- a szerszámokra éltartam vizsgálatot akarunk alkalmazni,
- a magazinban tárolt szerszámokra az alkatrészprogramból nem helykód szerint, hanem szerszámkód alapján kívánunk hivatkozni,
- ha a gépen alkalmazott szerszámcsere random tárkezelést igényel.

A szerszámkezelés adatai az NC-ben táblázatokban vannak tárolva. A PLC program az MR és az MW utasítások segítségével hozzáfér a táblázat adataihoz, olvasni és írni is tudja azokat.

6.26.1 A szerszámkezelő táblázat

A szerszámkezelő táblázatba írhatjuk be a vezérlésben alkalmazott szerszámok T kódját, azaz a típusszámukat, amelyre az alkatrészprogramban hivatkozunk.

Ebben a táblázatban adhatjuk meg az egyes szerszámok tulajdonságait, azaz, hogy normál méretűek, vagy extra nagyságúak, hogy részt vesznek-e a szerszám éltartam kezelésben, vagy sem, illetve a korrekciós tár melyik elemét rendeljük hozzájuk. Esetleg további technológiai adatokat is rendelhetünk az adott szerszámhoz, mint fordulatszám, előtolás, stb.

A szerszámkezelő táblázat globális, azaz minden csatornára közös.

Adatszám	Típus-szám (T)	Szerszám info	Alakszám	Éltartam státusz	Éltartam számláló	Éltartam
1	10002001	UENC V		Lejárt	150	150
2	10002001	SENC V		Használt	131	200
3	10002001	SENC V		Nem hasz.	0	170
4	150	SDLC V	2	Nem		
5	3210	UENT V		Törött	1h42m26s	2h30m00s
6	3210	SENT V		Használt	1h34m14s	1h50m30s
...						

A szerszámkezelő táblázat elemeit az alábbiakban foglaltuk össze. Az egyes elemek mellett feltüntettük azok helyfoglalását is:

1 Adatszám (DWORD)

A táblázat sorszáma. Nem szerkeszthető, a táblázat sorainak számát paraméter határozza meg. A szerszámhely táblázatban a szerszámkezelő táblázat sorszáma, azaz az adatszám, alapján tartjuk nyilván a szerszámokat.

2 Típuszám (DWORD)

Alkatrészprogramból T címen a Típuszámmal hivatkozunk a szerszáma. Ha a szerszámok egy csoportja részt vesz az éltartam kezelésben a csoport összes szerszámát ugyanazzal a Típuszámmal jelöljük. A Típuszámot max. 8 számjegyen adhatjuk meg:

T: 1-től 99 999 999-ig.

A kereső függvény mindig azt az adatszámú szerszámot választja ki, amelynek az éltartam számláló értéke maximális, de még nem járt le. A fenti táblázat alapján a

T10002001

hivatkozásra a 2-es adatszámú szerszámot választja. Ha az azonos Típuszámmú szerszámok mindegyikének az éltartamszámlálója egyforma értéken áll, a kisebb adatszámú szerszámot választja.

3 Szerszám info (DWORD)

A Szerszám info az alábbi bites információkat tartalmazza:

- #0 I (=0, Invalid): a szerszámkezelési táblázat teljes sora érvénytelen, törölhető
V (=1, Valid): a szerszámkezelési táblázat teljes sora érvényes.
- #1 C (=0, Count): éltartamvizsgálat a használat számára történik
T (=1, Time): éltartamvizsgálat időre történik.

Figyelem:

Ugyanazon Típuszámmú szerszámokhoz ugyanazt a C, vagy T szerszám infót kell beállítani.

- #2 N (=0, Normal): a szerszám normál méretű (egy zsebnyi helyet foglal)
L (=1, Large): a szerszám nagyméretű. több zsebet foglal el.
- #3 E (=0, Enabled): a szerszámkezelési táblázat teljes sora szerkeszthető
D (=1, Disabled): a szerszámkezelési táblázat teljes sora nem szerkeszthető.
- #4 Ha az éltartam státusz azt mondja, hogy az adott szerszám nem vesz részt az éltartam kezelésben és ez a bit
U (=0, Unsearchable): erre a szerszáma nem keres az NC
S (=1, Searchable): erre a szerszáma keres az NC

4 Alakszám (DWORD)

Ha a Szerszám infóban “L”-lel extra széles szerszámot definiálunk, meg kell határozni a szerszám alakját. Extra széles szerszámok alakját (zsebfoglalását) a szerszámalak táblázat tartalmazza.

Az alakszám a szerszámalak táblázat megfelelő sorára mutat.

Ha az alakszám értéke 0 a szerszám 1 zsebnyi helyet foglal. Ha a Szerszám infóban “N”-nel normál szerszámot definiálunk, az alakszám értéke érvénytelen.

5 Éltartam státusz (DWORD)

Az éltartam státusz az alábbi kódokat tartalmazza:

=0: Nem (Not performed)

A szerszám nem vesz részt az éltartam kezelésben, de ha a Szerszám info táblázatban “S”-nek van jelölve a keresési algoritmus T Típuszáma figyelembe veszi a szerszámot, egyébként nem.

=1: Nem használt (Not used)

Ha a szerszám részt vesz az éltartam figyelésben, de még nem volt használva, azaz az éltartam számláló értéke =0, ezt a státuszt írja ki. A keresési algoritmus figyelembe veszi.

=2: Használt (Used)

Ha a szerszám részt vesz az éltartam figyelésben és éltartam számlálójának értéke kisebb, mint a megadott éltartam, ezt a státuszt írja ki. A keresési algoritmus figyelembe veszi.

=3: Lejárt (Expired)

Ha a szerszám részt vesz az éltartam figyelésben és éltartam számlálójának értéke elérte a megadott éltartamot, ezt a státuszt írja ki. Erre a szerszámmra többé nem keres az algoritmus.

=4: Törött (Broken)

Ha a PLC az adott szerszámmra törést jelez, ezt a státuszt írja ki. A keresési algoritmus számára ugyanaz, mint a "Lejárt" státusz. Azokra a szerszámokra is vonatkozik, amelyek nem vesznek részt az éltartam kezelésben.

6 Éltartam számláló (DWORD)

Ha a Szerszám info táblázatban az éltartamvizsgálat be van kapcsolva és C-re van állítva a szerszámcseré végrehajtását számolja. Valahányszor M06, vagy T kód végrehajtása után a FIN jel megérkezik, a számláló értékét növeli 1-gyel. Amint az éltartam számláló értéke eléri az Éltartamon meghatározott értéket a szerszám státusza "Lejárt" állapotot vesz fel. Azt, hogy M06-ra, vagy T-re számoljon paraméter dönti el.

Ha a Szerszám info táblázatban az éltartamvizsgálat be van kapcsolva és T-re van állítva az éltartam figyelés időre történik. Az időt akkor méri, amikor

Start állapot van,
az override értéke nem 0,
az orsó forog,
és előtoló mozgást hajt végre.

Ha az éltartam számlálón mért idő értéke eléri az éltartam értékét a szerszám éltartam státusza "Lejárt" állapotra változik.

7 Éltartam (DWORD)

Az adott szerszámmra érvényes éltartamot lehet beállítani számra, vagy időre. Amikor egy szerszám Éltartam számlálójára eléri az itt meghatározott értéket, az Éltartam státusz állapota "Lejárt"-ra vált.

A táblázat további elemei

Adatszám	Típuszám (T)	Szerszám info	Figyelm. éltartam	H	D	S
1	10002001	UENC	20	1	1	12500
2	10002001	SENC	12	2	3	11400
3	10002001	SENC	15	31	31	13000

4	150	SDLCV		12	13	5400
5	3210	UENTV	5m30s	326	326	2500
6	3210	SENTV	4m10s	63	63	2700
...						

8 Figyelmeztető éltartam (DWORD)

Ha egy szerszámra a beállított Éltartam és az Éltartam számláló különbsége eléri az itt beállított értéket a vezérlő a PLC-nek egy figyelmeztető jelzést küld.

9 H: a hosszkorrekciós rekesz száma (DWORD)

Maró csatornában, ha a szerszám részt vesz az éltartam figyelésben, a szerszámhoz tartozó hosszkorrekciós rekesz száma (H kód).

10 D: a sugárkorrekciós rekesz száma (DWORD)

Maró csatornában, ha a szerszám részt vesz az éltartam figyelésben, a szerszámhoz tartozó sugárkorrekciós rekesz száma (D kód).

11 G: a geometriai korrekciós rekesz száma (DWORD)

Eszterga csatornában, ha a szerszám részt vesz az éltartam figyelésben, a szerszámhoz tartozó geometriai korrekciós rekesz száma.

12 W: a kopáskorrekciós rekesz száma (DWORD)

Eszterga csatornában, ha a szerszám részt vesz az éltartam figyelésben, a szerszámhoz tartozó kopáskorrekciós rekesz száma.

13 Orsó fordulatszám (DWORD)

A szerszámhoz tartozó orsó fordulatszám 1/min-ben.

A táblázat további elemei

Adatszám	Típuszám (T)	Szerszám info	F	Felhasználói, bites	Felhasználói 1	...
1	10002001	UENCV	3000			
2	10002001	UENCV	2800			
3	10002001	UENCV	3300			
4	150	SDLCV	650			
5	3210	UENTV	1800			
6	3210	UENTV	1700			
...						

14 Előtolás (double)

A szerszámhoz tartozó előtolás mm/min, inch/min, vagy mm/fordulat, inch/fordulat dimenzióban.

15 Felhasználói, bites adatok (DWORD)

Szerszámtípusonként 8 db bites adat adható meg, tetszőleges felhasználásra.

16 Felhasználói adatok (double)

Paraméteren megadott számú, lebegőpontos adat tárolására alkalmas oszlop.

6.26.2 A szerszámhely táblázat

A szerszámhely táblázatba az egyes magazinok zsebeiben lévő szerszámok adatszámát írjuk, azaz a szerszámkezelő táblázat megfelelő sorának a számát.

A szerszámhely táblázat szintén globális, minden csatornára közös táblázat.

Sorszám	Magazinszám	Zsebszám	Adatszám
1	1	1	0
2	1	2	4
3	1	3	3
...	...	...	...
24	1	24	1
25	2	1	12
26	2	2	28
27	2	3	0
...	...	...	...
40	2	16	62
...	...	...	...
41	10	1	2
42	11	1	0
43	20	1	5
44	21	1	6
...	...	...	...

A szerszámhely táblázatot a következőképp osztjuk fel:

1 Sorszám

A teljes szerszámhely táblázat sorszámát mutatja. A gépen előforduló összes szerszámzseb számával egyezik meg.

2 Magazinszám (DWORD)

Paraméteren maximum 4 különböző magazin vehető fel. Ezek sorszáma 1-től 4-ig változik. Az, hogy az egyes magazinokhoz mely cserélőkarok férnek hozzá, illetve a cserélőkarok mely orsókat tudják kiszolgálni, a gép mechanikai felépítésének kérdése.

Paraméteren lehet megadni, hogy az egyes magazinok lánc, vagy mátrix elrendezésűek, hány zsebet tartalmaznak, illetve mátrix elrendezésű magazinok esetén hány sortba vannak rendezve a zsebek.

Speciális magazinok

Ezen kívül minden, a szerszámkezelésben részt vevő orsót paraméteren hozzárendeljük a magazin táblázathoz, valamint az egyes orsók mellé egy készenléti magazint. Az orsómagazinok és a készenléti magazinok egyzsebes magazinok.

Az 1. orsó magazinszáma 10, a 2. orsómagaziné 20 és így tovább.

Az orsók mellé rendelt készenléti magazinok (pl. szerszámkarok) magazinszámai 11, 21 és így tovább.

Megjegyzés.

Előfordulhat, hogy a szerszám megmunkálási pozícióban nem orsóba kerül, hanem csak egy szerszámtartóba. Vegyünk egy olyan 2x2 tengelyes karusszel esztergát, amelyiknél a bal- és a jobboldali szerszámtartóban a szerszámok nem forgathatók. Ekkor is definiálni kell egy-egy virtuális orsót mindkét csatornában (PL: S1, S2), mert a vezérlő a PLC-NC kommunikációs jelzőket és a lehívott korrekciókat orsómagazinok alapján tartja nyilván és orsószám szerint indexeli.

3 Zsebszám DWORD

Egy magazinon belül a zsebszámok egyesével növekednek. A egyes magazinok kezdő zsebszámát paraméteren lehet megadni.

4 Adatszám DWORD

A Szerszámkezelő táblázatnak az adatszámát, vagyis sorszámát írjuk ide, amely a zsebben lévő szerszámra mutat.

Ha az adatszám értéke 0, a zsebben nincs szerszám.

6.26.3 A szerszámalak táblázat

A szerszámalak táblázatban adhatjuk meg azoknak a szerszámoknak az alakját, amelyek egy zsebnél több helyet foglalnak el a magazinban.

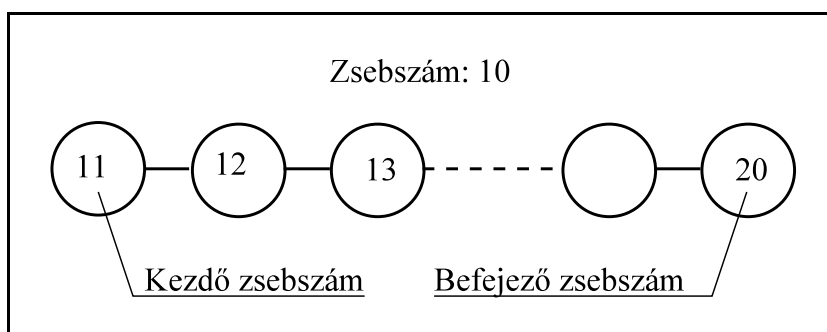
Ha egy szerszám túlméretes a Szerszámkezelő táblázat

- **Szerszám info** oszlopának #2-es bitjét “L”-re kell állítani, valamint az
- **Alakszám** oszlopba be kell írni, hogy a szerszám alakját a Szerszámalak táblázat hányadik sorában adtuk meg.

A szerszámalak táblázat használata előtt meg kell adni paraméteren az egyes magazinok típusát, azaz, hogy láncos, vagy mátrixos elrendezésűek-e.

Ezen kívül paraméteren meg kell adni

- **lánctár** esetén az egyes magazinok kezdő zsebszámát (nem kötelező 1-től kezdeni), illetve a magazinban lévő zsebek számát,



- **mátrix** elrendezésű magazin esetén meg kell adni a kezdő zsebszámot (nem kötelező 1-et adni), valamint a mátrixban lévő sorok, illetve oszlopok számát.

						Oszlopok száma
	1	2	3	4	5	
Kezdő zsebszám	21	22	23	24	25	1
	26	27	28	29	30	2
	31	32	33	34	35	3
	36	37	38	39	40	4
						Sorok száma

A vezérlő a mátrix elrendezésű magazinon belül a fenti ábrán látható módon értelmezi a zsebek, sorok és oszlopok számozását, tehát a számozás a bal felső sarokban indul és a jobb alsó sarokban ér véget.

A Szerszámalak táblázat globális, minden csatornára érvényes táblázat. A szerszámalak táblázat elemeinek száma maximum 20. Ez azt jelenti, hogy maximum 20 különböző alakú szerszámot tud a rendszer kezelni.

Alakszám	Balra	Jobbra	Fel	Le	Geometria
1					
...					

1 Alakszám DWORD

A szerszámalak táblázat sorszáma. A Szerszámkezelő táblázat Alakszám oszlopában megadott szám mutat a Szerszámalak táblázat megfelelő sorára.

2 Balra, Jobbra, Fel, Le

A Balra, Jobbra oszlopot úgy lánc-, mind mátrixtár esetén is használjuk. A Fel, Le oszlopok kitöltésének csak mátrix elrendezésű szerszámtárak esetén van értelme. Az oszlopokba írt szám azt mondja meg, hogy a szerszám a megfelelő irányban hány zsebet foglal el $\frac{1}{2}$ zseb

mértékegységben. Azokba a zsebekbe, amelyeket félig foglal le egy szerszám, természetesen már nem tehető másik szerszám. Viszont ezeknek a zsebeknek a szomszédságába még betehető olyan szerszám amelynek szintén $\frac{1}{2}$, $1\frac{1}{2}$, $2\frac{1}{2}$ zseb a helyfoglalása az ellenkező irányban.

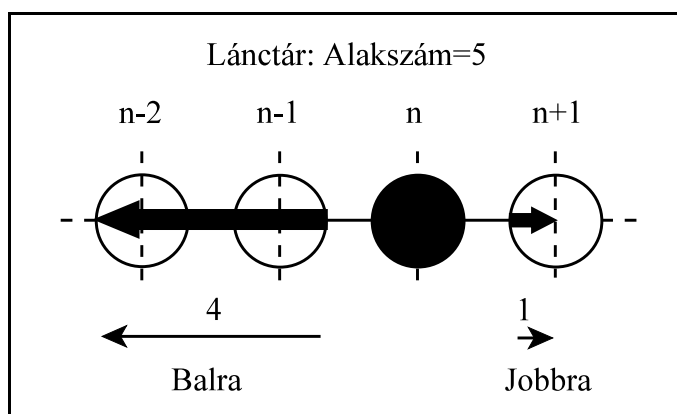
A különböző irányok értelmezése

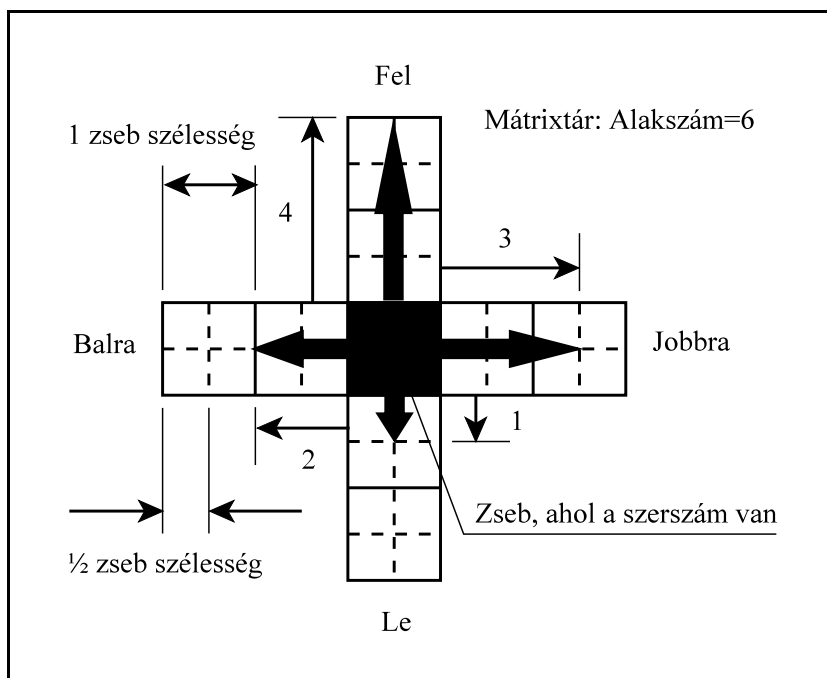
Balra: csökkenő zsebszámok (lánc- és mátrixtárban is)

Jobbra: növekvő zsebszámok (lánc- és mátrixtárban is)

Fel: csökkenő sorszámok

Le: növekvő sorszámok





A fenti ábrák példáiban a szerszámok helyfoglalását a következőképp kell megadni:

Alakszám	Balra	Jobbra	Fel	Le	Geometria
...					
5	4	1	0	0	
6	2	3	4	1	

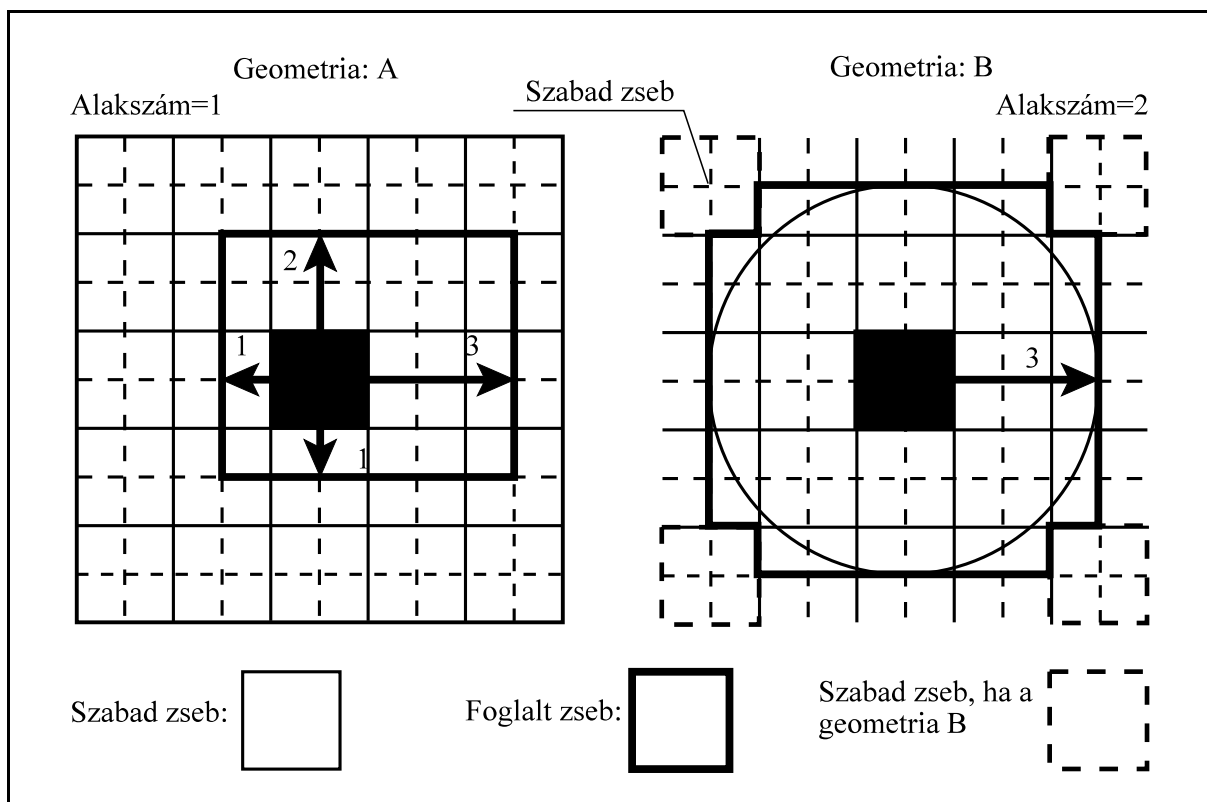
3 Geometria

A Geometria oszlopban azt adhatjuk meg, hogy a szerszám által lefoglalt terület:

A: téglalap,

B: kör

alakú. Kör alakú szerszámoknál, ha a geometriát B-nek adjuk meg a sarkokon szabad zseb lesz, szemben azzal, ha téglalapként adjuk meg a helyfoglalást.



Amint az ábrából is látszik a Geometria oszlopnak csak mátrix elrendezésű magazinok esetén van értelme. A fenti ábra alapján a táblázat kitöltése a következő:

Alakszám	Balra	Jobbra	Fel	Le	Geometria
1	1	3	2	1	A
2	3	3	3	3	B

6.26.4 Adatszám cseréje két különböző magazin két különböző zsebe között

Utasítás: MW

Funkció kód: 100

A Szerszámhely táblázat Magazinszám és Zsebszám által meghatározott két sorában lévő Adatszám értéket felcseréli.

Használata:

- a szerszámot kiveszi a. magazinból és beteszi a cserélőkarba, utána
- a szerszámot kiveszi a cserélőkarból és beteszi az orsóba, egyúttal az orsóban lévő szerszám bekerül a cserélőkarba, majd
- a cserélőkarból a szerszám bekerül a magazinba.

Minden ilyen szerszámmozgatást a Szerszámhely táblázat fenti utasítással történő rendezése követ.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 4
2	1. magazin száma
3	1. zseb száma
4	2. magazin száma
5	2. zseb száma

*: az adatot nem kell kitölteni

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz: 4
2	1. magazin száma
3	1. zseb száma
4	2. magazin száma
5	2. zseb száma

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz (nem 4)
3	Írásvédett
100	A megadott 1. magazinszám nem létezik
101	A megadott 1. zsebszám nem létezik
102	A megadott 2. magazinszám nem létezik
103	A megadott 2. zsebszám nem létezik

6.26.5 Üres zseb keresése

Utasítás: MR

Funkció kód: 101

A célmagazinba visszatérő szerszám számára keres üres zsebet, a szerszám Alakkódjának figyelembe vételével. Meg kell adni, hogy a célmagazin melyik zsebtől keressen. A forrásmagazin száma és zsebszáma azonosítja a szerszámot. Üres zsebet nem keres készenléti, illetve orsómagazinban.

 **Figyelem!** Az utasítás végrehajtását a N1274 PLC PrgConfig paraméter #0 MRW bitje befolyásolja!

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 7
2	Célmagazin száma
3	Célmagazin zsebszáma: melyik pozíciótól keressük a helyet
4	Keresés iránya a táblázatban: =0: kétirányú =1: pozitív irányba =2: negatív irányba

Cím	Adat
5	Forrásmagazin száma
6	Forrásmagazin zsebszáma
7	Kikeresett zsebszám *
8	Magazinforgatás iránya *

*: az adatot nem kell kitölteni

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz: 7
2	Célmagazin száma
3	Célmagazin zsebszáma: melyik pozíciótól keressük a helyet
4	Keresés iránya: =0: kétirányú =1: pozitív irányba =2: negatív irányba
5	Forrásmagazin száma
6	Forrásmagazin zsebszáma
7	Kikeresett zsebszám
8	Magazinforgatás iránya: =1: pozitív irányba =2: negatív irányba

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz (nem 7)
3	Írásvédett
100	A megadott célmagazinszám nem létezik

Kód	Jelentés
101	A megadott célzsebszám nem létezik
102	A megadott forrásmagazinszám nem létezik
103	A megadott forrászsebszám nem létezik
104	Nincs üres zseb: Kikeresett zsebszám=0

6.26.6 Új szerszám regisztrációja a Szerszámkezelő táblázatban

Utasítás: MW

Funkció kód: 102

Akkor használjuk, ha új szerszámot akarunk a PLC-n keresztül regisztrálni. Az utasítás megkeresi a Szerszámkezelő táblázatban az első üres, vagy érvénytelen (I) sort, azaz Adatszámot és erre jegyzi be az új szerszám adatait.

Példa az alkalmazásra: orsón keresztül, kézzel töltjük föl a magazint.

Szerszámregisztráció üzembe kapcsolunk a vezérlőn.

Egyedi mondatban T címen bevisszük a szerszám Típuszámát. A T kódon átadott Típuszám és az orsómagazin száma alapján regisztráljuk a szerszámot, majd a regisztráció után a Szerszámkezelő táblázat további elemét kézzel kitöltjük pl Alakkód, Éltartam stb.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: minimum 3, maximum 55
2	Magazin száma
3	Magazin zsebszáma: ahol a szerszám van
4	Típuszám
5	Szerszám info
6	Alakszám
7	Éltartam státusz
8	Éltartam számláló
9	Éltartam
10	Figyelmeztető éltartam

Cím	Adat
11	H: hosszkorrekciós rekesz száma maró csatornában
12	D: sugárkorrekciós rekesz száma maró csatornában
13	G: geometriai korrekciós rekesz száma eszterga csatornában
14	W: kopáskorrekciós rekesz száma eszterga csatornában
15	S: a szerszámhoz tartozó orsó fordulatszám
16	F: a szerszámhoz tartozó előtolás
17	
18	Felhasználói 1, bites
19	Felhasználói 2, lebegőpontos
20	
21	Felhasználói 3, lebegőpontos
22	
...	
...	
55	Felhasználói 20, lebegőpontos
56	

*: az adatot nem kell kitölteni

Az adathosszra minimum 3-at kell írni, mert a magazinszám, zsebszám és Típuszám adatok kitöltése kötelező. Az utasítás csak annyi oszlopot tölt ki a Szerszámkezelő táblázatban, amennyit az Adathossz paraméteren megadtunk mínusz 2 (magazinszám, zsebszám)

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz: minimum 3, maximum 55
2	Magazin száma
3	Magazin zsebszáma: ahol a szerszám van
4	Típuszám
5	Szerszám info

Cím	Adat
6	Alakszám
7	Éltartam státusz
8	Éltartam számláló
9	Éltartam
10	Figyelmeztető éltartam
11	H: hosszkorrekciós rekesz száma maró csatornában
12	D: sugárkorrekciós rekesz száma maró csatornában
13	G: geometriai korrekciós rekesz száma eszterga csatornában
14	W: kopáskorrekciós rekesz száma eszterga csatornában
15	S: a szerszámhoz tartozó orsó fordulatszám
16	F: a szerszámhoz tartozó előtolás
17	
18	Felhasználói 1, bites
19	Felhasználói 2, lebegőpontos
20	
21	Felhasználói 3, lebegőpontos
22	
...	
...	
55	Felhasználói 20, lebegőpontos
56	

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód

Kód	Jelentés
2	Érvénytelen adathossz: nem nagyobb mint 2, vagy nem kisebb, mint 55 vagy a megadott adathosszra nem írható be egész számú oszlopérték
3	Írásvédett
100	A megadott magazinszám nem létezik
101	A megadott zsebszám nem létezik
105	Típuszám hiba: nem ábrázolható 8 dec. számjegyen
106	Szerszám info hiba:
107	Alakszám hiba:
108	Éltartam státusz hiba:
109	Éltartam számláló hiba:
110	Éltartam hiba:
111	Figyelmeztető éltartam hiba:
112	H megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
113	D megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
114	G megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
115	W megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
116	S: a szerszámhoz tartozó orsó fordulatszám
117	F: a szerszámhoz tartozó előtolás
118	Felhasználói 1 hiba
119	Felhasználói 2 hiba
120	Felhasználói 3 hiba
...	
137	Felhasználói 20 hiba
138	Az éltartam táblázat betelt

6.26.7 Szerszám Szerszámkezelői adatainak átírása**Utasítás: MW****Funkció kód: 103**

Magazinszám és zsebszám alapján azonosított szerszám adatainak átírása a Szerszámkezelő táblázatban.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: minimum 3, maximum 55
2	Magazin száma
3	Magazin zsebszáma: ahol a szerszám van
4	Típusszám
5	Szerszám info
6	Alakszám
7	Éltartam státusz
8	Éltartam számláló
9	Éltartam
10	Figyelmeztető éltartam
11	H: hosszkorrekciós rekesz száma maró csatornában
12	D: sugárkorrekciós rekesz száma maró csatornában
13	G: geometriai korrekciós rekesz száma eszterga csatornában
14	W: kopáskorrekciós rekesz száma eszterga csatornában
15	S: a szerszámhoz tartozó orsó fordulatszám
16	F: a szerszámhoz tartozó előtolás
17	
18	Felhasználói 1, bites
19	Felhasználói 2, lebegőpontos
20	
21	Felhasználói 3, lebegőpontos

Cím	Adat
22	
...	
...	
55	Felhasználói 20, lebegőpontos
56	

*: az adatot nem kell kitölteni

Az adathosszra minimum 3-at kell írni, mert a magazinszám, és zsebszám adatok kitöltése kötelező és legalább a Típuszámot írni kell. Az utasítás csak annyi oszlopot ír át a Szerszámkezelő táblázatban, amennyit az Adathossz paraméteren megadtunk mínusz 2 (magazinszám, zsebszám).

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: minimum 3, maximum 55
2	Magazin száma
3	Magazin zsebszáma: ahol a szerszám van
4	Típuszám
5	Szerszám info
6	Alakszám
7	Éltartam státusz
8	Éltartam számláló
9	Éltartam
10	Figyelmeztető éltartam
11	H: hosszkorrekciós rekesz száma maró csatornában
12	D: sugárkorrekciós rekesz száma maró csatornában
13	G: geometriai korrekciós rekesz száma eszterga csatornában
14	W: kopáskorrekciós rekesz száma eszterga csatornában
15	S: a szerszámhoz tartozó orsó fordulatszám
16	F: a szerszámhoz tartozó előtolás
17	
18	Felhasználói 1, bites
19	Felhasználói 2, lebegőpontos
20	
21	Felhasználói 3, lebegőpontos
22	
...	
...	
55	Felhasználói 20, lebegőpontos
56	

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem nagyobb mint 2, vagy nem kisebb, mint 55 vagy a megadott adathosszra nem írható be egész számú oszlopérték
3	Írásvédett
100	A megadott magazinszám nem létezik
101	A megadott zsebszám nem létezik
105	Típuszám hiba: nem ábrázolható 8 dec. számjegyen
106	Szerszám info hiba:
107	Alakszám hiba:
108	Éltartam státusz hiba:
109	Éltartam számláló hiba:
110	Éltartam hiba:
111	Figyelmeztető éltartam hiba:
112	H megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
113	D megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
114	G megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
115	W megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
116	S: a szerszámhoz tartozó orsó fordulatszám
117	F: a szerszámhoz tartozó előtolás
118	Felhasználói 1 hiba
119	Felhasználói 2 hiba

Kód	Jelentés
120	Felhasználói 3 hiba
...	
137	Felhasználói 20 hiba
139	A megadott magazin zsebében nincs szerszám

6.26.8 Szerszám Szerszámkezelői adatainak beolvasása

Utasítás: MR

Funkció kód: 104

Magazinszám és zsebszám alapján azonosított szerszám adatainak beolvasása a PLC változók közé.

 **Figyelem!** Az utasítás végrehajtását a N1274 PLC PrgConfig paraméter #0 MRW bitje befolyásolja!

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: minimum 3, maximum 55
2	Magazin száma
3	Magazin zsebszáma: ahol a szerszám van
4	Típusszám*
5	Szerszám info*
6	Alakszám*
7	Éltartam státusz*
8	Éltartam számláló*
9	Éltartam*
10	Figyelmeztető éltartam*
11	H: hosszkorrekciós rekesz száma maró csatornában*
12	D: sugárkorrekciós rekesz száma maró csatornában*
13	G: geometriai korrekciós rekesz száma eszterga csatornában*

Cím	Adat
14	W: kopáskorrekciós rekesz száma eszterga csatornában*
15	S: a szerszámhoz tartozó orsó fordulatszám*
16	F: a szerszámhoz tartozó előtolás*
17	
18	Felhasználói 1, bites*
19	Felhasználói 2, lebegőpontos*
20	
21	Felhasználói 3, lebegőpontos*
22	
...	
...	
55	Felhasználói 20, lebegőpontos*
56	

*: az adatot nem kell kitölteni

Az adathosszra minimum 3-at kell írni, mert a magazinszám, és zsebszám adatok kitöltése kötelező és legalább a Típuszámot olvasni kell. Az utasítás csak annyi oszlopot olvas be a Szerszámkezelő táblázatból, amennyit az Adathossz paraméteren megadtunk minusz 2 (magazinszám, zsebszám).

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: minimum 3, maximum 55
2	Magazin száma
3	Magazin zsebszáma: ahol a szerszám van
4	Típuszám
5	Szerszám info
6	Alakszám
7	Éltartam státusz
8	Éltartam számláló

Cím	Adat
9	Éltartam
10	Figyelmeztető éltartam
11	H: hosszkorrekciós rekesz száma maró csatornában
12	D: sugárkorrekciós rekesz száma maró csatornában
13	G: geometriai korrekciós rekesz száma eszterga csatornában
14	W: kopáskorrekciós rekesz száma eszterga csatornában
15	S: a szerszámhoz tartozó orsó fordulatszám
16	F: a szerszámhoz tartozó előtolás
17	
18	Felhasználói 1, bites
19	Felhasználói 2, lebegőpontos
20	
21	Felhasználói 3, lebegőpontos
22	
...	
...	
55	Felhasználói 20, lebegőpontos
56	

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem nagyobb mint 2, vagy nem kisebb, mint 55 vagy a megadott adathosszra nem írható be egész számú oszlopérték
100	A megadott magazinszám nem létezik
101	A megadott zsebszám nem létezik

Kód	Jelentés
139	A megadott magazin zsebében nincs szerszám

6.26.9 Szerszám Szerszámkezelő adatainak törlése

Utasítás: MW

Funkció kód: 105

Magazinszám és zsebszám alapján azonosított szerszám adatainak törlését végzi a Szerszámkezelő táblázatban. Az egész sort törli.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 2
2	Magazin száma
3	Magazin zsebszáma: a szerszám azonosítója

*: az adatot nem kell kitölteni

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz: 2
2	Magazin száma
3	Magazin zsebszáma: a szerszám azonosítója

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz (nem 2)

Kód	Jelentés
3	Írásvédett
100	A megadott magazinszám nem létezik
101	A megadott zsebszám nem létezik
139	A megadott magazin zsebében nincs szerszám

6.26.10 Szerszám egy Szerszámkezelői adatának átírása

Utasítás: MW

Funkció kód: 106

Magazinszám és zsebszám alapján azonosított szerszám egy megadott adatának átírása a Szerszámkezelő táblázatban.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: minimum 4, maximum 5
2	Magazin száma
3	Magazin zsebszáma: a szerszám azonosítója
4	Adatszám: lásd az Adatszám táblázatot
5	Adat 1. szó
6	Adat 2. szó: csak akkor töltjük ki, ha lebegőpontos a beírandó adat

*: az adatot nem kell kitölteni

Az adathosszra 4-et kell írni, ha egész adatot írunk, illetve 5-öt, ha lebegőpontosat.

Az adatszámok értelmezése a Szerszámkezelő táblázatban:

Adatszám	Jelentés
1	Típuszám
2	Szerszám info
3	Alakszám

Adatszám	Jelentés
4	Éltartam státusz
5	Éltartam számláló
6	Éltartam
7	Figyelmeztető éltartam
8	H: hosszkorrekciós rekesz száma maró csatornában
9	D: sugárkorrekciós rekesz száma maró csatornában
10	G: geometriai korrekciós rekesz száma eszterga csatornában
11	W: kopáskorrekciós rekesz száma eszterga csatornában
12	S: a szerszámhoz tartozó orsó fordulatszám
13	F: a szerszámhoz tartozó előtolás
14	Felhasználói 1, bites
15	Felhasználói 2, lebegőpontos
16	Felhasználói 3, lebegőpontos
...	
33	Felhasználói 20, lebegőpontos

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz
2	Magazin száma
3	Magazin zsebszáma: a szerszám azonosítója
4	Adatszám: lásd az Adatszám táblázatot
5	Adat 1. szó
6	Adat 2. szó: csak akkor töltjük ki, ha lebegőpontos a beírandó adat

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 4, vagy 5 nem felel meg az adatszámhoz hivatkozott változó adathosszához
3	Írásvédett
100	A megadott magazinszám nem létezik
101	A megadott zsebszám nem létezik
105	Típuszám hiba: nem ábrázolható 8 dec. számjegyen
106	Szerszám info hiba:
107	Alakszám hiba:
108	Éltartam státusz hiba:
109	Éltartam számláló hiba:
110	Éltartam hiba:
111	Figyelmeztető éltartam hiba:
112	H megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
113	D megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
114	G megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
115	W megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
116	S: a szerszámhoz tartozó orsó fordulatszám
117	F: a szerszámhoz tartozó előtolás
118	Felhasználói 1 hiba
119	Felhasználói 2 hiba
120	Felhasználói 3 hiba
...	

Kód	Jelentés
137	Felhasználói 20 hiba
139	A megadott magazin zsebében nincs szerszám
140	Érvénytelen adatszám

6.26.11 Szerszám egy Szerszámkezelői adatának beolvasása

Utasítás: MR

Funkció kód: 107

Magazinszám és zsebszám alapján azonosított szerszám egy megadott adatának beolvasása a Szerszámkezelő táblázatból.

☞ **Figyelem!** Az utasítás végrehajtását a N1274 PLC PrgConfig paraméter #0 MRW bitje befolyásolja!

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: minimum 4, maximum 5
2	Magazin száma
3	Magazin zsebszáma: a szerszám azonosítója
4	Adatszám: lásd az Adatszám táblázatot
5	Adat 1. szó *
6	Adat 2. szó: csak akkor vesszük fel, ha lebegőpontos a beolvasandó adat *

*: az adatot nem kell kitölteni

Az adathosszra 3-at kell írni, ha egész adatot olvasunk, illetve 5-öt, ha lebegőpontosat.

Az adatszámok értelmezése a Szerszámkezelő táblázatban:

Adatszám	Jelentés
1	Típuszám
2	Szerszám info

Adatszám	Jelentés
3	Alakszám
4	Éltartam státusz
5	Éltartam számláló
6	Éltartam
7	Figyelmeztető éltartam
8	H: hosszkorrekciós rekesz száma maró csatornában
9	D: sugárkorrekciós rekesz száma maró csatornában
10	G: geometriai korrekciós rekesz száma eszterga csatornában
11	W: kopáskorrekciós rekesz száma eszterga csatornában
12	S: a szerszámhoz tartozó orsó fordulatszám
13	F: a szerszámhoz tartozó előtolás
14	Felhasználói 1, bites
15	Felhasználói 2, lebegőpontos
16	Felhasználói 3, lebegőpontos
...	
33	Felhasználói 20, lebegőpontos

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz
2	Magazin száma
3	Magazin zsebszáma: a szerszám azonosítója
4	Adatszám: lásd az Adatszám táblázatot
5	Adat 1. szó
6	Adat 2. szó: csak akkor érvényesi, ha lebegőpontos a kiolvasott adat

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 4, vagy 5 nem felel meg az adatszámban hivatkozott változó adathosszának
100	A megadott magazinszám nem létezik
101	A megadott zsebszám nem létezik
139	A megadott magazin zsebében nincs szerszám
140	Érvénytelen adatszám

6.26.12 Szerszám keresése Felhasználói adat alapján

Utasítás: MR

Funkció kód: 108

A megadott Felhasználói adat alapján a Szerszámkezelő táblázatban megkeresi az első szerszámot, amelynek adata egyezést mutat, és visszaadja a magazinszámot és zsebszámot.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 4, vagy 5
2	Magazin száma *
3	Magazin zsebszáma: a szerszám azonosítója *
4	Adatszám: Felhasználói adat száma, lásd az Adatszám táblázatot
5	Adat 1. szó
6	Adat 2. szó: csak akkor vesszük fel, ha lebegőpontos a keresett adat

*: az adatot nem kell kitölteni

Ha a Felhasználói 1 adatra keresünk egy szót, ha további Felhasználói lebegőpontos adatra keresünk 2 szót kell felvennünk.

A komparálást a paraméteren beállított inkremensrendszer alapján végzi. Tehát, ha pl. ISB=1, azaz 0.001 mm-es inkremensrendszer van beállítva a két adatot akkor tekinti egyezőnek, ha különbségük abszolút értékben kisebb, mint 0.001.

Az adatszámok értelmezése a Szerszámkezelő táblázatban:

Adatszám	Jelentés
14	Felhasználói 1, bites
15	Felhasználói 2, lebegőpontos
16	Felhasználói 3, lebegőpontos
...	
33	Felhasználói 20, lebegőpontos

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz: 4, vagy 5
2	Magazin száma
3	Magazin zsebszáma: a megtalált szerszám azonosítója
4	Adatszám
5	Adat 1. szó
6	Adat 2. szó

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 4, vagy 5
140	Érvénytelen adatszám
141	Nincs találat

6.27 A palettakezelő táblázat adatainak írása és olvasása

A palettakezelés adatai az NC-ben a palettakezelő táblázatban vannak tárolva. A PLC program az MR és az MW utasítások segítségével hozzáfér a táblázat adataihoz, olvasni és írni is tudja azokat. A palettakezelő táblázat alkalmazását az N3300 Pallet Contr. paraméter #0 PAL bitjén lehet engedélyezni, a bit 1-be írásával.

A palettakezelő táblázat globális, azaz minden csatornára közös.

Az alkatrészprogram utolsó két sora a következő, feltéve, ha M60-at jelölte ki a PLC a palettacsere számára:

```
Alkatrészprogram
...
...
M60 (palettacsere)
M30
```

Az M60 kódban a PLC végrehajtja az M30 utasítást, kikeresi a táblázatból a következő palettát és kijelöli a palettához rendelt programot végrehajtásra.

6.27.1 A palettakezelő táblázat

A palettakezelő táblázatban tároljuk a palettamagazinban, a munka- és a szerelőhelyen lévő, valamint a fenti helyekről törölt paletták adatait.

1-es számú palettamagazinnak nevezzük azt a tárolóhelyet, ahol a megmunkálásra kész paletták elhelyezkednek. Azt, hogy a magazin hány férőhelyes, az N3301 Pallet Pool Length paraméteren adhatjuk meg.

10-es magazinnak nevezzük a szerszámgép munkatérét, ahol a forgácsolás történik. Palettacsere esetén, pl. M60 funkcióra, a megmunkált darab visszakerül az 1-es magazinba, az új darab pedig a 10-esbe, azaz a munkatérbe kerül.

Az 1-es palettamagazin előtt elhelyezkedhet egy szerelőhely, amit **11-es magazinnak** nevezünk. Ha van szerelőhely a gépen, akkor az N3300 Pallet Contr. paraméter #1 MNT bitjét 1-be kell állítani. A szerelőhelyről, a 11-es magazinból, a megszerelt, nyers darabot tartalmazó palettát be lehet küldeni az 1-es magazinba, illetve a már kész, megmunkált darabot tartalmazó palettát kihozhatjuk a szerelőhelyre.

0-ás magazinnak, vagy **virtuális magazinnak** nevezzük a táblázat azon sorait, amely mögött nincs fizikai tárolóhely, csak a paletták adatait tárolja azonosítószámuk alapján. Ha a szerelőhelyről levesszünk egy palettát, PLC utasítással áttehetjük a paletta adatait a 0-ás magazinba. Ha új paletta kerül a szerelőhelyre PLC utasítással (pl. RFID olvasás hatására) a paletta adatait előhívhatja a 0-ás, virtuális magazinból, feltéve, hogy ezt az azonosítószámú darabot korábban már megmunkáltuk és az adatait eltároltuk a 0-ás, virtuális magazinban.

A palettakezelő táblázat 1-5 oszlopa:

Magazin- számok [] és Helyszámok	Adatszámok					
	1	2	3	4	5	..
	Paletta azonosító	Státusz	Prioritás	Végrehajtott programok száma	Kijelölt programok száma	..
Munka [10] 1						
Szerelő [11] 1						
[1] 1						
[1] 2						
...						
...						
[1] n						
[0] 1						
[0] 2						
...						

A Magazinszámok és helyszámok oszlop:

Munka sor [10] 1: a munkatérben lévő paletta adatait tartalmazza. Nem szerkeszthető automata üzemmódban. PLC szempontjából a 10-es magazin.

Szerelő sor [11] 1: a szerelőhelyen lévő paletta adatait tartalmazza. Csak akkor látható, ha az N3300 Pallet Contr. paraméter #2 MNT bitje 1. Csak akkor szerkeszthető, ha a PLC nem tiltja, például azért, mert már elindult egy palettacsere a szerelőhely és a magazin között. PLC szempontjából a 11-es magazin.

Palettamagazin [1] 1, 2, ... n helyszám sorok: Az 1-es palettamagazinban lévő paletták adatait tartalmazza. Annyi sort tartalmaz, amennyit az N3301 Pallet Pool Length paraméteren megadtunk. Csak akkor szerkeszthető, ha a PLC nem tiltja, például azért, mert már elindult egy palettacsere a munkatér és a magazin között.

Virtuális magazin [0] 1, 2, ... helyszám sorok: A 0-ás palettamagazin, a virtuális magazin, a fizikai magazinból törölt paletták adatait tartalmazza. A 0-ás magazinban elérhető sorok száma 128, levonva belőle az 1-es, 10-es 11-es magazin sorait. Csak akkor szerkeszthető, ha a PLC nem tiltja, például azért, mert már elindult egy palettacsere a munkatér és a magazin között.

A Paletta azonosító oszlop 1:

Adatszám=1, DWORD adat. A paletták tetszőleges azonosító számát tartalmazza. Beírhatja a kezelő kézzel, vagy a PLC is, például egy RFID kód beolvasása után. Ha a palettakezelő a beírt/beolvasott azonosítójú palettát talál a 0-ás magazinban, a megfelelő paletta adatait automatikusan átvezeti a táblázat megfelelő (fizikai) sorába.

Az MW41 PLC utasítás a munkahely palettaazonosítója alapján jelöli ki a programot(okat) végrehajtásra.

A Státusz oszlop 2:

Adatszám=2, DWORD érték. A palettahely állapotát tartalmazza. Beírhatja a kezelő kézzel, vagy a PLC program is.

Értéke=0: *Nem használható* A palettahely nem használható, pl. mechanikai meghibásodás miatt. Oda palettát nem szabad tenni.

Értéke=1: *Üres*. Az adott állomáson nincs sem paletta, sem paletta tartólap. Ez az eset akkor fordul elő, amikor az 1-es magazinban a palettákat egy tartólapon tároljuk és a tartólap is cserélődhet a palettával együtt, például a magazin és a szerelőhely között.

Értéke=2: *Csak tartólap*. Az adott állomáson nincs paletta, de van paletta tartólap. Ez az eset akkor fordul elő, ha például egy palettát beváltunk a munkahelyre, de oda nem kell a tartólap.

Értéke=3: *Szerelés alatt*. A paletta még nincs beváltásra engedélyezve. Amikor pl. a szerelőhelyről a paletta az 1-es magazinba kerül 3-as állapotot vehet fel. Fizikailag bekerül a beváltható paletták közé, viszont amíg nem engedélyezzük beváltásra, addig a munkahelyre nem lehet beváltani.

Értéke=4: *Beváltásra kész*. Előkészített és érvényesített paletta. A paletta készen áll a megmunkáláshoz való beváltásra.

Értéke=5: *Megmunkálás alatt*. A paletta megmunkálás alatt van, azon alkatrész program fut.

Értéke=6: *Kész*. A palettán megmunkált jó, darab van.

Értéke=7: *Selejt*. A palettán megmunkált, de selejt darab van

Értéke=8: *Újramunkálás*. A paletta javításra vissza lett küldve a munkatérbe.

A Prioritás oszlop 3:

Adatszám=3, DWORD típusú adat. Minél kisebb a prioritás értéke, annál előbb kerül sor az adott palettán lévő darab(ok) megmunkálására. Beírhatja a kezelő kézzel, vagy a PLC program is.

Ha több paletta prioritása megegyezik akkor a munkatérhez legközelebbi palettát váltja be, azaz az MR308 kereső utasítás ennek a palettának a pozícióját adja meg.

A Végrehajtott programok száma oszlop 4:

Adatszám=4, DWORD típusú adat. Az M60 utasítás végrehajtása során a PLC program inkrementálja a végrehajtott programok számát a munkahelyen, azaz a 10-es magazinban.

Amíg az MW41 programkijelölés utasítás 0-val tér vissza, nincs az összes program végrehajtva a palettán. A PLC ilyenkor nem keres új palettát a magazinban, hanem starttal elindítja a következő programot.

Ha az MW41 programkijelölés utasítás 47-tel tér vissza, új palettát kell keresni.

A Kijelölt programok száma oszlop 5:

Adatszám=5, DWORD típusú adat. Egy palettához több főprogramot is kijelölhetünk végrehajtásra. A kijelölt programok száma ebben az oszlopban található.

Amíg az MW41 programkijelölés utasítás 0-val tér vissza, nincs az összes program végrehajtva a palettán. Ha a végrehajtott programok száma=kijelölt programok száma, az utasítás 47-es kóddal tér vissza.

A palettakezelő táblázat 6-10 oszlopa:

Magazin- számok [] és Helyszámok	Adatszámok					
	..	6	7	8	9	10
		Programok azonosítója	Felhasz- nálói 1	Felhasz- nálói 2	Felhasz- nálói 3	Felhasz- nálói 4
Munka [10] 1						
Szerelő [11] 1						
[1] 1						
[1] 2						
...						
...						
[1] n						
[0] 1						
[0] 2						
...						

A Programok azonosítója oszlop 6:

Adatszám=6

Ha az N3300 Pallet Contr. paraméter #3 PRI=0:

A kijelölt programokat 4-, vagy 8-jegyű programszámok azonosítják. A programoknak a Programs könyvtár gyökerében kell lenniük, Ooooo.nct, vagy Ooooooooo.nct fájlneven.

Ebben az esetben az adatszámhoz tartozó változó DWORD típusú és a PLC program tudja írni illetve olvasni az oszlopot.

Ha az N3300 Pallet Contr. paraméter #3 PRI=1:

A végrehajtandó programot(okat) fájlnevük és elérési útjuk alapján kézzel választjuk ki a palettakezelő táblázat használatával. Ebben az esetben a PLC program nem tudja írni, vagy olvasni ezt az oszlopot.

Az N3302 No. of Custom Columns paraméteren max. 4 felhasználói oszlopot is ki lehet jelölni

A Felhasználói 1, oszlop 7:

Adatszám=7, Bites adat, 8 bit hosszúságú.

Akkor használható, ha az N3302 No. of Custom Columns paraméter>0.

A Felhasználói 2-4, oszlop 8-10:

Adatszám=8-10, double típusú adat.

Akkor használható, ha az N3302 No. of Custom Columns paraméter>1.

6.27.2 Adatok cseréje két különböző palettamagazin két különböző helye között**Utasítás: MW****Funkció kód: 300**

A palettamagazin táblázat magazinszám és helyszám által meghatározott két sorában lévő adatokat felcseréli.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 3
2	1. palettamagazin száma (1, 10, 11)
3	1. helyszáma
4	2. palettamagazin száma (0, 10, vagy 11)

*: az adatot nem kell kitölteni

Ha a palettamagazin és a munkahely között kell cserélni adatot

- 2 1. palettamagazin száma=1
- 3 1. magazin helyszáma= a megfelelő zseb száma
- 4 2. palettamagazin száma=10

Ha a palettamagazin és a szerelőhely között kell cserélni adatot

- 2 1. palettamagazin száma=1
- 3 1. magazin helyszáma= a megfelelő zseb száma
- 4 2. palettamagazin száma=11

Ha a szerelőhely és a virtuális magazin között kell cserélni adatot

- 2 1. palettamagazin száma=11
- 3 1. magazin helyszáma= 1
- 4 2. palettamagazin száma=0

Ha a palettamagazin és a virtuális magazin között kell cserélni adatot

- 2 1. palettamagazin száma=1
- 3 1. magazin helyszáma= a megfelelő zseb száma
- 4 2. palettamagazin száma=0

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz: 3
2	1. palettamagazin száma
3	1. helyszáma
4	2. palettamagazin száma

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz (nem 3)
3	Írásvédett
300	A 1. megadott paletta magazinszám nem létezik
301	A megadott palettahely nem létezik
302	A megadott 2. magazinszám nem létezik

6.27.3 Paletta adatainak átírása

Utasítás: MW

Funkció kód: 303

Magazinszám és helyszám alapján azonosított paletta adatainak átírása a Palettakezelő táblázatban.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: minimum 3, maximum 14
2	Palettamagazin száma
3	Palettamagazin helyszáma: ahol a paletta van
4	Paletta azonosító
5	Státusz
6	Prioritás
7	Végrehajtott programok száma
8	Kijelölt programok száma
9	Felhasználói 1
10	Felhasználói 2 (double)
11	
12	Felhasználói 3 (double)
13	
14	Felhasználói 4 (double)
15	

*: az adatot nem kell kitölteni

Az adathosszra minimum 3-at kell írni, mert a magazinszám, és helyszám adatok kitöltése kötelező és legalább a Paletta azonosítót írni kell. Az utasítás csak annyi oszlopot ír át a Palettakezelő táblázatban, amennyit az Adathossz paraméteren megadtunk mínusz 2 (magazinszám, helyszám). Ha a munkahelyen (magazinszám 10), vagy a szerelőhelyen (magazinszám 11) lévő paletta adatait írjuk át a helyszám 1.

Az utasítás a **Programok azonosítóját nem kezeli**, azt csak egyedi palettaadat írással, MW306 utasítással lehet elérni, abban az esetben, ha az N3300 Pallet Contr. paraméter #3 PRI bitje 0.

A **Paletta azonosító átírására a virtuális paletta táblázatból a rendszer áthelyezi az összes adatot** és felülírja az esetleges többi adatot, feltéve, hogy talál ilyen azonosítójú sort.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: minimum 3, maximum 14
2	Palettamagazin száma
3	Palettamagazin helyszáma: ahol a paletta van
4	Paletta azonosító
5	Státusz
6	Prioritás
7	Végrehajtott programok száma
8	Kijelölt programok száma
9	Felhasználói 1
10	Felhasználói 2 (double)
11	
12	Felhasználói 3 (double)
13	
14	Felhasználói 4 (double)
15	

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem nagyobb mint 2, vagy nem kisebb, mint 14

Kód	Jelentés
3	Írásvédelem
300	A megadott palettamagazinszám nem létezik
301	A megadott helyszám nem létezik
305	Érvénytelen palettaazonosító (<1, vagy >99999999)
306	Érvénytelen paletta státusz (>9)
307	Prioritás hiba (>128)
308	Végrehajtott programok száma értékhatáron kívül (nagyobb mint a kijelölt programok száma)
309	Programok száma értékhatáron kívül (> 64)
339	A palettahely üres

6.27.4 Paletta adatainak beolvasása

Utasítás: MR

Funkció kód: 304

Magazinszám és helyszám alapján azonosított paletta adatainak beolvasása a Palettakezelő táblázatból.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: minimum 3, maximum 14
2	Palettamagazin száma
3	Palettamagazin helyszáma: ahol a paletta van
4	Paletta azonosító*
5	Státusz*
6	Prioritás*
7	Végrehajtott programok száma*
8	Kijelölt programok száma*
9	Felhasználói 1*
10	Felhasználói 2* (double)
11	
12	Felhasználói 3* (double)
13	
14	Felhasználói 4* (double)
15	

*: az adatot nem kell kitölteni

Az adathosszra minimum 3-at kell írni, mert a magazinszám, és helyszám adatok kitöltése kötelező és legalább a Paletta azonosítót olvasni kell. Az utasítás csak annyi oszlopot olvas be a Palettakezelő táblázatból, amennyit az Adathossz paraméteren megadtunk mínusz 2 (magazinszám, helyszám). Ha a munkahelyen (magazinszám 10), vagy a szerelőhelyen (magazinszám 11) lévő paletta adatait olvassuk be a helyszám 1.

Az utasítás a **Programok azonosítóját nem kezeli**, azt csak egyedi palettaadat olvasással, MR307 utasítással lehet elérni, abban az esetben, ha az N3300 Pallet Contr. paraméter #3 PRI bitje 0.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: minimum 3, maximum 14
2	Palettamagazin száma
3	Palettamagazin helyszáma: ahol a paletta van
4	Paletta azonosító
5	Státusz
6	Prioritás
7	Végrehajtott programok száma
8	Kijelölt programok száma
9	Felhasználói 1
10	Felhasználói 2 (double)
11	
12	Felhasználói 3 (double)
13	
14	Felhasználói 3 (double)
15	

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem nagyobb mint 2, vagy nem kisebb, mint 14
3	Írásvédett
300	A megadott palettamagazinszám nem létezik
301	A megadott helyszám nem létezik
339	A palettahely üres

6.27.5 Paletta egy palettakezelő adatának átírása

Utasítás: MW

Funkció kód: 306

Palettamagazinszám és helyszám alapján azonosított paletta egy megadott adatának átírása a palettakezelő táblázatban.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 4, vagy 5
2	Palettamagazin száma
3	Palettamagazin helyszáma
4	Adatszám: lásd az Adatszám táblázatot
5	Adat (ha az adat double, 2 helyet kell foglalni)
6	

*: az adatot nem kell kitölteni

Ha a munkahelyen (magazinszám 10), vagy a szerelőhelyen (magazinszám 11) lévő paletta adatait írjuk át a helyszám 1.

A palettaazonosító átírására a választható paletták táblázatból a rendszer áthelyezi az összes adatot és felülírja az esetleges többi adatot.

A Programok azonosítóját (adatszám=6) csak akkor szabad írni, ha az N3300 Pallet Contr. paraméter #3 PRI bitje 0.

Az írás az előző kijelöléseket felülírja.

Egyszerre több programot is ki lehet jelölni. Ebben az esetben előbb a Kijelölt programok száma adatot kell beírni arra az értékre, ahány programot akarunk kijelölni. Ezután az MW306 utasítás adathossz paraméterét megnövelni annak függvényében hány programszámot akarunk felvenni. A programszámokat egy MW306 utasítással lehet feltölteni.

Az adatszámok értelmezése a Palettakezelő táblázatban:

Adatszám	Jelentés
1	Paletta azonosító
2	Státusz
3	Prioritás

Adatszám	Jelentés
4	Végrehajtott programok száma
5	Kijelölt programok száma
6	Programok azonosítója
7	Felhasználói 1
8	Felhasználói 2
9	Felhasználói 3
10	Felhasználói 4

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz: 4, vagy 5
2	Palettamagazin száma
3	Palettamagazin helyszáma
4	Adatszám: lásd az Adatszám táblázatot
5	Adat (ha az adat double, 2 helyet kell foglalni)
6	

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 4, vagy 5
3	Írásvédett
300	A megadott palettamagazin szám nem létezik
301	A megadott helyszám nem létezik
305	Érvénytelen palettaazonosító (<1, vagy >99999999)
306	Érvénytelen paletta státusz (>10)
307	Prioritás hiba (>128)

Kód	Jelentés
308	Végrehajtott programok száma értékhatáron kívül (nagyobb mint a kijelölt programok száma)
309	Programok száma hiba (>64)
339	A palettahely üres

6.27.6 Paletta egy palettakezelő adatának beolvasása

Utasítás: MR

Funkció kód: 307

Palettamagazinszám és helyszám alapján azonosított paletta egy megadott adatának beolvasása a palettakezelő táblázatból.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 4, vagy 5
2	Palettamagazin száma
3	Palettamagazin helyszáma
4	Adatszám: lásd az Adatszám táblázatot
5	Adat (ha az adat double, 2 helyet kell foglalni)*
6	

*: az adatot nem kell kitölteni

Ha a munkahelyen (magazinszám 10), vagy a szerelőhelyen (magazinszám 11) lévő paletta adatait olvassuk be, a helyszám 1.

A Programok azonosítóját (adatszám=6) csak akkor lehet olvasni, ha az N3300 Pallet Contr. paraméter #3 PRI bitje 0.

Először a Kijelölt programok száma adatot kell beolvasni. Ezután az MR307 utasítás adathossz paraméterét megnövelni annak függvényében hány programszámot kell beolvasni. A programszámokat egy MR307 utasítással lehet feltölteni.

Az adatszámok értelmezése a Palettakezelő táblázatban:

Adatszám	Jelentés
1	Paletta azonosító
2	Státusz
3	Prioritás
4	Végrehajtott programok száma
5	Kijelölt programok száma
6	Programok azonosítója
7	Felhasználói 1
8	Felhasználói 2
9	Felhasználói 3
10	Felhasználói 4

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz 4, vagy 5
2	Palettamagazin száma
3	Palettamagazin helyszáma
4	Adatszám: lásd az Adatszám táblázatot
5	Adat (ha az adat double, 2 helyet kell foglalni)
6	

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 4, vagy 5
3	Írásvédett

Kód	Jelentés
300	A megadott palettamagazin szám nem létezik
301	A megadott helyszám nem létezik
339	A palettahely üres

6.27.7 Paletta keresése adatszám értéke alapján

Utasítás: MR

Funkció kód: 308

Az utasítás kikeresi a bemenő paraméteren megadott adatszámú és értékű paletta helyét az 1-es palettamagazinból és visszaadja a kikeresett helyszámot.

Ha a 2-es adatszámmon 4-es státuszú, Beváltásra kész palettahelyre keresünk és több, ugyanolyan státuszú paletta is van a magazinban, a legnagyobb prioritású (amelynek a prioritásszáma a legkisebb) paletta helyét adja vissza. Ha több azonos prioritású paletta is van a magazinban, a megadott kiindulási pozícióhoz legközelebbi paletta helyét adja vissza.

Ha a 2-es adatszámmon nem 4-es státuszú palettahelyre keresünk, vagy más adatszámmon keresünk egy adott értékű palettát, a megadott kiindulási pozícióhoz legközelebbi paletta helyét adja vissza.

A palettamagazin mechanikai kialakításának függvényében, azaz csak egy vagy mindkét irányban forgatható, bemenő paraméterként megadhatjuk a keresés irányát is. Ennek függvényében az utasítás visszaadja a magazinforgatás irányát.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 6
2	Adatszám
3	Keresendő érték
4	Palettamagazin helyszáma: melyik pozíciótól keressen
5	Keresés iránya a táblázatban: =0: kétirányú =1: pozitív irányba =2: negatív irányba
6	Kikeresett helyszám *
7	Magazinforgatás iránya *

*: az adatot nem kell kitölteni

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja: lásd táblázat
1	Adathossz: 6
2	Adatszám
3	Keresendő érték
4	Palettamagazin helyszáma: melyik pozíciótól keressen
5	Keresés iránya a táblázatban
6	Kikeresett helyszám
7	Magazinforgatás =1: pozitív irányba =2: negatív irányba

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 2
12	Adatszám hiba
301	A megadott helyszám nem létezik
339	A palettahely üres (belső hiba)
341	Nincs a keresett státuszú paletta a magazinban

6.27.8 Paletta magazin adott adatszámú értékeinek kiolvasása**Utasítás: MR****Funkció kód: 309**

Palettakezelő táblázatból egy magazin tetszőleges oszlopának értékeit lehet kiolvasni az utasítás segítségével. Meg lehet adni, hogy a kiolvasást melyik magazinpozíciótól (a táblázat melyik sorától) kezdje és hány sort olvasson ki. Ha a kiolvasás a magazin végére ér, és a megadott darabszám még nem fogyott el, a kiolvasást a táblázat elejétől folytatja.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: $n \geq 4$
2	Adatszám: lásd az Adatszám táblázatot
3	Palettamagazin száma
4	Palettamagazin helyszáma
5	A megadott helyszámon lévő érték*
...	...*
4+n	...*

*: az adatot nem kell kitölteni

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: $n \geq 4$
2	Adatszám: lásd az Adatszám táblázatot
3	Palettamagazin száma
4	Palettamagazin helyszáma
5	A megadott helyszámon lévő érték
...	...
4+n	...

Példa:

Legyen az 1-es palettamagazin 5 férőhelyes és az adatokat a 4-es helyszámtól kezdve olvassuk ki. A táblázat első oszlopát, azaz a paletta azonosító kódokat olvassuk ki. A kiolvasás után kapott eredmény a következő lesz:

- 0: 0 (hibátlan művelet)
- 1: Adathossz: 10
- 2: Adatszám 1 (Paletta azonosító)
- 3: Palettamagazin száma: 1
- 4: Palettamagazin helyszáma: 4
- 5: Kiolvasott azonosító: 12345679 (4. hely)
- 6: Kiolvasott azonosító: 98764 (5. hely)
- 7: Kiolvasott azonosító: 123 (1. hely)
- 8: Kiolvasott azonosító: 456 (2. hely)
- 9: Kiolvasott azonosító: 789 (3. hely)
- 10: Kiolvasott azonosító: 12345679 (4. hely)
- 11: Kiolvasott azonosító: 98764 (5. hely)

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem $n \geq 4$
12	Adatszám hiba
300	A megadott palettamagazin szám nem létezik
301	A megadott helyszám nem létezik
339	A palettahely üres (belső hiba)

6.27.9 Palettaazonosítójával elérhető program kijelölése automata végrehajtásra

Utasítás: MW

Funkció kód: 41

Az írási művelet csak akkor hajtódik végre, amikor automata üzemben nem fut program és nincs is megszakítási állapot az adott csatornában, azaz a következő feltételek igazak:

CN_START=CN_STOP=CN_INTD=0.

Mindig a munkatérben (10-es magazinban) lévő paletta azonosítóját kell megadni.

Ha több program van a palettán kijelölve a táblázat Végrehajtott programok száma alapján jelöli ki a következő programot.

Bemenő paraméterek:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 2
2	A 10-es magazinban lévő Palettaazonosító
3	Csatorna száma: 1...8

*: az adatot nem kell kitölteni

Az adathosszra mindig 2-öt kell írni.

Kimenő paraméterek:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 2
2	Palettaazonosító
3	Csatorna száma: 1...8

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 2
41	Érvénytelen csatonra szám
45	A fájl nem létezik
46	Az adott csatornán még program végrehajtás fut
47	A palettán nincsen végrehajtható program
342	A megmunálási hely üres (nincsen benne paletta)

Ha több program van a palettán kijelölve végrehajtásra és az MW41 utasítás teljesítési kódja 0, még nem hajtott végre minden programot és az M60 palettacsere funkcióban starttal el lehet indítani a következő programot.

Ha az MW41 utasítás teljesítési kódja 47 minden programot végrehajtott a palettán és az M60 funkciónak új, beváltásra kész palettát kell keresnie. Ezután az új palettára kell alkalmazni az MW41 programkijelölési utasítást.

6.28 Mailbox kommunikáció a PLC program és tetszőleges EtherCAT eszköz között

Nem NCT gyártmányú, ismert protokolt alkalmazó eszköz EtherCAT mailboxai elérhetők a PLC programból az MR201, MW202 utasításokon keresztül.

A használt protokollok a következők lehetnek:

SoE: Sercos over EtherCAT,

CoE: CANopen over EtherCAT,

VoE: Vendor over EtherCAT (gyártó által definiált protokoll)

A mailbox kommunikációs utasítások használatára például a következő esetekben lehet szükség:

SoE, CoE hajtások hibatörlése, áram,

SoE, CoE hajtások áram, fordulatszám adatainak kiolvasására és kijelzésére.

6.28.1 EtherCAT mailbox adatainak olvasása

Utasítás: MR

Funkció kód: 201

Bemenő paraméterek:

VoE protokoll esetén:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 4 (VoE)
2	Protokoll kód: 15
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja

*: az adatot nem kell kitölteni

CoE protokoll, amikor nincs CompleteAccess és nem érdekes a hiba kód:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 5 (CoE)
2	Protokoll kód: 3
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban

Cím	Adat
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)

*: az adatot nem kell kitölteni

CoE protokoll, amikor nincs CompleteAccess:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 6 (CoE)
2	Protokoll kód: 3
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód *

*: az adatot nem kell kitölteni

SoE protokoll:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 7 (SoE)
2	Protokoll kód: 5
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	IDN: 2 Byte (0-15 bit): elem azonosítója
	Drive Index: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód *
8	ElementFlags **

*: az adatot nem kell kitölteni

** ElementFlags

Az IDN-en kiválasztott elemnek az ElementFlags biteken beállított adatai kérdezhetők le. Akár az összes adat lekérhető.

Az ElementFlags adatainak értelmezése bitenként:

bit	elnevezés
0	DataState (paraméter állapot)
1	Name (név ASCII kódolású)
2	Attribute (attribútum)
3	Unit (mértékegység)
4	Min (minimális érték)
5	Max (maximális érték)
6	Value (aktuális érték)
7	DefaultValue (alapérték)

CoE protokoll:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 7 (CoE)
2	Protokoll kód: 3
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód *
8	CompleteAccess (0/1 lehet) **

*: az adatot nem kell kitölteni

**** CompleteAccess:**

- Ha a CompleteAccess=0, az Index és a SubIndex mezőben meghatározott objektumot tölti le.
- Ha a CompleteAccess=1 és a SubIndex=0, vagy 1, az Index mezőben megadott összes objektumot letölti.

Kimenő paraméterek:

VoE protokoll esetén:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 4 (VoE)
2	Protokoll kód: 15
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja

CoE protokoll, amikor nincs CompleteAccess és nem érdekes a hiba kód:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 5 (CoE)
2	Protokoll kód: 3
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)

CoE protokoll, amikor nincs CompleteAccess:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 6 (CoE)
2	Protokoll kód: 3

Cím	Adat
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód

SoE protokoll:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 7 (SoE)
2	Protokoll kód: 5
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	IDN: 2 Byte (0-15 bit): elem azonosítója
	Drive Index: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód
8	ElementFlags

CoE protokoll:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 7 (CoE)
2	Protokoll kód: 3
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)

Cím	Adat
	SubIndex: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód
8	CompleteAccess (0/1 lehet)

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz: nem 3
200	Nincs megadott indexű EtherCAT eszköz
201	A SOE, vagy COE kommunikációt nem támogatja a megadott indexű eszköz
202	A megadott PLCmemória cím, vagy a mailbox adathossz hibás
203	Olvasás időn túl
204	
205	Eszköz hibát üzent: az eszköz átlal visszaküldött hibakód kiolvasható a 7-es, eszköz által visszaadott hibakód mezőből

6.28.2 EtherCAT mailbox adatainak írása

Utasítás: MW

Funkció kód: 202

Bemenő paraméterek:

VoE protokoll esetén:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 4 (VoE)
2	Protokoll kód: 15
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja

*: az adatot nem kell kitölteni

CoE protokoll, amikor nincs CompleteAccess és nem érdekes a hiba kód:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 5 (CoE)
2	Protokoll kód: 3
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)

*: az adatot nem kell kitölteni

CoE protokoll, amikor nincs CompleteAccess:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 6 (CoE)
2	Protokoll kód: 3

Cím	Adat
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód *

*: az adatot nem kell kitölteni

SoE protokoll:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 7 (SoE)
2	Protokoll kód: 5
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	IDN: 2 Byte (0-15 bit): elem azonosítója
	Drive Index: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód *
8	ElementFlags **

*: az adatot nem kell kitölteni

** ElementFlags

Az IDN-en kiválasztott elemnek az ElementFlags biteken beállított adatai írhatók. Akár az összes adat is beírható.

Az ElementFlags adatainak értelmezése bitenként:

bit	elnevezés
0	DataState (paraméter állapot)
1	Name (név ASCII kódolású)
2	Attribute (attribútum)

3	Unit (mértékegység)
4	Min (minimális érték)
5	Max (maximális érték)
6	Value (aktuális érték)
7	DefaultValue (alapérték)

CoE protokoll:

Cím	Adat
0	Teljesítés kódja *
1	Adathossz: 7 (CoE)
2	Protokoll kód: 3
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód *
8	CompleteAccess (0/1 lehet) **

*: az adatot nem kell kitölteni

** CompleteAccess:

- Ha a CompleteAccess=0, az Index és a SubIndex mezőben meghatározott objektumot írja.
- Ha a CompleteAccess=1 és a SubIndex=0, vagy 1, az Index mezőben megadott összes objektumot írja.

Kimenő paraméterek:

VoE protokoll esetén:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 4 (VoE)
2	Protokoll kód: 15

Cím	Adat
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja

CoE protokoll, amikor nincs CompleteAccess és nem érdekes a hiba kód:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 5 (CoE)
2	Protokoll kód: 3
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)

CoE protokoll, amikor nincs CompleteAccess:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 6 (CoE)
2	Protokoll kód: 3
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód

SoE protokoll:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 7 (SoE)
2	Protokoll kód: 5
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	IDN: 2 Byte (0-15 bit): elem azonosítója
	Drive Index: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód
8	ElementFlags

CoE protokoll:

Cím	Adat
0	Teljesítés kódja
1	Adathossz: 7 (CoE)
2	Protokoll kód: 3
3	ECAT eszköz index: az EtherCAT ablakban látható eszköz sorszáma
4	Adat hossza az eszköz számára Byte-ban
5	PLC memória cím, ahova az adatot várja
6	Index: 2 Byte (0-15 bit)
	SubIndex: 1 Byte (16-23 bit)
7	Az eszköz által visszaadott hibakód
8	CompleteAccess (0/1 lehet)

A teljesítés kódja a következő lehet:

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód

Kód	Jelentés
2	Érvénytelen adathossz: nem 3
200	Nincs megadott indexű EtherCAT eszköz
201	A SOE, vagy COE kommunikációt nem támogatja a megadott indexű eszköz
202	A megadott PLCmemória cím, vagy a mailbox adathossz hibás
203	
204	Írás időn túl
205	Eszköz hibát üzent: az eszköz által visszaküldött hibakód kiolvasható a 7-es, eszköz által visszaadott hibakód mezőből

6.29 Az MR, MW utasítások teljesítési kódjai

Kód	Jelentés
0	Normál végrehajtás
1	Érvénytelen Funkció kód
2	Érvénytelen adathossz
3	Írásvédett
4	Tár kontrol szumma hibás
6	Érvénytelen csatornaszám
10	A változó azonosítója nem 0...1023 közé esik
11	Változó kezdőcíme nem nagyobb, v. egyenlő, mint PLCNVRAM
12	Adatszám hiba
20	Nem létező makróváltozóra hivatkozás
21	A makróváltozó nem globális (makróváltozó indexe = 0)
22	A makróváltozó globális (makróváltozó indexe > 0)
23	Rossz írási/olvasási kód: a makróváltozó formátuma (DWORD, double) nincs összhangban az írás/olvasás kódjával
24	A makróváltozó nem olvasható
25	A makróváltozó nem írható
30	Nem létező paraméterre történő hivatkozás/érvénytelen tengelyindex
31	A paraméter nem globális (paraméter indexe = 0)
32	A paraméter globális (paraméter indexe > 0)
33	Rossz írási/olvasási kód: a paraméter formátuma (bit, DWORD, double) nincs összhangban az írás/olvasás kódjával
40	Nem létező program: a megadott program nincs a tárban
41	Érvénytelen csatonra szám

Kód	Jelentés
45	A fájl nem létezik
46	Az adott csatornán még program végrehajtás fut
47	A palettán nincsen végrehajtható program
60	Érvénytelen kapcsolat azonosító
61	Kapcsolat létrehozása sikertelen
62	Az adatfogadás során hiba történt
63	A megadott kapcsolat nincs nyitva
64	Az adattömb hossza hibás: =0, vagy >350
65	Nem érkezett adat
66	Hibás bejövő paraméterek
70	Érvénytelen ablakszám
71	Nincs kapcsolat a kijelzővel (kijelző nem fogadóképes)
80	Érvénytelen hajtás cím
81	Érvénytelen adattípus
82	Érvénytelen átadott adat érték
90	Belső hiba az utasítás nem használható
91	Tengelyindex nem létező tengelyre mutat, vagy pozíciószabályozó hurok nincs nyitva (AN_OPNA=0), vagy az útmérő eszköztől a pozícióátvétel nincs tiltva (AP_EFD=0)
100	A megadott 1. magazinszám nem létezik
101	A megadott 1. zsebszám nem létezik
102	A megadott 2. magazinszám nem létezik
103	A megadott 2. zsebszám nem létezik
104	Nincs üres zseb: Kikeresett zsebszám=0

Kód	Jelentés
105	Típuszám hiba: nem ábrázolható 8 dec. számjegyen
106	Szerszám info hiba:
107	Alakszám hiba:
108	Éltartam státusz hiba:
109	Éltartam számláló hiba:
110	Éltartam hiba:
111	Figyelmeztető éltartam hiba:
112	H megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
113	D megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
114	G megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
115	W megadás hiba: értéke nagyobb, mint a korrekciós táblázat hossza
116	S: a szerszámhoz tartozó orsó fordulatszám
117	F: a szerszámhoz tartozó előtolás
118	Felhasználói 1 hiba
119	Felhasználói 2 hiba
120	Felhasználói 3 hiba
...	
137	Felhasználói 20 hiba
138	Az éltartam táblázat betelt
139	A megadott magazin zsebében nincs szerszám
140	Érvénytelen adatszám
141	Nincs találat
200	Nincs megadott indexű EtherCAT eszköz

Kód	Jelentés
201	A SOE, vagy COE kommunikációt nem támogatja a megadott indexű eszköz
202	A megadott PLCmemória cím, vagy a mailbox adathossz hibás
203	Olvasás időn túl
204	Írás időn túl
205	Eszköz hibát üzent: az eszköz átlal visszaküldött hibakód kiolvasható a 7-es, eszköz által visszaadott hibakód mezőből
300	A 1. megadott paletta magazinszám nem létezik
301	A megadott palettahely nem létezik
302	A megadott 2. magazinszám nem létezik
305	Érvénytelen palettaazonosító (<1, vagy >99999999)
306	Érvénytelen paletta státusz (>9)
307	Prioritás hiba (>128)
308	Végrehajtott programok száma értékhatáron kívül (nagyobb mint a kijelölt programok száma)
309	Programok száma értékhatáron kívül (> 64)
339	A palettahely üres
341	Nincs a keresett státuszú paletta a magazinban
342	A megmunkálási hely üres (nincsen benne paletta)

7 A PLC program és az NC közötti kommunikáció

A PLC program és az NC, illetve a külvilág közötti kommunikáció a PLC program által használt memórián keresztül történik. A memóriaterület a Státuszbiteket (FLAGS) tartalmazó szót követően kezdődik és a PLCNVRAM címig tart. Ez a memóriaterület tartalmazza az úgynevezett **NC szimbólumokat**.

Ezen a memóriaterületen, az NC szimbólumokon keresztül kommunikál egymással a PLC program és

- az NC perifériák, azaz a ki-, és bemeneti hardver egységek, úgymint:
 - gépi kezelőpanel(ek),
 - kézikerek(ek),
 - kétállapotú és analóg interfész ki-, és bemenetek,
 - tapintók,
 - EtherCAT hajtások,
 - jeladó fogadó, illetve analóg, vagy CAN buszos hajtásvezérlő egységek,
- az NC egyes szoftver moduljai, úgymint:
 - különböző szolgáltatások moduljai, mint pl. funkciógombok, PLC paraméterek, stb,
 - globális,
 - tengelykezelő,
 - orsókezelő,
 - csatornakezelő modul.

A ki-, és bemeneti hardver egységek címét a megfelelő elem EtherCAT beállítási paneljén állíthatjuk be a vezérlőn, és ezek a címek felelnek meg a PLC memória megfelelő területének.

Az NC szoftver moduljaival való kommunikációs jelzők közül a globális, vagy közös jelzők nem indexelődnek, szemben a tengely-, orsó, vagy csatornakezelő változókkal, amelyek tengelyenként, orsónként, vagy csatornánként indexelődnek. Ezen azt értjük, hogy a pusztán szimbólumra való hivatkozás, mindig az első tengelyre, orsóra, csatornára hivatkozik. A többi tengely, orsó, vagy csatorna megfelelő változója indexelt hivatkozással érhető el.

Az NC szimbólum lehet

- bites
- duplaszavas (DWORD), vagy
- lebegőpontos (double)

érték.

Minden egyes kommunikációs memóriaterületre szimbolikusan hivatkozunk. Ismertetőnkben a kommunikációs memória, azaz az NC szimbólumok fizikai címét nem adjuk meg, mert az változhat a különböző szoftver verziók során.

A fenti ok miatt a PLC programban a FLAGS címtől a PLCNVRAM címig tartó memóriaterületre nem definiálhatunk fix című szimbólumokat, hanem csak olyanokat, amelyek egy NC szimbólumhoz képesti relatív hivatkozást tartalmaz.

1. példa:

Az MB_JOG1 szimbólum a jog gombok mátrixában a bal felső gombra hivatkozik. A konkrét gépen ez a gomb az X tengelyt negatív irányban mozgatja.

Ha a gombra B_JOG_XN szimbólummal akarunk hivatkozni, akkor a PLC Editorban a szimbólumot az MB_JOG1 szimbólumhoz, mint bázishoz képesti referencia hivatkozással vegyük fel, nulla eltolással – és semmiképp ne az MB_JOG1 szimbólum számszerű címét írjuk be a B_JOG_XN szimbólumra.

2. példa:

Az INP000 duplaszó 17-es bitjére MGZ_RPT szimbólummal (magazin a refpontkapcsolón) akarunk hivatkozni.

Az MGZ_RPT szimbólumot az INP000 szimbólumhoz, mint bázishoz való 0 eltolású referenciahivatkozással vegyük fel és a bitszámra adjunk 17-et.

7.1 NCT gépi kezelőpanelek

A rendszer az

MK19: 19 hüvelykes

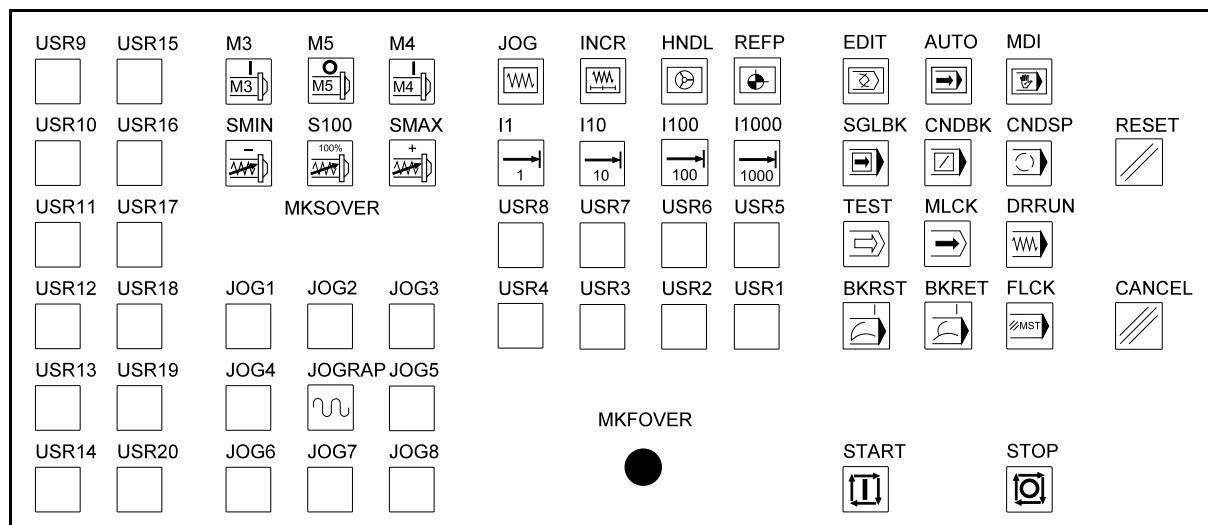
MK15: 15 hüvelykes

monitor alá szerelhető gépi kezelőpanel jeleit fogadja EtherCAT hálózaton keresztül.

A gépi kezelőpanel nyomógombkiosztása az alábbi ábrán látható. Minden egyes nyomógombhoz lámpa is tartozik. Minden nyomógomb fölött feltüntettük a gomb és lámpa szimbólumának közös szövegrészét is. A kezelőpanelhez tartoznak szabad felhasználású nyomógomb bemenetek és lámpa kimenetek. Ezek tetszőleges helyre köthetőek, az ábra nem tartalmazza ezeket.

Az orsó override gombok alatti MKSOVER, illetve az előtolás override fölötti MKFOVER, a megfelelő override érték átadó regiszterének szimbóluma.

Figyelem! Az MK15 kezelőpanel az USR9-től USR20-ig terjedő nyomógombokat nem tartalmazza!



A gépi kezelőpanel gombjainak és lámpáinak itt következő szimbólumai mind **bites** hivatkozású szimbólumok.

A kezelőpanel bites változói:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
MB_START	Start gomb	ML_START	Start lámpa
MB_STOP	Stop gomb	ML_STOP	Stop lámpa
MB_FLCK	Funkció zárva gomb	ML_FLCK	Funkció zárva lámpa
MB_INP1	Gépi kezelőpanel általános bemenet 1	ML_OUT1	Gépi kezelőpanel általános kimenet 1

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
MB_M3	M3 gomb	ML_M3	M3 lámpa
MB_M4	M4 gomb	ML_M4	M4 lámpa
MB_M5	M5 gomb	ML_M5	M5 lámpa
MB_INP2	Gépi kezelőpanel általános bemenet 2	ML_OUT2	Gépi kezelőpanel általános kimenet 2
MB_JOG1	Jog1 gomb	ML_JOG1	Jog1 lámpa
MB_JOG2	Jog2 gomb	ML_JOG2	Jog2 lámpa
MB_JOG3	Jog3 gomb	ML_JOG3	Jog3 lámpa
MB_JOG4	Jog4 gomb	ML_JOG4	Jog4 lámpa
MB_JOG5	Jog5 gomb	ML_JOG5	Jog5 lámpa
MB_JOG6	Jog6 gomb	ML_JOG6	Jog6 lámpa
MB_JOG7	Jog7 gomb	ML_JOG7	Jog7 lámpa
MB_JOG8	Jog8 gomb	ML_JOG8	Jog8 lámpa
MB_REFP	Refpont üzemmód gomb	ML_REFP	Refpont üzemmód lámpa
MB_HNDL	Kézikerék üzemmód gomb	ML_HNDL	Kézikerék üzemmód lámpa
MB_INCR	Léptetés üzemmód gomb	ML_INCR	Léptetés üzemmód lámpa
MB_JOG	Mozgatás üzemmód gomb	ML_JOG	Mozgatás üzemmód lámpa
MB_B20	Nem használt	ML_B20	Nem használt
MB_MDI	Kézi adatbevitel üzemmód gomb	ML_MDI	Kézi adatbevitel üzemmód lámpa
MB_AUTO	Automata üzemmód gomb	ML_AUTO	Automata üzemmód lámpa
MB_EDIT	Szerkesztés üzemmód gomb	ML_EDIT	Szerkesztés üzemmód lámpa
MB_TEST	Teszt gomb	ML_TEST	Teszt lámpa
MB_MLCK	Gép zárva gomb	ML_MLCK	Gép zárva lámpa
MB_DRRUN	Száraz futás gomb	ML_DRRUN	Szárazfutás lámpa
MB_BKRST	Mondat újra gomb	ML_BKRST	Mondat újra lámpa
MB_BKRET	Mondat vissza gomb	ML_BKRET	Mondat vissza lámpa
MB_CNDSP	Feltételes stop gomb	ML_CNDSP	Feltételes stop lámpa
MB_CNDBK	Feltételes mondat gomb	ML_CNDBK	Feltételes mondat lámpa
MB_SGLBK	Mondatonként gomb	ML_SGLBK	Mondatonként üzemmód lámpa
MB_I1	1 inkremens gomb	ML_I1	1 inkremens lámpa
MB_I10	10 inkremens gomb	ML_I10	10 inkremens lámpa
MB_I100	100 inkremens gomb	ML_I100	100 inkremens lámpa
MB_I1000	1000 inkremens gomb	ML_I1000	1000 inkremens lámpa
MB_SMAX	S+% gomb	ML_SMAX	S+% gomb lámpa
MB_S100	S100% gomb	ML_S100	S100% gomb lámpa
MB_SMIN	S-% gomb	ML_SMIN	S-% gomb lámpa

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
MB_JOGRAP	Jog gyorsmenet gomb	ML_JOGRAP	Jog gyorsmenet lámpa
MB_USR1	PLC-ben definiált gomb 1	ML_USR1	PLC-ben definiált lámpa 1
MB_USR2	PLC-ben definiált gomb 2	ML_USR2	PLC-ben definiált lámpa 2
MB_USR3	PLC-ben definiált gomb 3	ML_USR3	PLC-ben definiált lámpa 3
MB_USR4	PLC-ben definiált gomb 4	ML_USR4	PLC-ben definiált lámpa 4
MB_USR5	PLC-ben definiált gomb 5	ML_USR5	PLC-ben definiált lámpa 5
MB_USR6	PLC-ben definiált gomb 6	ML_USR6	PLC-ben definiált lámpa 6
MB_USR7	PLC-ben definiált gomb 7	ML_USR7	PLC-ben definiált lámpa 7
MB_USR8	PLC-ben definiált gomb 8	ML_USR8	PLC-ben definiált lámpa 8
MB_USR9	PLC-ben definiált gomb 9	ML_USR9	PLC-ben definiált lámpa 9
MB_USR10	PLC-ben definiált gomb 10	ML_USR10	PLC-ben definiált lámpa 10
MB_USR11	PLC-ben definiált gomb 11	ML_USR11	PLC-ben definiált lámpa 11
MB_USR12	PLC-ben definiált gomb 12	ML_USR12	PLC-ben definiált lámpa 12
MB_USR13	PLC-ben definiált gomb 13	ML_USR13	PLC-ben definiált lámpa 13
MB_USR14	PLC-ben definiált gomb 14	ML_USR14	PLC-ben definiált lámpa 14
MB_B23	Nem használt	ML_RESET	Reset gomb lámpa
MB_B24	Nem használt	ML_CANCEL	Cancel gomb lámpa
MB_USR15	PLC-ben definiált gomb 15	ML_USR15	PLC-ben definiált lámpa 15
MB_USR16	PLC-ben definiált gomb 16	ML_USR16	PLC-ben definiált lámpa 16
MB_USR17	PLC-ben definiált gomb 17	ML_USR17	PLC-ben definiált lámpa 17
MB_USR18	PLC-ben definiált gomb 18	ML_USR18	PLC-ben definiált lámpa 18
MB_USR19	PLC-ben definiált gomb 19	ML_USR19	PLC-ben definiált lámpa 19
MB_USR20	PLC-ben definiált gomb 20	ML_USR20	PLC-ben definiált lámpa 20
MB_USR21	PLC-ben definiált gomb 21	ML_USR21	PLC-ben definiált lámpa 21
MB_USR22	PLC-ben definiált gomb 22	ML_USR22	PLC-ben definiált lámpa 22
NB_RESET	Reset gomb		
NB_CANCEL	Cancel gomb		
NB_NCPC	Nem használt		

A kezelőpanel 2 DWORD regiszteren keresztül átadja az előtolás és az orsó override kapcsolók állását is.

A kezelőpanelhez duplaszavas változói:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
MKBTNS0	Gépi kezelőpanel alsó 32 gombja (DWORD)	MKLEDS0	Gépi kezelőpanel alsó 32 gomb lámpái (DWORD)
MKBTNS1	Gépi kezelőpanel felső 32 gombja (DWORD)	MKLEDS1	Gépi kezelőpanel felső 32 gomb lámpái (DWORD)
NCTBTNS	Gombok (DWORD)		
MKFOVER	Előtolás override kapcsoló állása: 0, 1, 2...		
MKSOVER	Oorsó override értéke: 0...10 (DWORD)		

A kezelőpanelek címzése

A vezérlőhöz maximum 4 gépi kezelőpanel csatlakoztatható.

A gépi kezelőpanel címének beállítását a vezérlő **Szervíz menüjében az ECAT beállítások** ablak beváltása után végezhetjük el. Válasszuk ki a bal oldali panelen a megfelelő egységet és a **Setting** fület választva állítsuk be az egység **címét**.

A címek értékei:

1, 2, 3, 4

lehetnek.

Az 1-es kezelőpanel címei közvetlenül a szimbólumra való hivatkozással érhetők el. Például:

MB_START

az 1. kezelőpanel START gombjára hivatkozik.

A többi (2., 3., 4.) kezelőpanel szimbólumai indexelt hivatkozással érhetők el. Például:

MB_START,#2

a 3. (2-es indexű) kezelőpanel START gombjára hivatkozik.

PLC-ben definiált gombok

A PLC-ben definiált gomboknak és lámpáknak a PLC programozója adhat funkciót. Ugyanez vonatkozik a JOG1...JOG8 gombokra és lámpákra. Ezekre a PLC programozója definiálhat egyedi szimbólumot annak megfelelően, hogy az egyes gombok mely tengelyeket mozgatják.. Ezeket a bites szimbólumokat mindig a megfelelő NC szimbólumhoz, mint bázishoz (pl. MB_JOG1) kell megadni 0 eltolással.

Példa:

Az MB_JOG1 szimbólum a jog gombok mátrixában a bal felső gombra hivatkozik. A konkrét gépen ez a gomb az X tengelyt negatív irányban mozgatja.

Ha a gombra B_JOG_XN szimbólummal akarunk hivatkozni, akkor a PLC Editorban a szimbólumot az MB_JOG1 szimbólumhoz, mint bázishoz képesti referencia hivatkozással vegyük

fel, nulla eltolással – és semmiképp ne az MB_JOG1 szimbólum számszerű címét írjuk be a B_JOG_XN szimbólumra.

Gombok, amelyeknek kimenetük és bemenetük is van a csatorna felé

Az alábbi üzemmódváltó, működési feltételeket beállító nyomógomboknak és a start, stop gombnak csatornánként indexelt kimenetei vannak az NC felé CP_ előtaggal és az NC-től csatornánként indexelt bemenetei vannak CN_ előtaggal a lámpák számára:

MB_JOG	– CP_JOG	CN_JOG	– ML_JOG
MB_INCR	– CP_INCR	CN_INCR	– ML_INCR
MB_HNDL	– CP_HNDL	CN_HNDL	– ML_HNDL
MB_REFP	– CP_REFP	CN_REFP	– ML_REFP
MB_EDIT	– CP_EDIT	CN_EDIT	– ML_EDIT
MB_AUTO	– CP_AUTO	CN_AUTO	– ML_AUTO
MB_MDI	– CP_MDI	CN_MDI	– ML_MDI
MB_TEST	– CP_TEST	CN_TEST	– ML_TEST
MB_MLCK	– CP_MLCK	CN_MLCK	– ML_MLCK
MB_DRRUN	– CP_DRRUN	CN_DRRUN	– ML_DRRUN
MB_BKRST	– CP_BKRST	CN_BKRST	– ML_BKRST
MB_BKRET	– CP_BKRET	CN_BKRET	– ML_BKRET
MB_FLCK	– CP_FLCK	CN_FLCK	– ML_FLCK
MB_START	– CP_START	CN_START	– ML_START
MB_STOP	– CP_STOP	CN_STOP	– ML_STOP

A PLC programnak meg kell vizsgálnia, hogy a nyomógomb megnyomása a gép oldaláról engedélyezett-e. Például MB_START gomb megnyomásakor be van-e a gép kapcsolva és a munkatér zárt-e.

Ha a gép oldaláról nincs akadály a PLC program a CP csatorna kezelő jelzőn keresztül kéri a kívánt hatást az NC-től. Az NC is megvizsgálja, hogy a gomb megnyomása jogos-e. Például CP_START=1 esetén futtatható-e program az adott üzemmódban, stb.

Ha az NC elfogadta a gombnyomást, a megfelelő CN csatorna kezelő jelzőn keresztül jelzi azt a PLC-nek. Például a CN_START=1 állapottal.

Ezután a PLC már feltétel nélkül bekapcsolhatja a gombhoz tartozó lámpát. Példánkánál maradva ML_START=1.

Gombok, amelyeknek csak kimenetük van a csatorna felé

Az alábbi feltétel kapcsoló nyomógomboknak csatornánként indexelt kimenetei vannak az NC felé CP_ előtaggal, de nem tartozik hozzájuk visszajelzés az NC oldaltól (CN jelző):

MB_JOGRAP	– CP_JOGRAP	– ML_JOGRAP
MB_SGLBK	– CP_SGLBK	– ML_SGLBK

MB_CNDBK	– CP_CNDBK	– ML_CNDBK
MB_CNDSP	– CP_CNDSP	– ML_CNDSP

Ezeknél a feltételkapcsolóknál a nyomógomb lámpáját a PLC programban eltárolt állapot alapján kell kezelni, vagy egyszerűen a gomb állapotát átmásolni a lámpára.

Gombok, amelyeknek kimenetük van a tengelykezelő felé

A jog gombokat a mozgatni kívánt tengelynek a tengelyenként indexelt, AP_ előtagú bemenetére kell kötni:

MB_JOGn – AP_JOGP (+irány), vagy AP_JOGN (- irány) – ML_JOGn

A jog gombok lámpáit közvetlen a gombra lehet kötni.

Az inkremensválasztó gombok kezelése

Az inkremensválasztó gombok parancsaiból kiszámolt lépés nagyságát a vezérlő csatornáként indexelt, lebegőpontos regisztereibe kell írni, lebegőpontos formában.

Például: *0.001, *0.01, *0.1, *1.

Figyelembe kell venni, a CN_INCH jelző alapján, hogy a vezérlőt metrikus, vagy inches adat-beadással használják, és ha kell, konvertálni a kimeneti mértékrendszerre az adatot. Kimeneti mértékrendszer: N0104 Unit of Measure paraméter #0 IND bit.

MB_I1, MB_I10, MB_I100, MB_I1000 – CP_INC

Az előtolás és gyorsmenet override kezelése

Az előtolás override kapcsoló MKFOVER (DWORD) állásából kiszámolt override értéket az NC csatornáként indexelt, lebegőpontos regisztereibe (double) kell írni, lebegőpontos formában.

100%-nak *1.0 felel meg mindkét override esetén.

Az MKFOVER regiszterben kapott érték változhat az override kapcsoló hardver kialakításának függvényében.

A gyorsmenet override bemenetet is köthetjük az előtolás override kapcsolóra.

MKFOVER – CP_FOVER, CP_ROVER

Az orsó override kezelése

Az orsó override-ot az MB_SMAX, MB_S100, MB_SMIN gombokról kezeljük.

A PLC program vagy ezeknek a gomboknak a megnyomására, vagy az MKSOVER regiszter adatainak felhasználásával kiszámol egy lebegőpontos override értéket.

MKSOVER: DWORD, 0-tól 10-ig terjedő értéket ad át a PLC-nek.

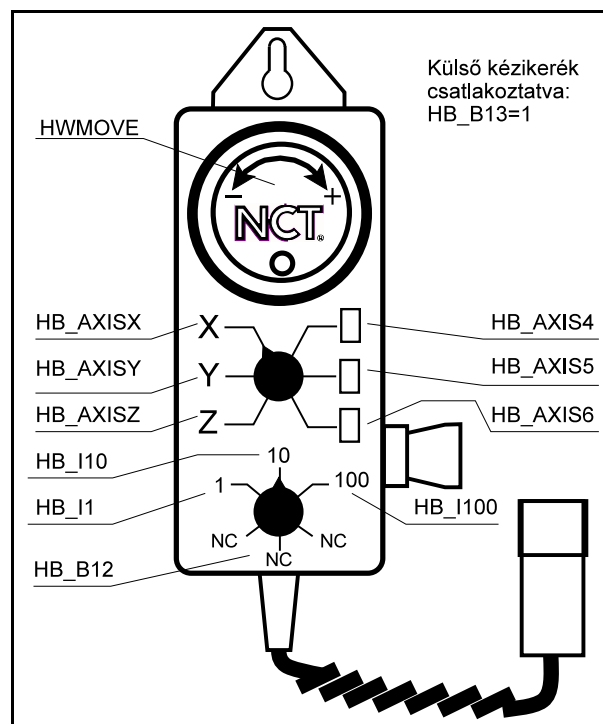
Az orsó override értékek számára orsónként indexelt, SP előtagú, lebegőpontos regiszterek állnak rendelkezésre. 100%-nak *1.0 felel meg:

MB_SMAX, MB_S100, MB_SMIN vagy MKSOVER – SP_SOVER

7.2 NCT kézikerek

Előlapra szerelt kézikerek esetén a tengely és inkremensválasztás a kezelőpanel gombjaival történik.

Külső kézikereknek nevezzük azt az egységet, amely külön dobozban van, és a dobozon tengely és inkremensválasztó forgókapcsolók vannak. A külső kézikerek forgókapcsolóinak állásaihoz beépített bites szimbólumok tartoznak.



A külső kézikerek bites változói a következők:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
HB_AXISX	X tengely bemenet (külső kzk.)	HL_AXISX	Nem használt
HB_AXISY	Y tengely bemenet (külső kzk.)	HL_AXISY	Nem használt
HB_AXISZ	Z tengely bemenet (külső kzk.)	HL_AXISZ	Nem használt
HB_AXIS4	4. tengely bemenet (külső kzk.)	HL_AXIS4	Nem használt
HB_AXIS5	5. tengely bemenet (külső kzk.)	HL_AXIS5	Nem használt
HB_AXIS6	6. tengely bemenet (külső kzk.)	HL_AXIS6	Nem használt
HB_AXIS7	7. tengely bemenet (külső kzk.)	HL_AXIS7	Nem használt
HB_AXIS8	8. tengely bemenet (külső kzk.)	HL_AXIS8	Nem használt
HB_I1	1 inkremens bemenet (külső kzk.)	HL_I1	Nem használt
HB_I10	10 inkremens bemenet (külső kzk.)	HL_I10	Nem használt
HB_I100	100 inkremens bemenet (külső kzk.)	HL_I100	Nem használt
HB_I1000	1000 inkremens bemenet (külső kzk.)	HL_I1000	Nem használt

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
HB_B12	Nem használt	HL_B12	Nem használt
HB_B13	Nem használt	HL_B13	Nem használt
HB_B14	Nem használt	HL_B14	Nem használt
HB_B15	Nem használt	HL_B15	Nem használt

A fenti ábrán látható külső kézikerék esetén a az alábbi szimbólumok vannak még bekötve:

HB_B12: az inkremensválasztó kapcsoló NC állása

HB_B13: a jelző 1 állapota jelenti, hogy **külső kézikerék** van a vezérlőhöz **csatolva**.

A kézikerék duplaszavas változói a következők:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
HWMOVE	Kézikerék: ciklusonkénti elmozdulás (csak Int0 modulban!) (DWORD)		
HWBITS	Külső kézikerék bemenetek (DWORD)	HWLEDS	Nem használt
		P_H1AS	1. Kézikerékhez rendelt tengely száma (=0 inaktív) (DWORD)
		P_H2AS	2. Kézikerékhez rendelt tengely száma (=0 inaktív) (DWORD)
		P_H3AS	3. Kézikerékhez rendelt tengely száma (=0 inaktív) (DWORD)
		P_H4AS	4. Kézikerékhez rendelt tengely száma (=0 inaktív) (DWORD)

HWMOVE: a regiszterből olvasható ki a kézikerék előző ciklushoz képesti elmozdulása.

Mivel a regiszter frissítése a TimeSlice ciklusidő gyakoriságával történik, ezért a regisztert a PLC program Int0 moduljában kell kiolvasni. A regiszter beépített és külső kézikerék esetén is olvasható.

P_HnAS: az 1-4 kézikerékhez rendelt regiszterbe annak a tengelynek a számát kell beírni, amelyiket az adott kézikerékkel mozgatni akadjuk. Ha a regiszter értéke 0, a kézikerék inaktív.

A kézikerek címzése

A vezérlőhöz maximum 4 kézikérék csatlakoztatható.

A kézikérék címének beállítását a vezérlő **Szerviz menü**jében az **ECAT beállítások** ablak be-
váltása után a **gépi kezelőpanel** beállításainál végezhetjük el. Válasszuk ki a bal oldali panelen
a megfelelő egységet és a **Setting** fület választva állítsuk be a kézikérék **címét**.

A címek értékei:

1, 2, 3, 4

lehetnek.

A beépített kézikérék címe mindig 4.

A külső kézikérék címe alapkivitelben (gyári beállítás): 4.

☞ **Figyelem!** Mivel alapkivitelben a 4. kézikereket használjuk, ne feledkezzünk meg a szimbó-
lumok indexeléséről (symbol,#3)!

7.3 Kétállapotú, 24 V-os interfész be-, kimenetek

A vezérlőhöz az alábbi NCT gyártmányú, EtherCAT-es, kétállapotú, 24 V-os interfész be-, kimeneti egységek csatlakoztathatók:

I16: 16 bites 24 V-os interfész bemenet,

I16S: 3x16 bites 24 V-os interfész bemenet, minden bemeneti ponthoz 24V-os és 0 V-os tápfeszültség bekötőpont tartozik (érzékelők számára),

I32: 32 bites 24 V-os interfész bemenet,

O16: 16 bites 24 V-os interfész tranzisztoros kimenettel,

O8RM: 8 bites 24 V-os interfész Morse érintkezős relékimenettel

O8R: 8 bites 24 V-os interfész záróérintkezős relékimenettel

A kétállapotú, 24 V-os interfész be-, kimenetek *NC szimbólumai 32 bites (DWORD)* hivatkozások.

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
INP000	Interfész bemenetek 0. DWORD	OUT000	Interfész kimenetek 0. DWORD
INP001	Interfész bemenetek 1. DWORD	OUT010	Interfész kimenetek 1. DWORD
INP010	Interfész bemenetek 2. DWORD	OUT020	Interfész kimenetek 2. DWORD
INP011	Interfész bemenetek 3. DWORD	OUT030	Interfész kimenetek 3. DWORD
INP020	Interfész bemenetek 4. DWORD	OUT040	Interfész kimenetek 4. DWORD
INP021	Interfész bemenetek 5. DWORD	OUT050	Interfész kimenetek 5. DWORD
INP030	Interfész bemenetek 6. DWORD	OUT060	Interfész kimenetek 6. DWORD
INP031	Interfész bemenetek 7. DWORD	OUT070	Interfész kimenetek 7. DWORD
INP040	Interfész bemenetek 8. DWORD		
INP041	Interfész bemenetek 9. DWORD		
INP050	Interfész bemenetek 10. DWORD		
INP051	Interfész bemenetek 11. DWORD		
INP060	Interfész bemenetek 12. DWORD		
INP061	Interfész bemenetek 13. DWORD		
INP070	Interfész bemenetek 14. DWORD		
INP071	Interfész bemenetek 15. DWORD		

Az I/O kártyák címezése

A vezérlőhöz csatolt *be-, kimeneti hardver egységek* be-, és kimeneteinek száma a byte (8 bit) egész számú többszöröse.

A be-, kimeneti egységek kezdőcímének beállítását a vezérlő **Szerviz menüjében** az **ECAT beállítások** ablak beváltása után végezhetjük el. Válasszuk ki a bal oldali panelen a megfelelő egységet és a **Setting** fület választva állítsuk be az egység **kezdőcímét**.

A címzés byte-onként történik. Az egységek címei külön a be-, és kimeneti egységekre 1-től 64-ig terjedhetnek. 8 bites egységek címfoglalása 1 cím, 16 bitesé 2 cím stb.

Az 1-es cím

bemeneti egység esetén az INP000 szó 0. byte-ja

kimeneti egység esetén az OUT000 szó 0. byte-ja,

azaz 00...07 bitjei.

A 2-es cím a

0. DWORD 1-es byte, azaz 08...15 bitjei és így tovább.

A címek sorrendjének nem kell követnie az EtherCAT láncra való felfűzés fizikai sorrendjét.

Az alábbi táblázat egy egyszerű mintát közöl a kezdőcímek kitöltésére és a jelek memóriában történő helyére.

Hardver egység	Kezdőcím	Hivatkozás a jelre
8 bites bemeneti egység	1	INP000: 00...07 bit
16 bites bemeneti egység	2	INP000: 08...23 bit
16 bites bemeneti egység	4	INP000: 24...31, INP001: 00...07 bit
8 bites bemeneti egység	6	INP001: 08...15 bit
16 bites kimeneti egység	1	OUT000: 00...15 bit
8 bites kimeneti egység	3	OUT000: 16...23 bit
16 bites kimeneti egység	4	OUT000: 24...31, OUT010: 00...07 bit
8 bites kimeneti egység	6	OUT010: 08...15 bit

Az egyedi, bites ki-, bemenetek szimbólumait a PLC programozó maga határozhatja meg. Ezeket a bites szimbólumokat mindig a megfelelő NC szimbólumhoz, mint bázishoz (pl. INP000) kell megadni. A duplaszón belül a bit kiválasztása (00, ..., 31) számszerűen történik az I/O kiosztásnak megfelelően.

Példa:

Az INP000 duplaszó 17-es bitjére MGZ_RPT szimbólummal (magazin a refpontkapcsolón) akarunk hivatkozni.

Az MGZ_RPT szimbólumot az INP000 szimbólumhoz, mint bázishoz való 0 eltolású referenciavivatkozással vegyük fel és a bitszámra adjunk 17-et.

7.4 NCT tapintó illesztő kártyák be-, kimenetei

A vezérlőhöz EtherCAT buszon keresztül tapintók jeleinek fogadására alkalmas kártyákat lehet kötni. Ezek típusa:

ETPC: 2 csatornás tapintó kezelő kártya. A kártyán a tapintó nyomva jelen kívül tapintónként rendelkezésre áll még 3 db 24 Voltos bemenet és 2 db. 24 Voltos kimenet.

A vezérlő 8 tapintó jeleit tudja kezelni.

A tapintó illesztő kártyák bites be-, kimenetei:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
TN_TS1	Tapintó 1 nyomva van	TP_OUT11	Tapintó 1 kimenete 1
TN_INP11	Tapintó 1 bemeneti jele 1	TP_OUT12	Tapintó 1 kimenete 2
TN_INP12	Tapintó 1 bemeneti jele 2	TP_OUT21	Tapintó 2 kimenete 1
TN_INP13	Tapintó 1 bemeneti jele 3	TP_OUT22	Tapintó 2 kimenete 2
TN_TS2	Tapintó 2 nyomva van	TP_OUT31	Tapintó 3 kimenete 1
TN_INP21	Tapintó 2 bemeneti jele 1	TP_OUT32	Tapintó 3 kimenete 2
TN_INP22	Tapintó 2 bemeneti jele 2	TP_OUT41	Tapintó 4 kimenete 1
TN_INP23	Tapintó 2 bemeneti jele 3	TP_OUT42	Tapintó 4 kimenete 2
TN_TS3	Tapintó 3 nyomva van	TP_OUT51	Tapintó 5 kimenete 1
TN_INP31	Tapintó 3 bemeneti jele 1	TP_OUT52	Tapintó 5 kimenete 2
TN_INP32	Tapintó 3 bemeneti jele 2	TP_OUT61	Tapintó 6 kimenete 1
TN_INP33	Tapintó 3 bemeneti jele 3	TP_OUT62	Tapintó 6 kimenete 2
TN_TS4	Tapintó 4 nyomva van	TP_OUT71	Tapintó 7 kimenete 1
TN_INP41	Tapintó 4 bemeneti jele 1	TP_OUT72	Tapintó 7 kimenete 2
TN_INP42	Tapintó 4 bemeneti jele 2	TP_OUT81	Tapintó 8 kimenete 1
TN_INP43	Tapintó 4 bemeneti jele 3	TP_OUT82	Tapintó 8 kimenete 2
TN_TS5	Tapintó 5 nyomva van		
TN_INP51	Tapintó 5 bemeneti jele 1		
TN_INP52	Tapintó 5 bemeneti jele 2		
TN_INP53	Tapintó 5 bemeneti jele 3		
TN_TS6	Tapintó 6 nyomva van		
TN_INP61	Tapintó 6 bemeneti jele 1		
TN_INP62	Tapintó 6 bemeneti jele 2		
TN_INP63	Tapintó 6 bemeneti jele 3		
TN_TS7	Tapintó 7 nyomva van		
TN_INP71	Tapintó 7 bemeneti jele 1		

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
TN_INP72	Tapintó 7 bemeneti jele 2		
TN_INP73	Tapintó 7 bemeneti jele 3		
TN_TS8	Tapintó 8 nyomva van		
TN_INP81	Tapintó 8 bemeneti jele 1		
TN_INP82	Tapintó 8 bemeneti jele 2		
TN_INP83	Tapintó 8 bemeneti jele 3		

A tapintó illesztő címzése

A vezérlő maximum 8 tapintó jelét kezeli.

Egy tapintó illesztő kártya 2 tapintó be-, kimenetet illeszt.

Az illesztőkártya címeinek beállítását a vezérlő **Szerviz menüjében az ECAT beállítások** ablak beváltása után a **tapintó illesztő** beállításainál végezhetjük el. Válasszuk ki a bal oldali panelen a megfelelő egységet és a **Setting** fület választva állítsuk be a kártyán használt két tapintó be-, kimenet **címeit**.

A címek értékei:

Not used, 1, 2, ..., 8

lehetnek. A címek értékei a tapintó be-, kimeneteinek szimbólum indexeinek felelnek meg.

A tapintó illesztő bemenetei

TN_TS_n: Tapintó n nyomva van

Ez a bemenet jelzi, hogy az n című tapintó szára kitért, vagy a tapintó gombja nyomva van.

A tapintó nyomva jel kötelezően aktív 0! Az illesztőkártyát úgy kell beállítani, hogy ez a feltétel teljesüljön.

TN_INP_{n1}: Tapintó n bemeneti jele 1

TN_INP_{n2}: Tapintó n bemeneti jele 2

TN_INP_{n3}: Tapintó n bemeneti jele 3

Az n című tapintó 3 db opcionális 24 Voltos bemenete.

A bemenetek felhasználhatók pl. a tapintó üzembesz állapotának, vagy az elem töltöttségének jelzésére.

A tapintó illesztő kimenetei

TP_OUT_{n1}: Tapintó n kimenete 1

TP_OUT_{n2}: Tapintó n kimenete 2

Az n című tapintó 2 db opcionális 24 Voltos kimenete.

A kimenetek például a tapintó be-, kikapcsolására. használhatók.

7.5 NCT érzékelő bemenetek

Az EtherCAT hálózatra köthető, érzékelő bemeneteket tartalmazó kártya típusa:

SENS: A kártya 8 db. KTY84/130 hőmérő analóg bemenetet és 1 db. 12 bites 4-20 mA A-D átalakítót tartalmaz. A hőmérő bemenetekre komparálási érték állítható be.

Az érzékelő bites bemenetei

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
IN_1EN0	0. analóg bemenet > komparálási érték		
IN_1EN1	1. analóg bemenet > komparálási érték		
IN_1EN2	2. analóg bemenet > komparálási érték		
IN_1EN3	3. analóg bemenet > komparálási érték		
IN_1EN4	4. analóg bemenet > komparálási érték		
IN_1EN5	5. analóg bemenet > komparálási érték		
IN_1EN6	6. analóg bemenet > komparálási érték		
IN_1EN7	7. analóg bemenet > komparálási érték		
IN_1EN8	Nem használt		

Az érzékelő duplaszavas bemenetei

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
ANINPUTS	Analóg bemenetek: 32db DWORD		
IN_1	Analóg komparátor bemenetek státusza (DWORD)		

A SENS kártya címzése és a komparálási érték beállítása

A vezérlő maximum 32 SENS kártya jelét kezeli.

A SENS kártya címeinek beállítását a vezérlő **Szerviz menüjében** az **ECAT beállítások** ablak beváltása után a **kártya** beállításainál végezhetjük el. Válasszuk ki a bal oldali panelen a megfelelő egységet és a **Setting** fület választva állítsuk be a kártya címét.

A címek értékei:

Not used, 1, 2, ..., 32

lehetnek. Az IN_1ENn, IN_1 és ANINPUTS szimbólumok az 1. (0. indexű) kártyára vonatkoznak, a többi kártya jelei indexelt hivatkozással érhetők el.

A komparálási értékeket fokban, csatornánként az **ECAT beállítások** ablak beváltása után a kártya beállításainál végezhetjük el.

Az érzékelő bites bemenetei:

IN_1ENn: n. analóg bemenet > komparálási érték

Ha a kártya n. bemenetén az analóg jel nagysága meghaladja a beállított komparálási értéket a jel 1-be megy.

Az érzékelő duplaszavas bemenetei

ANINPUTS: Analóg bemenetek: 32db DWORD

Az ANINPUTS változón, megfelelő indexeléssel, a kártya analóg bemenetének értéke olvasható ki.

IN_1: Analóg komparátor bemenetek státusza (DWORD)

A változóról duplaszavas formában érhetők el a komparátor bemenetek.

7.6 NCT analóg bemenetek

Az EtherCAT hálózatra köthető, analóg bemeneteket tartalmazó kártya típusa:

DANI: A kártya 6 db. 12 bites analóg digitál átalakítót tartalmaz. Az analóg bemenetek konfigurálhatóak +/-10 V-ra, vagy 0-20 mA-re.

Az analóg jelek duplaszavas bemenetei

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
ANINPUTS	Analóg bemenetek: 32db DWORD		

A DANI kártya címzése

A vezérlő maximum 32 analóg bemenet jelét kezeli.

A DANI kártya címeinek beállítását a vezérlő *Szerviz menü*jében az *ECAT beállítások* ablak beváltása után a *kártya* beállításainál végezhetjük el. Válasszuk ki a bal oldali panelen a megfelelő egységet és a *Setting* fület választva állítsuk be az analóg bemenetek címeit egyenként.

A címek értékei:

Not used, 1, 2, ..., 32

lehetnek. Az ANINPUTS szimbólum az 1. (0. indexű) analóg bemenetre vonatkozik, a többi bemenet jelei indexelt hivatkozással érhetők el.

Az analóg jelek duplaszavas bemenetei

ANINPUTS: Analóg bemenetek: 32db DWORD

Az ANINPUTS változón, megfelelő indexeléssel, a kártya megfelelő analóg bemenetének értéke olvasható ki.

7.7 Az EtherCAT-es NCT hajtások be-, kimenetei

A vezérlőhöz EtherCAT buszon keresztül a következő típusú NCT hajtásokat lehet kötni:

DS-i/I EE: szinkron szervomotorokhoz,

DA-i/I EE: aszinkron motorokhoz,

ahol: i: névleges áram

I: maximális áram

EE: EtherCAT, EnDat 2.2 interfész.

A rendszer maximum 48 db. EtherCAT-es hajtás jeleit tudja kezelni (32 tengely + 16 orsó).

Az EtherCAT-es hajtások átadják az NC-nek az EtherCAT hálózaton a motorra szerelt jeladóról mért pozíciót, motoráramot, fordulatszámot stb, a státuszukat, illetve a hibaállapotuk kódjait. Ugyanígy fogadják az NC-től a sebességalapjelet, illetve az NC-től jövő egyéb parancsokat.

A fenti típusú NCT EtherCAT-es hajtások jelei a következők:

A hajtások bites státusz és kontrol jelei

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
DN_ENA	Hajtás engedélyezve	DP_ENA1	Hajtás engedélyezés 1 bit
DN_RDY	Hajtás üzemkész	DP_ENA2	Hajtás engedélyezés 2 bit
DN_INC	Inkrementális jeladó a hajtáson	DP_EMG	Nincs vészfékezés
DN_PRM1	Aktív paramétertábla 1bit	DP_POSLCK	Hajtás pozícióban tartás be
DN_PRM2	Aktív paramétertábla 2bit	DP_PRM1	Hajtás paramtábla választás 1 bit
		DP_PRM2	Hajtás paramtábla választás 2 bit
		DP_MOD1	Hajtás szabályozó mód választás 1 bit
		DP_MOD2	Hajtás szabályozó mód választás 2 bit
		DP_SILCK	Sebesség szabályozó integrátor befagyasztás
		DP_ERRCLR	Hajtás hiba törlés

A hajtások bites hibajelzői

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
DN_ERROR	Hajtás hiba van (DN_ERR>0)		
DN_EDERR1	Jeladó hiba		

7.7 Az EtherCAT-es NCT hajtások be-, kimenetei

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
DN_EDERR2	Jeladó hőmérséklete magas		
DN_IEERR1	Nem használt		
DN_IEERR2	Nem használt		
DN_BVERR	Túlfeszültség hiba a sínen		
DN_CURERR	Túl nagy a motoráram		
DN_CMEERR	Árammérési hiba		
DN_HALERR	Hall kommutáló jel hiba		
DN_HASERR	Hall szekvencia jel hiba		
DN_SRTERR	Időtúllépés hiba		
DN_CWDER1	CAN WatchDog hiba		
DN_CWDER2	Nem használt		
DN_CHERR1	CAN egyéb hiba		
DN_CHERR2	Nem használt		
DN_ECTERR	EtherCAT időtúllépés hiba		
DN_PDPINT	IGBT hiba		
DN_PRMERR	Hibás a paramétertábla		
DN_PRGERR	Hibás a hajtásprogram		
DN_FOLERR	Sebességkövetési hiba		
DN_OVHERR	Hővédelem		

A hajtások duplaszavas változói

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
DN_STAT	Hajtás státusz regiszter (DWORD)	DP_CTRL	Hajtás kontrol regiszter (DWORD)
DN_ERR	Hajtás hiba jelzők regisztere (DWORD)		

Az EtherCAT-es hajtások címezése

A vezérlő maximum 48 db. EtherCAT-es hajtás jeleit kezeli.

A hajtások címeinek beállítását a vezérlő **Szervíz menüjében** az **ECAT beállítások** ablak be-váltása után a **megfelelő hajtást kiválasztva** végezhetjük el. Válasszuk ki a bal oldali panelen a megfelelő egységet és a **Setting** fület választva állítsuk be a hajtás **címét**.

A címek értékei:

Not used, 1, 2, ..., 48

lehetnek.

A hajtások és a tengelyek címeinek összerendelése

A vezérlő maximum 32 tengelyt kezel. A tengelyek címei a vezérlőben:

1, 2, ..., 32

lehetnek.

Az alapjel kiadása:

Az N0502 Axis Output Type paraméteren válasszuk az EtherCAT-32 opciót.

Az N0503 Axis Output Address paraméterre írjuk be annak a hajtásnak a címét, amelyikkel az adott tengelyt mozgatni kívánjuk.

Példa:

Az

N0100 Axis Name1 A4=C

paraméter a 4. tengely címét C-nek határozza meg.

A C tengely hajtásának címe az EtherCAT beállítások között: 6.

Ekkor az N0503 Axis Output Address A4=6,

azaz a vezérlő 4. tengelye a 6. hajtásnak adja ki az alapjelet.

A jeladó jeleinek fogadása:

Az N0500 Axis Input Type paraméteren válasszuk az EtherCAT opciót.

Az N0501 Axis Input Address paraméterre írjuk be annak a hajtásnak a címét, amelyiktől a jeladó jeleit fogadni kívánjuk.

 **Figyelem!** Az N0503 Axis Output Address paraméter és az N0501 Axis Input Address paraméter lehet különböző!

Példa:

Az

N0100 Axis Name1 A4=C

paraméter a 4. tengely címét C-nek határozza meg.

A C tengely hajtásának címe az EtherCAT beállítások között: 6.

A jeladó jeleit a hajtás által kezelt jeladóról kívánjuk venni.

Ekkor az N0501 Axis Input Address A4=6,

azaz a vezérlő 4. tengelye a 6. hajtástól veszi a jeladó jeleit.

A hajtások és az orsók címeinek összerendelése

A vezérlő maximum 16 orsót kezel. Az orsók címei a vezérlőben:

1, 2, ..., 16

lehetnek.

Az alapjel kiadása:

Az N0602 Spindle Output Type paraméteren válasszuk az EtherCAT opciót.

Az N0603 Spindle Output Address paraméterre írjuk be annak a hajtásnak a címét, amelyikkel az adott orsót mozgatni kívánjuk.

Példa:

Az

N0605 Spindle Name2 S3=3

paraméter a 3. orsó címét S3-nak határozza meg.

Az S3 orsó hajtásának címe az EtherCAT beállítások között: 8.

Ekkor az N0603 Spindle Output Address S3=8,

azaz a vezérlő 3. orsója a 8. hajtásnak adja ki az alapjelet.

A jeladó jeleinek fogadása:

Az N0600 Spindle Input Type paraméteren válasszuk az EtherCAT opciót.

Az N0601 Spindle Input Address paraméterre írjuk be annak a hajtásnak a címét, amelyiktől a jeladó jeleit fogadni kívánjuk.

☞ **Figyelem!** Az N0603 Spindle Output Address paraméter és az N0601 Spindle Input Address paraméter lehet különböző!

Példa:

Az

N0605 Spindle Name2 S3=3

paraméter a 3. orsó címét S3-nak határozza meg.

Az S3 orsó hajtásának címe az EtherCAT beállítások között: 8.

A jeladó jeleit a hajtás által kezelt jeladóról kívánjuk venni.

Ekkor az N0601 Spindle Input Address S3=8,

azaz a vezérlő 3. orsója a 8. hajtástól veszi a jeladó jeleit.

Hivatkozás a PLC programban tengelyekre, orsókra és hajtásokra

Az

AN_ xx és AP_ xx

NC szimbólumok mindig tengelyekre vonatkoznak. Ezekre a szimbólumokra mindig tengely-index alapján kell hivatkozni:

#0: 1. tengely, ..., #31: 32. tengely.

Az

SN_ xx és SP_ xx

NC szimbólumok mindig orsókra vonatkoznak. Ezekre a szimbólumokra mindig orsóindex alapján kell hivatkozni:

#0: 1. orsó, ..., #15: 16. orsó.

A

DN_ xx és DP_ xx

NC szimbólumok mindig hajtásokra vonatkoznak. Ezekre a szimbólumokra mindig hajtásindex alapján kell hivatkozni:

#0: 1. hajtás, ..., #47: 48. hajtás.

Egy adott tengely, vagy orsó tengely-, illetve orsóindexe különböző lehet az adott tengely vagy orsó hajtásának indexétől!

A hajtások státuszjelei (bemenetek)**DN_ENA:** Hajtás engedélyezve

A DP_ENA1 és a DP_ENA2 jelek nyugtázó jelzője. Ha a jelző értéke 1, a hajtás működteti a motort, az feszültség alatt van.

DN_RDY: Hajtás üzemkész

Ha a jel értéke 1 a hajtás üzemkész, engedélyezhető.

DN_INC: Inkrementális jeladó a hajtáson

Ha a jelző DN_INC=0, a motoron abszolút, EnDAT jeladó van,
ha a jelző DN_INC=1, a motoron inkrementális jeladó van.

DN_PRM1: Aktív paramétertábla 1 bit**DN_PRM2:** Aktív paramétertábla 2 bit

A fenti két bit azt mondja meg, hogy a hajtás melyik paramétertáblából dolgozik. Lásd még: DP_PRM1 és a DP_PRM2 bitek leírását.

DN_PRM2	DN_PRM1	Aktív paramétertábla
0	0	1. paramétertábla aktív
0	1	2. paramétertábla aktív
1	0	3. paramétertábla aktív
1	1	4. paramétertábla aktív

A hajtások kontrol jelei (kimenetek)**DP_ENA1:** Hajtás engedélyezés 1 bit**DP_ENA2:** Hajtás engedélyezés 2 bit

A hajtás engedélyezése az alábbi bitváltásra történik meg

DP_ENA1 1 → 0, és

DP_ENA2 0 → 1.

DP_ENA2	DP_ENA1	Engedélyezés állapota
0	0	Nincs engedélyezve
0	1	Nincs engedélyezve
1	0	Engedélyezve van
1	1	Nincs engedélyezve

DP_EMG: Nincs vészfékezés

Ha DP_EMG=0, a hajtás, a bemenetén kapott alapjeltől függetlenül, leállítja a motort.

A leállítás addig tart, amíg a hajtás engedélyezve van.

DP_EMG=1 normál működés, a kapott alapjel szerint.

DP_POSLCK: Hajtás pozícióban tartás be

DP_POSLOCK=0, normál üzemmód,

DP_POSLOCK=1, a jel 0 → 1 átmenetkor a jeladóról mért pozícióban tartja a hajtás a motort a kapott alapjeltől függetlenül.

DP_PRM1: Hajtás paramtábla választás 1 bit

DP_PRM2: Hajtás paramtábla választás 2 bit

A fenti két bit azt mondja meg, hogy a hajtás melyik paramétertáblából dolgozzon. Paraméterváltás után meg kell várni, hogy a hajtás a DN_PRM1 és a DN_PRM2 biteken visszaadja a beállított bitmintát.

DP_PRM2	DP_PRM1	Paramétertáblaválasztása
0	0	1. paramétertábla választása
0	1	2. paramétertábla választása
1	0	3. paramétertábla választása
1	1	4. paramétertábla választása

DP_MOD1: Hajtás szabályozó mód választás 1 bit

DP_MOD2: Hajtás szabályozó mód választás 2 bit

A jelzők állításával az alábbi szabályozási módok közül lehet választani:

DP_MOD2	DP_MOD1	Szabályozási mód választása
0	0	Sebességszabályozó üzemmód
0	1	Áramszabályozó (nyomaték) üzemmód
1	0	Pozíciósabályozó üzemmód
1	1	Növelt pontosságú Sebességszabályozó üzemmód

DP_SILCK: Sebesség szabályozó integrátor befagyasztás

DP_SILCK=0, normál üzemmód,

DP_SILCK=1: a sebességszabályozó integrátorának kikapcsolása.

Példa:

Ellenorsós, rúdból dolgozó esztergán át kell venni a darabot az ellenorsóba. Az ellenorsót szinkronizáljuk a főorsóhoz, majd az ellenorsón zárjuk a tokmányt, hogy leszúrás után a darabot fogja az ellenorsó. Amíg a darab nincs leszúrva és mindkét tokmány fogja, addig az ellenorsó hajtásán ki kell a sebességszabályozó integrátorát kapcsolni, különben a rendszer túhatározottsága miatt lengés alakulhat ki.

DP_ERRCLR: Hajtás hiba törlés

Ha a hajtáson hiba van, azaz DN_ERR>0 (a hibaregiszter nem 0) a hibát törölni kell.

A DP_ERRCLR=1 állapot törli a hibát. A jelet addig 1-ben kell tartani, amíg a

DN_ERR hibaregiszter 0 nem lesz, vagy a DN_ERROR=0 állapot bekövetkezik.

A hajtások bites hibajelzői (bemenetek)

A PLC programnak nem kell a különböző hajtáshibákra üzenetet küldeni, ezt az NC megteszi. Csak a hiba törlését kell kiadni a DP_ERRCLR=1 jelző állítással.

DN_ERROR: Hajtás hiba van (DN_ERR>0)

DN_ERROR=1, ha a hibaregiszter valamelyik további bitje hibát jelez.

DN_EDERR1: Jeladó hiba

DN_EDERR1=1, ha a hajtás hibát észlel a motorra szerelt jeladóról.

DN_EDERR2: Jeladó hőmérséklete magas

DN_EDERR2=1, ha a motorra EnDat jeladó van szerelve, és a jeladóba épített hőmérséklet érzékelő a beállított értéknél magasabb hőmérsékletet mér.

DN_BVERR: Túlfeszültség hiba a sínén

DN_BVERR=1, ha a sínfeszültség meghaladta a beállított értéket.

DN_CURERR: Túl nagy a motoráram

DN_CURERR=1, ha a motoráram meghaladta a hajtáson beállított $I_{\max}$ értéket.

DN_CMEERR: Árammérési hiba

DN_CMEERR=1, ha árammérési hiba lépett föl.

DN_HALERR: Hall kommutáló jel hiba

DN_HALERR=1, ha kommutáló jel hiba lépett föl.

DN_HASERR: Hall szekvencia jel hiba

DN_HASERR=1, ha a kommutáló jel szekvenciája nem Gray kód szerinti.

DN_SRTERR: Időtúllépés hiba

DN_SRTERR=1, ha a hajtásban lejárt a központi watchdog időzítő. A hajtás rossz.

DN_CWDER1: CAN watchdog hiba

DN_CWDER1=1, ha a hajtásban lejárt a CAN kommunikáció watchdog időzítője. A CAN kommunikáció rossz.

DN_CHERR1: CAN egyéb hiba

DN_CHERR1=1, ha a CAN kommunikációban hiba lépett föl.

DN_ECTERR: EtherCAT időtúllépés hiba

DN_ECTERR=1, ha a hajtásban lejárt az EtherCAT kommunikáció watchdog időzítője. Az EtherCAT kommunikáció rossz.

DN_PDPINT: IGBT hiba

DN_PDPINT=1, ha hiba lépett föl a motor táplálásánál. Lehet áramköri, kábelezési hiba.

DN_PRMERR: Hibás a paramétertábla

DN_PRMERR=1, ha az aktív paramétertábla sérült.

DN_PRGERR: Hibás a hajtásprogram

DN_PRGERR=1, ha a hajtás működtető programja megsérült, az ellenőrző összege hibás.

DN_FOLERR: Sebességkövetési hiba

DN_FOLERR=1, ha a sebesség alapjelet a hajtás nem tudta követni egy beállított időn belül.

DN_OVHERR: Hővédelem

DN_OVHERR=1, ha a motor hővédelme megszólalt. Lehet PTC, vagy hőmérsékleti modell.

A hajtások duplaszavas bemenő regiszterei

DN_STAT: Hajtás státusz regiszter (DWORD)

A szimbólummal a hajtás státuszbitjeihez duplaszavasan lehet hozzáférni.

DN_ERR: Hajtás hiba jelzők regisztere (DWORD)

A szimbólummal a hajtás hibabitjeihez duplaszavasan lehet hozzáférni.

A hajtások duplaszavas kimenő regiszterei

DP_CTRL: Hajtás kontrol regiszter (DWORD)

A szimbólummal a hajtás kontroljeleihez duplaszavasan lehet hozzáférni.

7.8 Jeladó fogadó bemenetek és analóg/léptető motor/CAN illesztő kimenetek

A jeladó fogadó bemenetek és analóg/léptető motor/CAN illesztő kimenetek az EtherCAT buszra illeszthetők.

A be-, és kimeneti illesztő egységeket az alábbi esetekben használjuk:

- Az EtherCAT-es hajtások a motorra szerelt jeladóról mért pozíciót adják át az NC-nek.
Azokban az esetekben, amikor a motorra szerelt jeladó helyett másikat kell használni, megfelelő illesztő elektronikáról kell a jelet venni. Ilyen esetek például az eszterga orsókra szerelt menetvágó jeladók, vagy ha egy tengelyen mérőlécet használunk pozíció-mérésre.
- A másik eset, amikor nem EtherCAT-es, hanem például analóg, vagy CAN buszos bemenetű, vagy léptetőmotoros hajtásokat kell a vezérlőhöz illeszteni.

Az alábbi illesztő egységek állnak rendelkezésre:

– ENDAT:

Bemenetei:

2 db. EnDat 2.2 kommunikációs protokollal rendelkező jeladó (mérőléc, szögmérő) jeleinek fogadására alkalmas.

– TTLAI:

Bemenetei:

Az egység a bemenetén 2db. inkrementális, TTL jeladó jeleit képes fogadni.

Kimenetei:

ECAT-TTLASM szoftver verzió esetén a 2 db. kimenetén kiadható

analóg alapjel, vagy

Pulse-Dir impulzussorozat és iránybit léptetőmotorok számára, vagy

CW-CCW kétirányú impulzussorozat léptetőmotorok számára.

Mindhárom kimenet kiadásra kerül csatornánként, bekötéssel választható ki, hogy melyiket alkalmazzuk.

ECAT-TACHO szoftver verzió esetén a 2 db. kimenetén kiadható

az analóg alapjel és tachojel különbségéből képzett rendelkezőjel analóg bemenetű hajtások számára.

– TTLCAN:

Bemenetei:

Az egység a bemenetén 2db. inkrementális, TTL jeladó jeleit képes fogadni.

Kimenetei:

2 db. CAN-buszos bemenetű NCT hajtás kezelésére alkalmas.

Az illesztő egységek kommunikációs regiszterei és jelzői a DN_, DP_ hajtás jelzőkön tartanak a PLC-vel kapcsolatot. A rendszer maximum 48 db. EtherCAT-es hajtás és illesztő egység jeleit tudja kezelni (32 tengely + 16 orsó).

Az alábbiakban csak azokat a jeleket tüntetjük fel, amelyeket az illesztő egységek kezelnek.

Az illesztő egységek bites státusz és kontrol jelei

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
DN_INC	Inkrementális jeladó a hajtáson	DP_ERRCLR	Hajtás hiba törlés

Az illesztő egységek bites hibajelzői

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
DN_ERROR	Hajtás hiba van (DN_ERR>0)		
DN_EDERR1	Jeladó hiba		
DN_ECTERR	EtherCAT időtúllépés hiba		

Az illesztő egységek duplaszavas változói

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
DN_STAT	Hajtás státusz regiszter (DWORD)	DP_CTRL	Hajtás kontrol regiszter (DWORD)
DN_ERR	Hajtás hiba jelzők regisztere (DWORD)		

Az EtherCAT-es illesztő egységek címzése

A vezérlő maximum 48 db. EtherCAT-es hajtás és illesztő egység jeleit kezeli.

Az illesztő egységek címeinek beállítását a vezérlő *Szerviz menü*jében az *ECAT beállítások* ablak beváltása után a ***megfelelő illesztő egységet kiválasztva*** végezhetjük el. Válasszuk ki a bal oldali panelen a megfelelő egységet és a *Setting* fület választva állítsuk be az illesztő egység *címét*.

A címek értékei:

Not used, 1, 2, ..., 48

lehetnek.

Az illesztő egységek és a tengelyek címeinek összerendelése

A vezérlő maximum 32 tengelyt kezel. A tengelyek címei a vezérlőben:

1, 2, ..., 32

lehetnek.

Az alapjel kiadása:

Az N0502 Axis Output Type paraméteren válasszuk az EtherCAT-32 opciót.

Az N0503 Axis Output Address paraméterre írjuk be annak az illesztő egységeknek a címét, amelyikkel az adott tengelyt mozgatni kívánjuk.

Példa:

Az

N0100 Axis Name1 A4=C

paraméter a 4. tengely címét C-nek határozza meg.

A C tengelyt egy analóg bemenetű hajtás mozgatja. TTLAI illesztő egységének címe az EtherCAT beállítások között: 6.

Ekkor az N0503 Axis Output Address A4=6,

azaz a vezérlő 4. tengelye a 6. illesztő egységnek adja ki az alapjelet.

A jeladó jeleinek fogadása:

Az N0500 Axis Input Type paraméteren válasszuk az EtherCAT opciót.

Az N0501 Axis Input Address paraméterre írjuk be annak az illesztő egységnek a címét, amelyikről a jeladó jeleit fogadni kívánjuk.

☞ **Figyelem!** Az N0503 Axis Output Address paraméter és az N0501 Axis Input Address paraméter lehet különböző!

Példa:

Az

N0100 Axis Name1 A2=Y

paraméter a 2. tengely címét Y-nak határozza meg.

Az Y tengely EtherCAT-es hajtásának címe az EtherCAT beállítások között legyen 2. (N0503 Axis Output Address A2=2)

Az Y tengelyre EnDat mérőléc van szerelve. Az Y mérőléc jeleit a 12-es című ENDAT illesztő egység kezeli.

Ekkor az N0501 Axis Input Address A2=12,

azaz a vezérlő 2. tengelye a 12. illesztő egységtől veszi a jeladó jeleit.

Az illesztő egységek és az orsók címeinek összerendelése

A vezérlő maximum 16 orsót kezel. Az orsók címei a vezérlőben:

1, 2, ..., 16

lehetnek.

Az alapjel kiadása:

Az N0602 Spindle Output Type paraméteren válasszuk az EtherCAT opciót.

Az N0603 Spindle Output Address paraméterre írjuk be annak az illesztő egységnek a címét, amelyikkel az adott orsót mozgatni kívánjuk.

Példa:

Az

N0605 Spindle Name2 S1=1

paraméter az 1. orsó címét S1-nek határozza meg.

Az S1 orsó hajtás CAN buszos bemenetű, az alapjelet a hajtásnak a TTLCAN illesztő egység adja, amelynek címe az EtherCAT beállítások között: 8.

Ekkor az N0603 Spindle Output Address S1=8,
azaz a vezérlő 1. orsója a 8. illesztő egységnek adja ki az alapjelet.

A jeladó jeleinek fogadása:

Az N0600 Spindle Input Type paraméteren válasszuk az EtherCAT opciót.

Az N0601 Spindle Input Address paraméterre írjuk be annak az illesztő egységnek a címét, amelyiktől a jeladó jeleit fogadni kívánjuk.

☞ **Figyelem!** Az N0603 Spindle Output Address paraméter és az N0601 Spindle Input Address paraméter lehet különböző!

Példa:

Az

N0605 Spindle Name2 S1=1

paraméter az 1. orsó címét S1-nek határozza meg.

Az S1 orsó hajtás illesztő egységének címe az EtherCAT beállítások között: 8.

A jeladó jeleit az orsóra szerelt jeladóról kívánjuk venni. (Nem a motorról) Az orsóra szerelt jeladó jeladó jeleit egy TTLAI illesztő egység fogadja, amelynek címe 21.

Ekkor az N0601 Spindle Input Address S1=21,

azaz a vezérlő 1. orsója a 21. illesztő egységtől veszi a jeladó jeleit.

Hivatkozás a PLC programban tengelyekre, orsókra és hajtásokra

Az

AN_ xx és AP_ xx

NC szimbólumok mindig tengelyekre vonatkoznak. Ezekre a szimbólumokra mindig tengely-index alapján kell hivatkozni:

#0: 1. tengely, ..., #31: 32.tengely.

Az

SN_ xx és SP_ xx

NC szimbólumok mindig orsókra vonatkoznak. Ezekre a szimbólumokra mindig orsóindex alapján kell hivatkozni:

#0: 1. orsó, ..., #15: 16. orsó.

A

DN_ xx és DP_ xx

NC szimbólumok mindig hajtásokra, vagy illesztő egységekre vonatkoznak. Ezekre a szimbólumokra mindig hajtásindex alapján kell hivatkozni:

#0: 1. hajtás, ..., #47: 48. hajtás.

Egy adott tengely, vagy orsó tengely-, illetve orsóindexe különböző lehet az adott tengely vagy orsó hajtásának indexétől!

Az illesztő egységek státuszjelei (bemenetek)

DN_INC: Inkrementális jeladó a hajtáson

Ha a jelző DN_INC=0, a motoron abszolút, EnDAT jeladó van,
ha a jelző DN_INC=1, a motoron inkrementális jeladó van.

Az illesztő egységek kontrol jelei (kimenetek)

DP_ERRCLR: Hajtás hiba törlés (Illesztő egység hiba törlés)

Ha az illesztő egységen hiba van, azaz DN_ERR>0 (a hibaregiszter nem 0) a hibát törölni kell.

A DP_ERRCLR=1 állapot törli a hibát. A jelet addig 1-ben kell tartani, amíg a DN_ERR hibaregiszter 0 nem lesz, vagy a DN_ERROR=0 állapot bekövetkezik.

Az illesztő egységek bites hibajelzői (bemenetek)

A PLC programnak nem kell a különböző illesztő egység hibákra üzenetet küldeni, ezt az NC megteszi. Csak a hiba törlését kell kiadni a DP_ERRCLR=1 jelző állítással.

DN_ERROR: Hajtás hiba van (DN_ERR>0) (Illesztő egység hiba van)

DN_ERROR=1, ha a hibaregiszter valamelyik további bitje hibát jelez.

DN_EDERR1: Jeladó hiba

DN_EDERR1=1, ha az illesztő egység hibát észlel a jeladóról.

DN_ECTERR: EtherCAT időtúllépés hiba

DN_ECTERR=1, ha az illesztő egységben lejárt az EtherCAT kommunikáció WatchDog időzítője. Az EtherCAT kommunikáció rossz.

Az illesztő egységek duplaszavas bemenő regiszterei

DN_STAT: Hajtás státusz regiszter (DWORD) (Illesztő egység státusz regiszter)

A szimbólummal az illesztő egység státuszbitjeihez duplaszavasán lehet hozzáférni.

DN_ERR: Hajtás hiba jelzők regisztere (DWORD) (Illesztő egység hiba jelzők regisztere)

A szimbólummal az illesztő egység hibabitjeihez duplaszavasán lehet hozzáférni.

A hajtások duplaszavas kimenő regiszterei

DP_CTRL: Hajtás kontrol regiszter (DWORD) (Illesztő egység kontrol regiszter)

A szimbólummal az illesztő egység kontroljeleihez duplaszavasán lehet hozzáférni.

TTLAI illesztő egység ECAT-TACHO szoftver esetén a kontrol szó kezelése:

A PLC-n keresztül a DP_CTRL felső szavába (16-tól 31 bitig) a következő képlet szerint be kell írni egy egész számot:

$(\text{jeladó impulzusszáma} * 4) * (\text{maximális motorfordulatszám} / 60) / 5000$

Ha ide nullát írunk, akkor az egység nem végez tachó-kompenzációt, az alapjelnek megfelelő analóg jelet adja ki.

7.9 PLC-ből elérhető funkciógombok

A PLC program számára a vezérlő képernyőjén elérhető 32 db. funkciógomb. A funkciógombok felíratát a PLC programban kell megadni a PLC Editor jobb alsó paneljén a PLC gombok földre kattintva.

A 32 funkciógombhoz 2 duplaszavas regiszter tartozik: a bemenő regiszter a gombhoz, a kimenő a gomb lámpájához.

A funkciógombok duplaszavas változói:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_MSG	32 funkciógomb gombja a képernyőn, amit PLC használhat (DWORD)	P_MSG	32 funkciógomb lámpája a képernyőn, amit PLC használhat (DWORD)

Az N_MSG és a P_MSG regiszterekhez mint referenciához képest bites szimbólumokat hozhatunk létre, amely szimbólumok az egyes funkciógombokat és azok lámpáit jelölik.

Példa:

B_LUB szimbólum legyen a kenőszivattyú kézi indításának szimbóluma, amelyet az F23 funkciógombról akarunk indítani.

A PLC gombok panelen az F23 gomb szövegdobozába a SZÁNKENÉS felíratot írtuk, amely a képernyőn a megfelelő funkciógombon megjelenik.

A szimbólum hozzáadása panelt így töltjük ki:

Válasszunk referencia megadást,

Szimbólikus név: B_LUB, szánkenés gomb

Bázis: N_MSG,

Eltolás: 0,

Bit: 22 (23. gomb = 22-es bit).

Szimbólikus név: L_LUB, szánkenés gomb lámpája

Bázis: P_MSG,

Eltolás: 0,

Bit: 22 (23. gomb = 22-es bit).

7.10 A paraméteren állítható pozíció kapcsolók

Paraméteren felvehető 32 db. pozíció kapcsoló.

Az

N1100+n SWn Axis Number

paraméter azt mondja meg, hogy az n-edik kapcsolót melyik tengelyre állítjuk be.

Az

N1132+n SWn Min Pos n

paraméter az n. kapcsoló negatív irányba eső végének gépi pozícióját határozza meg az adott tengelyen.

Az

N1164+n SWn Max Pos n

paraméter az n. kapcsoló pozitív irányba eső végének gépi pozícióját határozza meg az adott tengelyen.

Minden **PLC ciklus** előtt a vezérlő megállapítja, hogy az n-edik kapcsoló létezik-e. Ha igen, akkor a kapcsolóra kijelölt tengely **jeladóról** kiszámolt abszolút pozíciója alapján az n-edik kapcsolóra (n-1 indexű PLC jelző) az alábbi jeleket kezeli.

A pozíció kapcsolók jelei a PLC számára bemenő jelek.

A pozíció kapcsolók bites változói:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_SW	Tengely 1. paraméteres kapcsolón van		
N_SWN	Tengely 1. paraméteres kapcsolótól negatív irányban van		
N_SWP	Tengely 1. paraméteres kapcsolótól pozitív irányban van		

A pozíció kapcsolók bemenetei

N_SW: Tengely 1. paraméteres kapcsolón van

Ha a kijelölt tengely az 1. kapcsolón van, azaz

N1133 SW1 Min Pos 1 < Pozíció < N1165 SW1 Max Pos 1

a jelző 1 állapotot vesz fel.

N_SWN: Tengely 1. paraméteres kapcsolótól negatív irányban van

Ha a kijelölt tengely az 1. kapcsolótól negatív irányban van, azaz

Pozíció < N1133 SW1 Min Pos 1

a jelző 1 állapotot vesz fel.

N_SWP: Tengely 1. paraméteres kapcsolótól pozitív irányban van
 Ha a kijelölt tengely az 1. kapcsolótól pozitív irányban van, azaz
 Pozíció > N1133 SW1 Max Pos 1
 a jelző 1 állapotot vesz fel.

A többi kapcsoló az 1. kapcsolóhoz képesti indexelt hivatkozással érhető el. Az egyes kapcsolóknak szimbolikus nevet is adhatunk, de vigyázzunk arra, hogy a neveket az N_SW, ... stb. szimbólumokhoz, mint bázishoz képest relatív hivatkozással vegyük fel.

 **Figyelem:** Vegye figyelembe a jelek alkalmazásánál a tengely maximális mozgási sebességét, a kapcsoló hosszát és a mintavétel frekvenciáját (PLC ciklusidő).

A pozíció kapcsolók duplaszavas változói:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_SW0	A Position Switches paraméteren megadott 1. kapcsoló regisztere (DWORD)		

N_SW0: A Position Switches paraméteren megadott 1. kapcsoló regisztere (DWORD)
 Az 1. pozíció kapcsoló jelei az N_SW0 regiszterben duplaszavasán is elérhetőek. A többi kapcsoló regiszterét indexelt hozzáféréssel lehet kiolvasni.

7.11 Hozzáférés a PLC Constants paramétercsoporthoz

A PLC Constants paramétercsoportban a PLC program számára rendelkezésre áll

64 db. bites,

32 db. duplaszavas egész és

32 db. lebegőpontos

típusú, szabad felhasználású paraméter. Ezeket a paramétereket a PLC program direkt olvashatja a PLC memóriából, a bites paraméterekre érintkezőt definiálhat.

A PLC paraméterek közvetlenül nem írhatók a PLC programból!

A bites PLC paraméterek:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_P00	PLCBits0 paraméter P00 bitje		
N_P01	PLCBits0 paraméter P01 bitje		
N_P02	PLCBits0 paraméter P02 bitje		
N_P03	PLCBits0 paraméter P03 bitje		
N_P04	PLCBits0 paraméter P04 bitje		
N_P05	PLCBits0 paraméter P05 bitje		
N_P06	PLCBits0 paraméter P06 bitje		
N_P07	PLCBits0 paraméter P07 bitje		
N_P10	PLCBits1 paraméter P10 bitje		
N_P11	PLCBits1 paraméter P11 bitje		
N_P12	PLCBits1 paraméter P12 bitje		
N_P13	PLCBits1 paraméter P13 bitje		
N_P14	PLCBits1 paraméter P14 bitje		
N_P15	PLCBits1 paraméter P15 bitje		
N_P16	PLCBits1 paraméter P16 bitje		
N_P17	PLCBits1 paraméter P17 bitje		
N_P20	PLCBits2 paraméter P20 bitje		
N_P21	PLCBits2 paraméter P21 bitje		
N_P22	PLCBits2 paraméter P22 bitje		
N_P23	PLCBits2 paraméter P23 bitje		
N_P24	PLCBits2 paraméter P24 bitje		
N_P25	PLCBits2 paraméter P25 bitje		
N_P26	PLCBits2 paraméter P26 bitje		
N_P27	PLCBits2 paraméter P27 bitje		
N_P30	PLCBits3 paraméter P30 bitje		

7.11 Hozzáférés a PLC Constants paramétercsoporthoz

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_P31	PLCBits3 paraméter P31 bitje		
N_P32	PLCBits3 paraméter P32 bitje		
N_P33	PLCBits3 paraméter P33 bitje		
N_P34	PLCBits3 paraméter P34 bitje		
N_P35	PLCBits3 paraméter P35 bitje		
N_P36	PLCBits3 paraméter P36 bitje		
N_P37	PLCBits3 paraméter P37 bitje		
N_P40	PLCBits4 paraméter P40 bitje		
N_P41	PLCBits4 paraméter P41 bitje		
N_P42	PLCBits4 paraméter P42 bitje		
N_P43	PLCBits4 paraméter P43 bitje		
N_P44	PLCBits4 paraméter P44 bitje		
N_P45	PLCBits4 paraméter P45 bitje		
N_P46	PLCBits4 paraméter P46 bitje		
N_P47	PLCBits4 paraméter P47 bitje		
N_P50	PLCBits5 paraméter P50 bitje		
N_P51	PLCBits5 paraméter P51 bitje		
N_P52	PLCBits5 paraméter P52 bitje		
N_P53	PLCBits5 paraméter P53 bitje		
N_P54	PLCBits5 paraméter P54 bitje		
N_P55	PLCBits5 paraméter P55 bitje		
N_P56	PLCBits5 paraméter P56 bitje		
N_P57	PLCBits5 paraméter P57 bitje		
N_P60	PLCBits6 paraméter P60 bitje		
N_P61	PLCBits6 paraméter P61 bitje		
N_P62	PLCBits6 paraméter P62 bitje		
N_P63	PLCBits6 paraméter P63 bitje		
N_P64	PLCBits6 paraméter P64 bitje		
N_P65	PLCBits6 paraméter P65 bitje		
N_P66	PLCBits6 paraméter P66 bitje		
N_P67	PLCBits6 paraméter P67 bitje		
N_P70	PLCBits7 paraméter P70 bitje		
N_P71	PLCBits7 paraméter P71 bitje		
N_P72	PLCBits7 paraméter P72 bitje		
N_P73	PLCBits7 paraméter P73 bitje		
N_P74	PLCBits7 paraméter P74 bitje		

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_P75	PLCBits7 paraméter P75 bitje		
N_P76	PLCBits7 paraméter P76 bitje		
N_P77	PLCBits7 paraméter P77 bitje		

Az egyes N_Pij paramétereknek szimbolikus nevet is adhatunk, de vigyázzunk arra, hogy a neveket az N_Pij szimbólumokhoz, mint bázishoz képest relatív hivatkozással vegyük fel.

A duplaszavas PLC paraméterek:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_PDW1	PLC DWord1 paraméter értéke (DWORD)		

N_PDW1: PLC DWord1 paraméter értéke (DWORD)

Az N_PDW1 regiszterből az N1209 PLC DWord1 paraméter értéke olvasható ki. A többi paraméter értékhez indexelt utasítással lehet hozzáférni.

Az egyes paramétereknek szimbolikus nevet is adhatunk, de vigyázzunk arra, hogy a neveket az N_PDW1 szimbólumokhoz, mint bázishoz képest relatív hivatkozással vegyük fel.

A lebegőpontos PLC paraméterek:

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_PDB1	PLC Double1 paraméter értéke (double)		

N_PDB1: PLC Double1 paraméter értéke (double)

Az N_PDB1 regiszterből az N1241 PLC Double1 paraméter értéke olvasható ki. A többi paraméter értékhez indexelt utasítással lehet hozzáférni.

☞ **Figyelem!** Indexelt hivatkozásnál figyelembe kell venni, hogy a double érték 2 DWORD-ön van ábrázolva, ezért az indexet 2-esével kell léptetni.

Az egyes paramétereknek szimbolikus nevet is adhatunk, de vigyázzunk arra, hogy a neveket az N_PDB1 szimbólumokhoz, mint bázishoz képest relatív hivatkozással vegyük fel.

7.12 A globális változók

Az globális vagy közös változók olyan az NC-től a PLC-be, vagy a PLC-ből az NC-be menő változók, amelyek az összes csatornára közősek.

Az

N-nel kezdődő változók az NC-től a PLC-be mennek, míg a
P-vel kezdődő változók a PLC-től az NC-be.

7.12.1 A bites globális változók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_P2MS	2 Time Slice periódusú órajel (csak Int0 modulban, direkt lekérdezés)	P_MONREQ	Machine On bekapcsolás kérés a PLC-ből
N_P2T	2 PLC ciklus periódusú órajel	P_HOLD0	Előtolás állj minden csatornán
N_P100MS	100 msec periódusú órajel	P_SHTDNREQ	NC leállítás kérése
N_P1S	1 sec periódusú órajel		
N_P1M	1 min periódusú órajel		
N_ON	mindig 1		
N_OFF	mindig 0		
N_B7	Fentartva		
N_NVRAMOK	A nem felejtő PLC változók sikeresen visszatöltődtek		
N_FIRSTCC	Bekapcsolás utáni első futás		
N_NCREADY	NC üzemkész		
N_MONST	Machine ON jel állapota		
N_MONDIS	Machine ON jel bekapcsolása tiltott		
N_CLRMSG	Üzenet törlése		
N_MSGA	Üzenet van a képernyőn		
N_MSG0	PLC üzenet a képernyőn		
N_MSG1	Tengely, vagy orsó üzenet van a képernyőn		
N_MSG2	Csatorna mondatelőkészítő üzenet van a képernyőn		
N_MSG3	Csatorna végrehajtó üzenet van a képernyőn		
N_MSG4	Makró hibaüzenet (#3000) van a képernyőn		

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_MSG5	Makró üzenet (#3006) van a képernyőn		
N_MSG6	Fenntartva		
N_MSG7	Fenntartva		
N_MSG8	Valós idejű rendszerüzenet van a képernyőn		
N_MSG9	Ember-gép kapcsolat üzenet van a képernyőn		
N_TLSRCH	Szerszámkeresés folyamatban		
N_TLMD	A szerszámkezelési táblázat módosítás alatt		
N_TLSV	A szerszámkezelési táblázatok mentés alatt		
N_TLEDT	A szerszámkezelési táblázatok szerkesztés alatt		
N_PTEDT	A palettakezelő táblázat szerkesztés alatt		
N_SIMU	Az NC szoftver szimulátoron (PC-n) fut		
N_NVECAT	A nemfelejtő változók EtherCAT-en keresztül érhetők el		
		P_DIR	Hozzáférés tiltás: Könyvtár műveletek
		P_PRGE	Hozzáférés tiltás: Program szerkesztés
		P_WOFFS	Hozzáférés tiltás: Munkadarab nullpontok
		P_COMPG	Hozzáférés tiltás: Geometriai korrekciók
		P_COMPW	Hozzáférés tiltás: Kopás korrekciók
		P_TLTAB	Hozzáférés tiltás: Szerszám táblázatok
		P_MAC	Hozzáférés tiltás: Makróváltozók
		P_TRCTR	Hozzáférés tiltás: Idő, mkdb. számlálók
		P_RUNAUT	Hozzáférés tiltás: Auto futtatás
		P_RUNMDI	Hozzáférés tiltás: MDI futtatás
		P_PAR	Hozzáférés tiltás: Paraméterek
		P_PLC	Hozzáférés tiltás: PLC program

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
		P_SVRC	Hozzáférés tiltás: Szerviz műveletek
		P_PTTAB	Hozzáférés tiltás: Paletta táblázat

Az NC-től a PLC-be menő bites globális változók

N_P2MS: 2 Time Slice periódusú órajel (csak Int0 modulban, direkt lekérdezés)

N_P2T: 2 PLC ciklus periódusú órajel

N_P100MS: 100 msec periódusú órajel

N_P1S: 1 sec periódusú órajel

N_P1M: 1 min periódusú órajel

Az NC által szolgáltatott órajelek közül az N_P2MS jelet a jel frekvenciája miatt csak az Int0 modulban van értelme felhasználni. A többi órajel a Főprogramban is felhasználható.

N_ON: mindig 1

N_OFF: mindig 0

A két szimbólumot akkor használhatjuk, ha, például, egy utasításdoboz valamelyik bemenetére fix értéket kell kötni, vagy egy érintkezőt ki akarunk kommentezni.

N_NVRAMOK: A nem felejtő PLC változók sikeresen visszatöltődtek

Ha a nem felejtő PLC tárban adatokat tárolunk, a bekapcsolás utáni első ciklusban (N_FIRSTCC=1) megvizsgálhatjuk a jelzőt. Ha a jelző értéke 0, a PLC változók megsérültek.

N_FIRSTCC: Bekapcsolás utáni első futás

A jelző a vezérlő bekapcsolása, vagy újraindítása után 1 PLC ciklus ideig 1. Ez alatt az idő alatt kell a PLC programnak a szükséges inicializálásokat elvégeznie.

N_NCREADY: NC üzemkész

Az NC üzemkész állapotát jelzi.

N_MONST: Machine ON jel állapota

Ha a jelző a gép bekapcsolt állapotában 0-ra vált vészleállítást kell kezdeményezni a PLC-ből.

N_MONDIS: Machine ON jel bekapcsolása tiltott

A gép bekapcsolását kezdeményezni, azaz a P_MONREQ jelzőt bekapcsolni csak akkor szabad, ha a gép bekapcsolása nincs tiltva, azaz N_MONDIS jelző 0.

N_CLRMSG: Üzenet törlése (aktív 1)

Az üzenet a kijelző legfelső sorában jelenik meg.

Ha egy időben több üzenet váratkozik, mindig az időben legutoljára bejött üzenet látszik az üzenetmezőben. (Last in First Out)

A

CANCEL

gomb mindig a kijelzőn látható, utoljára beérkezett üzenetet törli és 1-be állítja az N_CLRMSG jelzőt.

Ha az üzenetmezőre kattintunk, legördülő listában az összes üzenet láthatóvá válik.

Ekkor kiválaszthatunk egy tetszőleges üzenetet és a

Törlés (kiválasztott)

funkciógombról törölhetjük azt. Ekkor (a CANCEL gomb megnyomása nélkül is) az N_CLRMSG jelző működik.

Választhatjuk az

Összes törlése

funkciógombot is, melynek hatására a pufferben lévő összes üzenet törlődik. Ekkor is az N_CLRMSG jelző működik.

A PLC programban mindig az N_CLRMSG jelzőt kell használni az üzenet törlésére.

N_MSGA: Üzenet van a képernyőn

A jelző értéke 1, ha bármilyen üzenet van a képernyő legfelső, üzenetsorában. A jelzőt fel lehet használni pl. hibajelző géplámpa bekapcsolására.

N_MSG0: PLC üzenet a képernyőn**N_MSG1:** Tengely, vagy orsó üzenet van a képernyőn**N_MSG2:** Csatorna mondatelőkészítő üzenet van a képernyőn**N_MSG3:** Csatorna végrehajtó üzenet van a képernyőn**N_MSG4:** Makró hibaüzenet (#3000) van a képernyőn**N_MSG5:** Makró üzenet (#3006) van a képernyőn**N_MSG6:** Fenntartva**N_MSG7:** Fenntartva**N_MSG8:** Valós idejű rendszerüzenet van a képernyőn**N_MSG9:** Ember-gép kapcsolat üzenet van a képernyőn

Az N_MSG0, ..., N_MSG9 jelzők a képernyő legfelső, üzenetsorában látható, utoljára bejött üzenet fajtáját jelzi.

N_TLSRCH: Szerszámkeresés folyamatban**N_TLMD:** A szerszámkezelési táblázatok módosítás alatt**N_TLSV:** A szerszámkezelési táblázatok mentés alatt**N_TLEDT:** A szerszámkezelési táblázatok szerkesztés alatt

A fenti jelzők információt nyújtanak a szerszámkezelő táblázat foglaltságának okáról.

N_PTEDT: A palettakezelő táblázat szerkesztés alatt

A palettakezelő táblázat valamelyik sorát szerkesztik.

N_SIMU: Az NC szoftver szimulátoron (PC-n) fut

Ha az NC szoftver szimulátoron, vagyis PC-n fut, a PLC erről a tényről az N_SIMU jelző 1 állapota alapján kap értesítést.

N_NVECAT: A nemfelejtő változók EtherCAT-en keresztül érhetőek el

Ha a vezérlőhöz nincs akkumulátor, hanem EtherCAT-en keresztül működő FRAM-os egység van csatolva, akkor N_NVECAT=1. Ilyenkor a PLCNVRAM-tól a PLCRAM-ig terjedő PLC memóriaterület nem kerül kikapcsoláskor mentésre.

A megőrzendő PLC változók mentéséről, illetve azok beolvasásáról a PLC program-mozónak kell gondoskodnia a z MR10 és MW11 utasítások használatával.

Az PLC-től az NC-be menő bites globális változók

P_MONREQ: Machine On bekapcsolás kérés a PLC-ből

A gép bekapcsolását végző MON (Machine ON) kimenet, minden EtherCAT fejegységen megtalálható.

A MON kimenet címének beállítását a vezérlő *Szerviz menüjében az ECAT beállítások* ablak beváltása után a *gépi kezelőpanel* beállításainál végezhetjük el. Válasszuk ki a bal oldali panelen az EtherCAT-Head egységet és a *Setting* fület választva állítsuk be a MON kimenet *címét*. Ha egy fejegységen a

Not used

opciót választjuk azon az egységen a MON kimenetet nem kezeli. Ha

1, 2, stb.

címet állítunk be, azon az egységen a MON kimenetet kezeli.

A P_MONREQ jelző bekapcsolása bekapcsolja az összes, nem Not used-ra állított fejegységen a MON kimenetet. A jelzőt csak akkor szabad bekapcsolni, ha a gép bekapcsolása nincs tiltva, azaz N_MONDIS jelző 0.

P_HOLD0: Előtölés állj minden csatornán

A jelző 1-be kapcsolása leállítja az összes tengely mozgását az összes csatornában még akkor is, ha az override és a stop tiltva van. Ha az override és stop tiltva van, menetfűrés, vagy menetvágás esetén az orsót a PLC-nek le kell állítania.

P_SHTDNREQ: NC leállítás kérése (Shutdown Request)

Az NC szerelhető akkumulátoros tápegységgel is, amely képes 1-2 percre biztosítani a vezérlő és a szükséges EtherCAT elektronika tápellátását. Hálózatkimaradás, vagy a gép főkapcsolójának kikapcsolása esetén a tápegység egy tetszőleges interfész

bemeneten jelez a PLC-nek. A PLC program a P_SHTDNREQ jelző bekapcsolásával kérheti a rendszer leállítását.

P_DIR: Hozzáférés tiltás: Könyvtár műveletek

P_DIR=1 állapotban
nem törölhet,
nem írhat felül fájlt, vagy mappát.
Új fájlt létrehozhat, illetve fájlt másolhat.

P_PRGE: Hozzáférés tiltás: Program szerkesztés

P_PRGE=1 állapotban minden alkatrészprogram szerkesztése tiltott.

P_WOFFS: Hozzáférés tiltás: Munkadarab nullpontok

P_WOFFS=1 állapotban nem írhatja át a munkadarab nullpontokat, sem táblázatban, sem beméréssel.

P_COMPG: Hozzáférés tiltás: Geometriai korrekciók

P_COMPG=1 állapotban nem írhatjs át a geometriai korrekciókat.

P_COMPW: Hozzáférés tiltás: Kopás korrekciók

P_COMPW=1 állapotban nem írhatja át a kopáskorrekciókat.

P_TLTAB: Hozzáférés tiltás: Szerszám táblázatok

P_TLTAB=1 állapotban nem írhatja át a szerszámkezelő táblázat egyik elemét sem.

P_MAC: Hozzáférés tiltás: Makróváltozók

P_MAC=1 állapotban ne írhat át egyetlen lokális, vagy globális makróváltozót sem.

P_TRCTR: Hozzáférés tiltás: Idő, mkdb. számlálók

P_TRCTR=1 állapotban nem írhatja át az idő munkadarab számláló táblázat egyik értékét sem.

P_RUNAUT: Hozzáférés tiltás: Auto futtatás

P_RUNAUT=1 állapotban
nem jelölhet ki programot automata üzemmódban való futásra,
nem törölhet automata futásra kijelölt programot.

P_RUNMDI: Hozzáférés tiltás: MDI futtatás

P_RUNMDI=1 állapotban
nem jelölhet ki programot kézi adatbeviteli üzemmódban való futásra,
nem törölhet kézi adatbeviteli üzemmódban való futásra kijelölt programot.

P_PAR: Hozzáférés tiltás: Paraméterek

P_PAR=1 állapotban
nem írhat át (szerkeszthet) paramétert,
nem tölthet be paramétert,
de paramétert menthet, vagy exportálhat.

P_PLC: Hozzáférés tiltás: PLC program

P_PLC=1 állapotban
nem írhat át (szerkeszthet) PLC programot,
nem tölthet be PLC programot,
de PLC programot menthet, zöld folyamat nézhet.

P_SVRC: Hozzáférés tiltás: Szerviz műveletek

P_SVRC=1 állapot a szerviz műveleteket a következőképp befolyásolja:
I/O teszt: BitSet, BitReset, HexaSet művelet tiltva,
Szimb. I/O, Logikai analízátor: mentésnél felülírás tiltva
ECAT Beállítások: a panelek editálása tiltott.

P_PTTAB: Hozzáférés tiltás: Paletta táblázat

P_PTTAB=1 állapotában a paletta táblázat nem szerkeszthető.

7.12.2 A duplaszavas globális változók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
N_ACTMSG	A kijelzőn látszó aktív üzenet azonosító száma (DWORD)	P_CHSEL	A tasztatúra melyik csatornára érvényes (0,1,2...) (DWORD)

Az NC-től a PLC felé menő duplaszavas globális változók

N_ACTMSG: A kijelzőn látszó aktív üzenet azonosító száma (DWORD)

A legfelső, üzenet sortól balra lévő ablakban kiolvasható az utolsó, a kijelzőn megjelenő üzenet 8 decimális számjegyű kódja. Az N_ACTMSG regiszterben ez a kód jelenik meg.

Az PLC-től az NC felé menő duplaszavas globális változók

P_CHSEL: A tasztatúra melyik csatornára érvényes (0,1,2...) (DWORD)

Többcsatornás vezérlőn van jelentése. Képernyőkép lehívásakor, például az Eltolások menüpontban a Bemérés tételt válsztva a vezérlő annak a csatornának a Bemérés

képernyőjét hozza be elsőként, amelyik csatorna száma a P_CHSEL regiszterben be lett állítva. Utána már a PgUp, PgDn gombokkal lapozhatunk a különböző csatornák között.

Egycsatornás vezérlőn nem kell használni.

7.13 A tengelykezelő változók

Az tengelykezelő változók olyan az NC-től a PLC-be, vagy a PLC-ből az NC-be menő változók, amelyek tengelyenként indexelődnek. Az összes itt közölt szimbólum az első (0-ás indexű) tengelyre vonatkozik. A többi tengely megfelelő változója indexelt címmel érhető el. A vezérlő maximum 32 tengelyt kezel.

Az

AN-nel kezdődő változók az NC-től a PLC-be mennek (Bemenetek), míg az AP-vel kezdődő változók a PLC-től az NC-be (Kimenetek).

7.13.1 A bites tengelyváltozók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
AN_DETCH	Tengely kikapcsolás nyugtázása	AP_DETCHR	Tengely kikapcsolás kérés
AN_OPNA	Tengely hurok nyitás nyugtázása	AP_OPNR	Pozíció szabályozó hurok nyitás kérés
AN_INPOS	Tengely pozícióban	AP_END	Jeladó figyelés tiltás
AN_AXALM	Hiba a tengelyen	AP_FLWU	A szervókövetés bekapcsolása
AN_RAPR	Gyorsmenet kérés	AP_JOGP	Jog + irányban kérés
AN_MTNRP	Mozgás kérés pozitív irányban	AP_JOGN	Jog - irányban kérés
AN_MTNRN	Mozgás kérés negatív irányban	AP_DECSW	Lassító kapcsolón
AN_LUBR	Kenés kérés a tengelyen	AP_RAPD	Gyorsmeneti mozgás tiltott
AN_RPE	Van érvényes Referenciapont	AP_MTNDP	Mozgás tiltás pozitív irányban
AN_REFEND	Referencia felvétel vége	AP_MTNDN	Mozgás tiltás negatív irányban
AN_OTP	Pozitív irányban végálláson	AP_LIMP	Pozitív végállás kapcsolón
AN_OTN	Negatív irányban végálláson	AP_LIMN	Negatív végállás kapcsolón
AN_REFP1	Tengely Referencia ponton	AP_LIMSELP	1B pozitív paraméteres végállás tartomány választás
AN_REFP2	Tengely Reference Position 2-ön	AP_LIMSELN	1B negatív paraméteres végállás tartomány választás
AN_REFP3	Tengely Reference Position 3-on	AP_LCK	A tengely zárva: minden mozgás tiltott
AN_REFP4	Tengely Reference Position 4-en	AP_DISPD	A tengely kijelzés kikapcsolva
AN_PARKA	Tengely parkol nyugtázás	AP_PARKR	Tengely parkol kérés
AN_SYNCA	Tengely szinkronizálás nyugtázása	AP_SYNCR	Tengely szinkronizálás kérés

7.13 A tengelykezelő változók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
AN_MIXA	Tengely csere nyugt.	AP_MIXR	Tengely csere kérés
AN_SPRPNA	Tengelyre szuperponálás nyugtázása	AP_SPRPNR	Tengelyre szuperponálás kérés
AN_EGBS	Tengely EGB slave	AP_DIARAD	Tengely átmérőben van programozva
AN_INDP	Tengely Index pozícióban	AP_SSLOP	Szolga tengely huroknyitás és csak alapjel átvétele
AN_MIXM	Tengelycsere mester tengelye	AP_FEEDD	Előtoló mozgás tiltás
AN_SYNCM	Szinkron mozgás mester tengelye	AP_PLCR	PLC elkéri a tengelyt
AN_SPRPNM	Szuperponált mozgás mester tengelye	AP_GOR	Mozgás menjen a PLC tengelyen
AN_PLCA	NC odaadta a tengelyt a PLC-nek	AP_RES	PLC tengely reset
AN_BEPTY	PLC tengely mondatpuffer üres	AP_EFD	Útmérés tiltás
AN_GOA	PLC tengely mozog	AP_RPE	Van érvényes referenciapont
AN_IEPTY	PLC tengely mozgás lejárt	AP_MIRR	Tengelyirány váltás (tükrözés) kérés
AN_MIRA	PLC által kért tengelyirány váltás (tükrözés) nyugtázása		

Az NC-től a PLC-be menő bites tengelyváltozók

AN_DETCH: Tengely kikapcsolás nyugtázása

AN_DETCH az AP_DETCHR tengely kikapcsolás kérés jel nyugtázó jelzője.
Az AP_DETCHR=1 tengely kikapcsolás kérésre a tengelykezelő AN_DETCH=1-et ad vissza. Az AP_DETCHR=0 bekapcsolás kérésre a tengelykezelő AN_DETCH=0-át ad vissza. Lásd: AP_DETCHR jelző leírását.

AN_OPNA: Tengely hurok nyitás nyugtázása

AN_OPNA az AP_OPNR pozíció szabályozó hurok nyitás kérés jel nyugtázó jelzője.
Az AP_OPNR=1 huroknyitás kérésre a pozíciószabályozó AN_OPNA=1-et ad vissza.
Az AP_OPNR=0 hurokzárás kérésre a szabályozó kör AN_OPNA=0-át ad vissza.
Lásd: AP_OPNR jelző leírását.

AN_INPOS: Tengely pozícióban

Ha a tengely parancs pozíciójának és a jeladóról mért pozíciójának a különbsége az N0516 Inpos paraméter által meghatározott ablakon belül van a jel értéke 1.

AN_AXALM: Hiba a tengelyen

Ha a tengelykezelő a szervokör bármely elemén (jeladó, EtherCAT-en keresztül működő hajtás, pozíciószabályozó kör) szervohibát észlel az AN_AXALM jelzőt 1-be állítja.

☞ **Figyelem!** A PLC program feladata a hibajelzésre reagálni, hajtást leállítani, vészállapotot kiváltani! Az NC nem kapcsolja ki a gépet automatikusan, csak a PLC parancsára!

AN_RAPR: Gyorsmenet kérés

A jelzőt az interpolátor az AN_MTNRP, AN_MTNRN mozgáskérés jelzőkkel együtt kezeli.

Ha AN_RAPR=0, az interpolátor előtolással akar mozogni, és AP_FEEDD=0 (előtoló mozgás engedélyezve) és AP_RAPD=1 (gyorsmeneti mozgás tiltott) állapotot vár.

Ha AN_RAPR=1, az interpolátor gyorsmenettel akar mozogni, és AP_FEEDD=1 (előtoló mozgás tiltott) és AP_RAPD=0 (gyorsmeneti mozgás engedélyezve) állapotot vár.

Lásd még AN_MTNRP, AN_MTNRN, AP_RAPD, AP_FEEDD jelzőket.

Ennek a jelzőnek a kezelését például előtoló és gyorsmeneti mozgások közötti áttétel-változtatásra lehet használni.

AN_MTNRP: Mozcás kérés pozitív irányban**AN_MTNRN:** Mozcás kérés negatív irányban

Minden mozgás indítása előtt az interpolátor a megfelelő irányú mozgáskérés jelzőt 1-be kapcsolja. Körinterpoláció esetén a kör síkjába eső tengelyeken, mindkét irányában kér mozgást.

Ha a PLC engedélyezi a mozgást a megfelelő irányban, az AP_MTNDP=0, vagy az AP_MTNDN=0 jelzőket kell beállítania.

Lásd még AN_RAPR, AP_RAPD, AP_FEEDD jelzőket.

A jelzők például tengelyek rögzítéséhez, oldására használhatók.

AN_LUBR: Kenés kérés a tengelyen

Ha az N1273 Lubrication Distance paraméter értéke nem 0, és az adott tengelyen az elmozdulás nagysága meghaladta a a paraméterre írt távolságot, az AN_LUBR jelző 1 PLC ciklus erejéig 1-be íródik.

A jelre pl. kenőszivattyú indítható.

AN_RPE: Van érvényes Referenciapont

Az AN_RPE=1 állapotában a tengelyen már megtörtént a referenciapont felvétele. Abszolút jeladók esetén a jelző mindig 1.

AN_REFEND: Referencia felvétel vége

A jelző referenciapont felvétel üzemmódban működik. Akkor íródik 1-be, ha az adott tengelyen megtörtént a referenciapont felvétel és a tengely már nem mozog.

AN_OTP: Pozitív irányban végálláson

AN_OTN: Negatív irányban végálláson

Ha a tengely az Axis Limits paramétercsoportban beállított végállásra, vagy végállás kapcsolóra futott, a megfelelő irány jelzője 1-be íródik.

AN_REFP1: Tengely referencia ponton

$| \text{Referenciapont} - \text{jeladóról mért gépi pozíció} | < \text{N0516 Inpos}$

Ha a tengely jeladóról mért gépi pozíciója az

N0200 Reference Position1 paraméteren megadott pozíció

N0516 Inpos paraméter által meghatározott sugarú körén belül, van a jel értéke 1.

AN_REFP2: Tengely Reference Position 2-ön

$| \text{N0201 Reference Position2} - \text{jeladóról mért gépi pozíció} | < \text{N0516 Inpos}$

Ha a tengely jeladóról mért gépi pozíciója az

N0201 Reference Position2 paraméteren megadott pozíció

N0516 Inpos paraméter által meghatározott sugarú körén belül van, a jel értéke 1.

AN_REFP3: Tengely Reference Position 3-on

$| \text{N0202 Reference Position3} - \text{jeladóról mért gépi pozíció} | < \text{N0516 Inpos}$

Ha a tengely jeladóról mért gépi pozíciója az

N0202 Reference Position2 paraméteren megadott pozíció

N0516 Inpos paraméter által meghatározott sugarú körén belül van a jel értéke 1.

AN_REFP4: Tengely Reference Position 4-en

$| \text{N0203 Reference Position4} - \text{jeladóról mért gépi pozíció} | < \text{N0516 Inpos}$

Ha a tengely jeladóról mért gépi pozíciója az

N0203 Reference Position4 paraméteren megadott pozíció

N0516 Inpos paraméter által meghatározott sugarú körén belül van a jel értéke 1.

Az AN_REFP2, AN_REFP3, AN_REFP4 jelekhez tartozó pozíciók cserehelyek bemért pozíciói. Mielőtt a PLC olyan tevékenységet végez, amelyhez egy tengely cserepozíciója szükséges a PLC a jelet vizsgálva meggyőződhet arról, hogy a tengely jó helyzetben van-e.

AN_PARKA: Tengely parkol nyugtázás

AN_PARKA az AP_PARKR parkolás kérés jel nyugtázó jelzője.

Az AP_PARKR=1 parkolás kérésre a tengelykezelő AN_PARKA=1-et ad vissza. Az AP_PARKR=0 parkolás kikapcsolás kérésre a tengelykezelő AN_PARKA=0-át ad vissza. Lásd: AP_PARKR jelző leírását.

AN_SYNCA: Tengely szinkronizálás nyugtázása

AN_SYNCA az AP_SYNCR tengely szinkronizálás kérés jel nyugtázó jelzője. A szinkronizálást kérő tengelyhez (szolga) tartozó jelzõn adja vissza a jelet.

Az AP_SYNCR=1 szinkronizálás kérésre a tengelykezelő AN_SYNCA=1-et ad vissza. Az AP_SYNCR=0 szinkronizálás kikapcsolás kérésre a tengelykezelő AN_SYNCA=0-át ad vissza. Lásd: AP_SYNCR jelző leírását.

AN_MIXA: Tengelycsere nyugtázása

AN_MIXA az AP_MIXR tengelycsere kérés jel nyugtázó jelzője. A cserélendő tengelyhez tartozó jelzõn adja vissza a jelet.

Az AP_MIXR=1 tengelycsere kérésre a tengelykezelő AN_MIXA=1-et ad vissza. Az AP_MIXR=0 tengelycsere kikapcsolás kérésre a tengelykezelő AN_MIXA=0-át ad vissza. Lásd: AP_MIXR jelző leírását.

AN_SPRPNA: Tengelyre szuperponálás nyugtázása

AN_SPRPNA az AP_SPRPNR szuperponált mozgás kérés jel nyugtázó jelzője. A szuperponálást kérő tengelyhez (szolga) tartozó jelzõn adja vissza a jelet.

Az AP_SPRPNR=1 szuperponált mozgás kérésre a tengelykezelő AN_SPRPNA=1-et ad vissza. Az AP_SPRPNR=0 szuperponált mozgás kikapcsolás kérésre a tengelykezelő AN_SPRPNA=0-át ad vissza. Lásd: AP_SPRPNR jelző leírását.

AN_EGBS: Tengely EGB slave

A jelző 1-be vált,

ha az adott tengely az N1802 EGB Slave paraméteren elektronikus hajtómű szolgálata tengelyének van kijelölve és

G81.8 hajtómű kapcsolat (fogazás) parancs van érvényben.

AN_INDP: Tengely Index pozícióban

A jelző értéke 1, ha a tengely pozíciója az indexpozíció N0516 Inpos paraméter által meghatározott sugarú körén belül van.

Indexpozíciónak nevezzük egy tengelyen a diszkrét, egymáshoz képest egyenlő távolságra lévő, valamilyen szempontból kitüntetett pozíciókat. Pl: a Hirth tárcsás körasztal, amely 5 fokonként ültethető le, 5 fokonként indexelhető.

Az N0106 Axis Properties paraméter #3 IDX=1 beállítással kijelölhetünk egy tengelyt indexelt tengelynek. Az N0110 Indexing Amount paraméteren beállíthatjuk az indexelési távolságot. Pl: a fenti, Hirth tárcsás körasztal esetén 5-t.

AN_MIXM: Tengelycsere mester tengelye

Ha a tengely tengelycsere csere tengelye a jelzőt 1-be kapcsolja. *Az AN_MIXA jelzőt a cserélendő tengelyre állítja (a két tengelyindex különböző)!*

AN_SYNCM: Szinkron mozgás mester tengelye

Ha a tengely szinkron mozgás mester tengelye a jelzőt 1-be kapcsolja. *Az AN_SYNCA jelzőt a szinkron szolga tengelyre állítja (a két tengelyindex különböző)!*

AN_SPRPNM: Szuperponált mozgás mester tengelye

Ha a tengely szuperponált mozgás mester tengelye a jelzőt 1-be kapcsolja. *Az AN_SPRPNMA jelzőt a szuperpozíciós szolga tengelyre állítja (a két tengelyindex különböző)!*

AN_PLCA: NC odaadta a tengelyt a PLC-nek

AN_PLCA az AP_PLCR tengely PLC általi elkérésének nyugtázó jelzője.

Az AP_PLCR=1 PLC tengely kérésre a tengelykezelő AN_PLCA=1-et ad vissza. Az AP_PLCR=0 tengely NC-nek való visszaadása kérésre a tengelykezelő AN_PLCA=0-át ad vissza. Lásd: AP_PLCR jelző leírását.

AN_BEPTY: PLC tengely mondatpuffer üres

A MOVCMMD PLC tengelymozgató utasításoknak van egy tengelyenkénti egymondatos puffere. Ez azt jelenti, hogy, ha az interpolátor kivett egy mondatot a pufferből végrehajtásra, a PLC betöltheti a következő mondatot. Lásd a [6.15](#) A tengelymozgató utasítás: MOVCMMD című fejezet. Valójában a MOVCMMD utasítás addig nem is hajtódik végre, amíg nem tudta a mondatot betenni a pufferbe.

Ha AN_BEPTY=1 a mondatpuffer üres.

AN_GOA: PLC tengely mozog

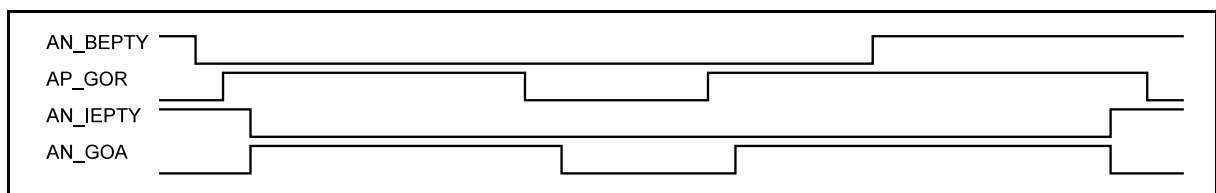
Ha a PLC tengely interpolátora nem üres, azaz AN_IEPTY=0, **ÉS** az AP_GOR=1, PLC tengely mozgást kér állaptra elindul a tengely és a jelző AN_GOA=1 állapotot vesz fel.

Ha a PLC tengely interpolátora üres, azaz AN_IEPTY=1, **VAGY** az AP_GOR=0 PLC tengely nem kér mozgást állaptra a tengely megáll és a jelző AN_GOA=0 állapotot vesz fel.

Lásd AN_GOR jelzőt.

AN_IEPTY: PLC tengely mozgás lejárt

Ha a PLC tengely mozgás lejárt, azaz az interpolátor kiürült, a jelző 1 állapotot vesz fel.

**AN_MIRA:** PLC által kért tengelyirány váltás (tükrözés) nyugtázása

A PLC program az AP_MIRR jelzőn keresztül kérheti egy tengelyen a tengelyirány váltását. Az AN_MIRA jelző a kérés nyugtázó jele.

Ilyenkor alkatrész programból való tengelymozgatás esetén az adott tengely mozgás-iránya ellentétesre vált.

A kézi mozgatások iránya (jog gombok, kézikerek) nem változik.

Ha a jelző értéke:

=0: a tengely az eredeti, paraméteren beállított irányban mozog alkatrész programból,

=1: a tengely a paraméteren beállított iránnyal ellentétes irányban mozog alkatrész programból.

Figyelem! Tengely irányváltását mindig a mondat pufferelesét megakadályozó funkcióban kell végezni (lásd Program paramétercsoport).

A PLC-től az NC-be menő bites tengelyváltozók**AP_DETCHR:** Tengely kikapcsolás kérés

Ha egy tengely működését ki kell kapcsolni, a megfelelő tengely AP_DETCHR jelzőjének 1-be írásával kérhetjük azt a rendszertől. Miután a tengelykezelő leállította a tengely működését visszaadja az AN_DETCHA=1 nyugtázó jelet.

Visszakapcsoláskor az AP_DETCHR jelzőt 0-ba kell írni, és megvárni, amíg a tengelykezelő AN_DETCHA=0 állapotot ad vissza.

Ha az AN_DETCHA jel 1 állapotban van a tengelykezelő nem működik:

törli a tengelyen a “van referenciapont” bejegyzést: AN_RPE=0,
nem méri és nem tartja nyilván a jeladóról a pozíciót,
nem zárja a pozíciószabályozó hurkot,
semmilyen jelet nem ad ki a tengely hajtása felé.,
alkatrészprogramban és PLC-ben nem lehet a tengelyre hivatkozni,

a tengely nem létezik.

Fordítva, ha az AP_DETCHR=0 és az NC kikapcsolja az AN_DETCHA=0 jelzőt a tengely újra létezik.

Egy tengely akkor létezik, ha az

N0002 Axis Assign paraméteren értéke > 0, és

AN_DETCHA=0.

☞ **Figyelem!** *Tengely ki, majd visszakapcsolását mindig a mondat pufferelését megakadályozó funkcióban kell végezni (lásd Program paramétercsoport).*

Példa:

Eszterga orsóját C tengellyé (körasztallá) kell alakítani, majd vissza orsóvá. Legyen a C tengely száma 3 (N0002 Axis Assign A3=1, N0100 Axis Name1 A3=C) az S1 orsó száma pedig 1. (N0605 Spindle Name2 S1=1)

Bekapcsolás után orsóként áll fel:

SP_SDETCHR,#0=0, SN_SDETCHA,#0=0

AP_DETCHR,#2=1, AN_DETCHA,#2=1

Ha át akarjuk kapcsolni az S1 orsót C tengellyé programozzunk egy megfelelő pufferelemnyomó M funkciót és a jelzőket a következő módon állítsuk át:

SP_SDETCHR,#0=1, SN_SDETCHA,#0=1

AP_DETCHR,#2=0, AN_DETCHA,#2=0

Ettől kezdve ugyanarról az orsóhajtásról a jeladó jeleket nem az S1 orsókezelő veszi, hanem a C tengelykezelő. Az alapjelet nem az S1 orsókezelő adja ki az orsóhajtásnak, hanem a C tengely kezelő. A C tengelyt lehet mozgatni, C címre lehet alkatrészprogramban hivatkozni.

C tengely S1 orsóvá való visszaalakítása hasonló módon történik.

AP_OPNR: Pozíció szabályozó hurok nyitás kérés

Ha egy tengely pozíciószabályozó körét fel kell nyitni, a megfelelő tengely AP_OPNR jelzőjének 1-be írásával kérhetjük azt a rendszertől. Miután a pozíciószabályozó leállította a hurok zárását visszaadja az AN_OPNA=1 nyugtázó jelet.

Visszakapcsoláskor az AP_OPNR jelzőt 0-ba kell írni, és megvárni, amíg a pozíciószabályozó AN_OPNA=0 állapotot ad vissza. A hajtás felé 0 parancsjel megy ki.

Azt, hogy a hurokzárás bekapcsolása szervokövetéssel, vagy anélkül történik az N0514 Servo Control paraméter #5 FUP bitje és az AP_FLWU jelzők együtt határozzák meg. (Lásd: AP_FLWU jel leírását)

☞ **Figyelem!** Pozíció szabályozó hurok nyitását, majd visszazárását mindig a mondat puffelését megakadályozó funkcióban kell végezni (lásd Program paramétercsoport)

AP_END: Jeladó figyelés tiltás

Ha a jelzőt 1-be kapcsoljuk a szervokör nem figyel a jeladó hibáit (nem ad Jeladó hibajelzést).

AP_FLWU: A szervókövetés bekapcsolása

A pozíciószabályozó hurok bekapcsolása esetén szabályozza a hurokzárás módját:

Ha az

N0514 Servo Control paraméter #5 FUP=0 **ÉS** a PLC jelző AP_FLWU=0 a pozíciószabályozó hurok zárása előtt a **parancspozíció felveszi** az abszolút (jeladóról mért) pozíció értékét és a tengelyen **elmozdulás nem történik**,. A gép az elmozdult pozícióban marad mindaddig, amíg abszolút mozgást nem programoztunk.

Ha az

N0514 Servo Control paraméter #5 FUP=1 **VAGY** a PLC jelző AP_FLWU=1 a pozíciószabályozó hurok zárása előtt a **parancspozíció nem veszi fel** az abszolút (jeladóról mért) pozíció értékét és a **tengelyen elmozdulás történik**, lelépi a szervó kikapcsolt állapotában felhalmozott elmozdulást.

AP_JOGP: Jog + irányban kérés

AP_JOGN: Jog - irányban kérés

Ha Jog, vagy Inkrementális jog üzemben a jelzőt 1-be kapcsoljuk a tengely a megfelelő (+/-) irányban mozog, ha a jelzőt 0-ba kapcsoljuk a tengely megáll.

Az MB_JOGn gombok köthetők ezekre a jelekre.

Ha referenciapontfelvétel üzemben, bármelyik AP_JOGP, vagy AP_JOGN jelzőt 1-be kapcsoljuk a tengely nem a jog irány, hanem az N0901 Reference Distance és az N0900 Reference Type #5 DIR paraméterek által meghatározott irányban mozog, addig, amíg valamelyik jelző 1-ben van, megáll, ha egyik jelző sincs 1-ben, vagy megtörtént a referenciapont felvétele.

AP_DECSW: Lassító kapcsolón

Ha a tengelyen az N0900 Reference Type #0 SWT=1 paraméterállás van beállítva, a referenciapont felvétele kapcsolóra futással történik.

Ha az AP_DECSW jelző értéke 1, a tengely ráfutott a referenciapont kapcsolóra.

AP_RAPD: Gyorsmeneti mozgás tiltott

Az AN_RAPR gyorsmeneti mozgáskérés jelzőre adott válaszjel.

Ha AN_RAPR=1, a gyorsmeneti mozgás akkor indul el, ha a PLC az AP_RAPD=0 jelet beállította.

Ha AP_RAPD=1 a gyorsmeneti mozgás nem megy.

A jelzőt az AP_MTNDP, AP_MTNDN mozgástiltás jelzőkkel együtt kell kezelni.

Ennek a jelzőnek a kezelését például előtoló és gyorsmeneti mozgások közötti áttétel-változtatásra lehet használni.

AP_MTNDP: Mozcás tiltás pozitív irányban

AP_MTNDN: Mozcás tiltás negatív irányban

Az AN_MTNRP és az AN_MTNRN mozgáskérés jelzőkre adott válaszjelek.

Minden mozgás indítása előtt az interpolátor a megfelelő irányú mozgáskérés jelzőt 1-be kapcsolja (AN_MTNRP=1, AN_MTNDN=1). Kőrinterpoláció esetén a kőr síkjába eső tengelyeken, mindkét irányában kér mozgást.

A tengely addig nem mozog, amíg a PLC nem engedélyezi a mozgást a megfelelő irányban, az AP_MTNDP=0, vagy az AP_MTNDN=0 jelzőállással.

A tengely megáll, ha a megfelelő irányú mozgástiltás jelzőt a PLC 1-be kapcsolja.

Lásd még AN_RAPR, AP_RAPD, AP_FEEDD jelzőket.

A jelzők például tengelyek rögzítéséhez, oldására használhatók.

AP_LIMP: Pozitív végállás kapcsolón

AP_LIMN: Negatív végállás kapcsolón

Ha a tengely valamelyik irányban ráfutott a végálláskapcsolóra a megfelelő jelzőt 1-be kell írni: AP_LIMP=1, vagy AP_LIMN=1. A jelző beállítása után az interpolátor lassítással megáll és csak a másik irányban lehet elindítani. A megálláshoz szükséges lassítási utat a kapcsoló beállításánál figyelembe kell venni.

AP_LIMSELP: 1B pozitív paraméteres végállás tartomány választás

AP_LIMSELN: 1B negatív paraméteres végállás tartomány választás

Tengelyenkénti és irányonkénti paraméteres végállásválasztás PLC-ből.

Ha az

N1001 StrkCont paraméter #4 ABA=0

az adott csatornában a tengelyenkénti és irányonkénti paraméterválasztást lehetővé tevő PLC jelzők hatásosak: AP_LIMSELP pozitív irányban AP_LIMSELN negatív irányban, az 1A és 1B tartomány közül választ.

Pozitív irányban

AP_LIMSELP=0: N1002 Range1A Positive paramétert,

AP_LIMSELP=1: N1004 Range1B Positive paramétert

választ.

Negatív irányban

AP_LIMSELN=0: N1003 Range1A Negative paramétert,

AP_LIMSELN=1: N1005 Range1B Negative paramétert

választ.

A végállásválasztás változtatását a tengelyek álló állapotában kell végezni.

☞ **Lásd még:** CP_LIMSEL jelzőt.

AP_LCK: A tengely zárva: minden mozgás tiltott

Az AP_LCK=1 állapotban az interpolátor nem ad ki mozgásparancsot a tengely felé. A pozíciókijelzőn látszik a mozgás.

Ha AP_LCK=0-ra vált, az interpolátor a mérőrendszerből frissíti a pozíciókat és a tengely újra mozgatható.

A jelzőt változtatni csak akkor szabad, ha nem fut alkatrészprogram és a tengely áll!

AP_DISPD: A tengely kijelzés kikapcsolva

Ha egy tengely N0002 Axis Assign paramétere nem 0 és az N0100 Axis Name1, ... paramétereken nevet adtunk neki, a tengely automatikusan megjelenik a pozíciókijelzőn.

Ha a jelző AP_DISPD=1, az adott tengely eltűnik a pozíciókijelzőről.

Például, ha egy tengelyt az AP_DETCHR jelzővel kikapcsolunk a kijelzését is ki lehet kapcsolni.

AP_PARKR: Tengely parkol kérés

Parkolást csak szinkron mozgásban részt vevő tengelyre lehet kérni, azaz ha a tengely vagy

AN_SYNCA=1: szinkron szolga , vagy

AN_SYNCM=1: szinkron mester.

A parkolást a parkolandó tengely kezdeményezi a hozzá tartozó AP_PARKR parkolás kérés jelző beírásával. A parkolás akkor jön létre, ha a tengelykezelő 1-be írta az

AN_PARKA nyugtázó jelet. A parkolás kikapcsolásakor az AP_PARKR jelet 0-ba kell írni, és megvárni, hogy a tengelykezelő visszaadja az AN_PARKA=0 állapotot. Parkolás az amikor egy alkatrészprogram szinkron működtetéssel van megírva, de vagy a szinkron szolgál, vagy a szinkron mester tengelyt nem akarjuk mozgatni, mert például azon az oldalon nincs darab. Ekkor vagy kapcsolóval, vagy M funkcióval bekapcsolhatjuk a megfelelő tengelyen a parkolást. Lásd még az AP_SYNCR jelet.

☞ **Figyelem!** *Parkolás be-, majd kikapcsolását mindig a mondat pufferelesét megakadályozó funkcióban kell végezni (lásd Program paramétercsoport). A nyugtázó jel addig nem jön meg, amíg a puffer nem üres. Parkolást kapcsolóval működtetni csak akkor szabad, amikor nem fut program és a tengely nyugalomban van.*

AP_SYNCR: Tengely szinkronizálás kérés

A szinkron mozgatást a szinkron szolgál tengely kezdeményezi a hozzá tartozó AP_SYNCR szinkronizálás kérés jelző beírásával. A szinkron kapcsolat akkor jön létre, ha a tengelykezelő 1-be írta az AN_SYNCA nyugtázó jelet. A szinkron mozgatás kikapcsolásakor az AP_SYNCR jelet 0-ba kell írni, és megvárni, hogy a tengelykezelő visszaadja a AN_SYNCA=0 állapotot.

Szinkron megmunkálásnak nevezzük, amikor egy tengely, a szinkron szolgál, egy másik tengely, a szinkron mester mozgásparancsai hatására mozog. Ebben az állapotban a szinkron szolgára mozgásparancsot kiadni nem lehet. A szinkron tengelymozgatást alkatrészprogramból pl. M funkció hatására be, illetve ki lehet kapcsolni.

A szinkron szolgál mester tengelyének számát az N2101 Synchronous Master paraméterben a szinkron szolgálhoz tartozó tengelyszámra írjuk be. Ha a paraméter értéke 0, a tengely nem szinkron szolgál. Például, ha a 4. tengely szinkron szolgál és a mester tengelyének száma 2, a paramétert a következőképp kell kitölteni:

N2101 Synchronous Master A4=2

A szinkron szolgál elmozdulásának irányát a mesterhez képest az N2102 Synchron Config paraméter #0 MSY bitje határozza meg.

☞ **Figyelem!** *A szinkron tengelymozgatás nem tévesztendő össze a pozíciószinkronnal, amikor pl. állványos gépen (gantry) két motor mozgatja ugyanazt a tengelyt. Ekkor a N2102 Synchron Config paraméter #4 PSN=1, míg szinkron mozgásnál PSN=0.*

☞ **Figyelem!** *Szinkron megmunkálás be-, majd kikapcsolását mindig a mondat pufferelesét megakadályozó funkcióban kell végezni (lásd Program paramétercsoport). A nyugtázó jel addig nem jön meg, amíg a puffer nem üres.*

AP_MIXR: Tengely csere kérés

A tengelycserét a cserélendő tengely kezdeményezi a hozzá tartozó AP_MIXR csere kérés jelző beírásával. A csere akkor jön létre, ha a tengelykezelő 1-be írta az AN_MIXA nyugtázó jelet. A tengelycsere kikapcsolásakor az AP_MIXR jelet 0-ba kell írni, és megvárni, hogy a tengelykezelő visszaadja a AN_MIXA=0 állapotot.

Tengelycsere esetén a cserélendő tengely címén (N0100 Axis Name1, ...paramétereken meghatározott címek) egy másik számú tengelyt, a cseretengelyt lehet mozgatni és fordítva, a cseretengely címén a cserélendő tengely mozog. A tengelycsereát alkatrészprogramból pl. M funkció hatására be, illetve ki lehet kapcsolni. *A tengelycsere csak az alkatrészprogramból a tengely címére történő hivatkozásra vonatkozik, a jog és kézikerék mozgásokra nem, mert ezek tengelyszámokra vonatkoznak, ezért ezeket az eseteket a PLC programból kell kezelni!*

Az cserélendő tengelyhez tartozó N2104 Composit Axis paraméter mondja meg, hogy a cserélendő tengelyt hányadik tengellyel kell felcserélni. Ha a paraméter értéke 0, a tengely nem cserélendő. Például, ha a 4. tengely cserélendő tengely és a csere tengelyének száma 1, a paramétert a következőképp kell kitölteni:

N2104 Composit Axis A4=1

Tengelycsereénél mindkét tengely dolgozhat másik csatornában is (nem abban, amelyikre az Axis Assign paraméter kijelölte), onnan kapja a mozgásparancsokat és a másik csatornából kapja az eredeti tengely nullponteltolását. Csere után mindkét csatornában lehet a tengelyeket mozgatni, kivéve, ha az egyik tengely hipotetikus. Ha egy tengely hipotetikusnak van kijelölve egy adott csatornában (N0106 Axis Properties #7 HYP=1), a csere után a másik csatornában, vagy ugyanabban, nem lehet rá hivatkozni.

Azt, hogy a cserélendő, illetve a cseretengely az eredetihez képest ugyanabba az irányba, vagy fordítva mozogjon az N2105 Composit Config #0 MMI paramétere határozza meg.

 **Figyelem!** *Tengelycsere be-, majd kikapcsolását mindig a mondat puffrelését megakadályozó funkcióban kell végezni (lásd Program paramétercsoport). A nyugtázó jel addig nem jön meg, amíg a puffér nem üres.*

AP_SPRPNR: Tengelyre szuperponálás kérés

A szuperponált mozgást a szuperpozíciós szolga tengely kezdeményezi a hozzá tartozó AP_SPRPNR szuperponálás kérés jelző beírásával. A szuperpozíciós kapcsolat akkor jön létre, ha a tengelykezelő 1-be írta az AN_SPRPNA nyugtázó jelet. A szuperpozíciós mozgás kikapcsolásakor az AP_SPRPNR jelet 0-ba kell írni, és megvárni, hogy a tengelykezelő visszaadja a AN_SPRPNA=0 állapotot.

Szuperpozíciós megmunkálásnak nevezzük, amikor egy tengely, a szuperpozíciós szolga, két mozgásparancs hatására mozog: a saját címén kapott elmozdulások és a szuperpozíciós mestertől kapott elmozdulások összeadódnak és így alakul ki a mozgás. A szolga és a mester tengely lehetnek ugyanabban, de különböző csatornában is. A szuperpozíciós tengelymozgást alkatrészprogramból pl. M funkció hatására be, illetve ki lehet kapcsolni.

A szuperpozíciós szolga mester tengelyének számát az N2107 Superimposed Master paraméterben a szuperpozíciós szolgálhoz tartozó tengelyszámra írjuk be. Ha a paraméter értéke 0, a tengely nem szuperpozíciós szolga. Például, ha a 4. tengely

szuperpozíciós szolga és a mester tengelyének száma 2, a paramétert a következőképp kell kitölteni:

N2107 Superimposed Master A4=2

A szuperpozíciós szolga elmozdulásának irányát a mesterhez képest az N2108 Superimposed Config paraméter #0 MSU bitje határozza meg.

☞ **Figyelem!** Szuperpozíciós mozgás be-, majd kikapcsolását mindig a mondat puffereelését megakadályozó funkcióban kell végezni (lásd Program paramétercsoport). A nyugtázó jel addig nem jön meg, amíg a puffer nem üres.

AP_DIARAD: Tengely átmérőben van programozva

A jelző állapotát akkor veszi figyelembe a rendszer, ha az N0106 Axis Properties paraméter #5 MGD bitjének értéke =0.

Egy adott tengelyen az adatbevitel és pozíciókijelzés módját szabályozza:

Ha az

N0106 Axis Properties paraméter #0 DIA=0 **ÉS** a PLC jelző AP_DIARAD=0 az adott tengelyen az adatbevitel és pozíciókijelzés **sugárban** történik.

Ha az

N0106 Axis Properties paraméter #0 DIA=1 **VAGY** a PLC jelző AP_DIARAD=1

az adott tengelyen az adatbevitel és pozíciókijelzés **átmérőben** történik.

Az adatbevitel az alkatrészprogramra, korrekciókra és nullponteltolásokra terjed ki.

☞ **Figyelem!** A sugárból átmérőbe, vagy átmérőből sugárba történő átkepcsolást mindig a mondat puffereelését megakadályozó funkcióban kell végezni (lásd Program paramétercsoport).

AP_SSLOP: Szolga tengely huroknyitás és csak alapjel átvétele

Ha a tengely **nem szinkron szolga**, hatása megegyezik az AP_OPNR jelével.

Ha a tengely **szinkron szolga**, azaz a tengelyen AN_SYNCA=1, nyitja a pozíciószabályozó hurkot, a szolga tengely átveszi a mester tengely sebesség parancsjelét és azt küldi ki a szolga hajtásának.

Példa:

A főorsó és az ellenorsó is egyben Z tengely, azaz mindkettő mozgatható. Ha egy hosszú tengelyt kívánunk megmunkálni, az ellenorsóval támaszthatjuk a darabot. Az ellenorsó Z tengelyét jelöljük ki a főorsó Z tengely szinkron szolgájának és kapcsoljuk össze a két mozgást (AP_SYNCR=1). Ezután, valahányszor a fő- és az ellenorsón is zárva a tokmány, a túlhatározottságot elkerülendő, az AP_SSLOP=1 beállítással nyissuk a szolga tengelyen a pozíciószabályozó hurkot. A főorsó Z tengelye mozgatható, a szolga átveszi a mestertől a sebesség parancsjeleket és együtt mozognak. Lásd még DP_SILCK hajtásjelet.

AP_FEEDD: Előtoló mozgás tiltás

Az AN_RAPR gyorsmeneti mozgáskérés jelzőre adott válaszjel.

Ha AN_RAPR=0, az előtoló mozgás akkor indul el, ha a PLC az AP_FEEDD=0 jelet beállította.

Ha AP_FEEDD=1 az előtoló mozgás nem megy.

A jelzőt az AP_MTNDP, AP_MTNDN mozgástiltás jelzőkkel együtt kell kezelni.

Ennek a jelzőnek a kezelését például előtoló és gyorsmeneti mozgások közötti áttétel-változtatásra lehet használni.

AP_PLCR: PLC elkéri a tengelyt

Bármely, az N0002 Axis Assign paraméteren megadott tengelyt bekapcsolás után az NC kezel. A PLC bármely tengelyt elkérhet az NC-től és a [6.15](#) A tengelymozgató utasítás: MOVCMMD című fejezetben leírtak szerint mozgathat.

A tengelyt a PLC az AP_PLCR jelző 1-be írásával kérheti el. Mindig meg kell várni, hogy a tengelykezelő visszaadja az AN_PLCA=1 állapotot. Fordítva, amikor a PLC visszaadja a tengelyt az AP_PLCR jelzőt 0-ba írja és meg kell várnia az AN_PLCA=0 állapotot.

Figyelem! A PLC által történő tengely elkérést és visszaadást mindig a mondat pufferelesét megakadályozó funkcióban kell végezni (lásd Program paramétercsoport). A nyugtázó jel addig nem jön meg, amíg a puffer nem üres.

AP_GOR: Mozdás menjen a PLC tengelyen

Ha a PLC tengely interpolátora nem üres, azaz AN_IEPTY=0, az AP_GOR=1 jelző állapotra elindul a mozgás, míg az AP_GOR=0 hatására megáll. Arról, hogy a mozgás megy-e az AN_GOA jelző tájékoztat. Lásd AN_GOA jelzőt.

AP_RES: PLC tengely reset

Ha az AP_RES jel 1-be megy:

a PLC tengely lassítással megáll,

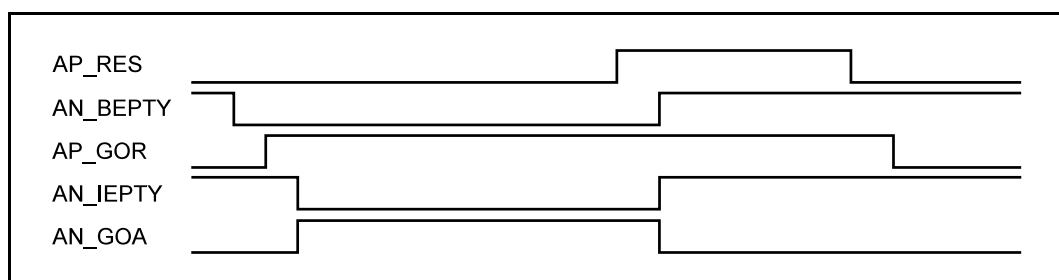
törli az aktuálisan végrehajtás alatt álló parancsot,

törli a pufferben lévő parancsot, és beállítja a következő jelzőket:

AN_BEPTY=1

AN_GOA=0

AN_IEPTY=1



AP_EFD: Útmérés tiltás

A jelző csak a pozíciószabályozó hurok nyitott állapotában (AN_OPNA=1) hatásos. Ha a jelző értéke 1, a tengelykezelő a jeladó impulzusairól nem frissíti a tengely pozícióját, befagyasztja annak értékét addig, amíg a jelző újra 0-ra nem vált. Ettől kezdve a befagyasztott pozíciótól számolja a jeladóról jövő impulzusokat. Ez a funkció akkor használható, ha több tengely ugyanazt a jeladót használja útmérésre. A PLC jelző kapcsolatban van az N0514 Servo Control paraméter #4 EFD bitjével.

AP_RPE: Van érvényes referenciapont

Ha a pozíciószabályozást nem az NC végzi, hanem a hajtás, akkor a referenciapont felvételt is a hajtás hajtja végre. A referenciapont meglétét a PLC program a hajtás kontrol bitjeinek kiolvasása után az AP_RPE jelző bekapcsolásával közli az NC-vel. Ezt az esetet az N0900 Reference Type paraméteren a #3 ABS=1 és #4 FLO=1 állítással adjuk meg.

AP_MIRR: Tengelyirány váltás (tükrözés) kérés

A PLC program az AP_MIRR jelzőn keresztül kérheti egy tengelyen a tengelyirány váltását. Az AN_MIRA jelző a kérés nyugtázó jele, amit mindig meg kell várni.

Ilyenkor alkatrész programból való tengelymozgatás esetén az adott tengely mozgásiránya ellentétesre vált.

A kézi mozgatások iránya (jog gombok, kézikerek) nem változik.

Ha a jelző értéke:

=0: a tengely az eredeti, paraméteren beállított irányban mozogjon alkatrész programból,

=1: a tengely a paraméteren beállított iránnyal ellentétes irányban mozogjon alkatrész programból.

☞ **Figyelem!** *Tengely irányváltását mindig a mondat puffereklését megakadályozó funkcióban kell végezni (lásd Program paramétercsoport).*

7.14 Az orsókezelő változók

Az orsókezelő változók olyan az NC-től a PLC-be, vagy a PLC-ből az NC-be menő változók, amelyek orsónként indexelődnek. Az összes itt közölt szimbólum az első (0-ás indexű) orsóra vonatkozik. A többi orsó megfelelő változója indexelt címmel érhető el. A vezérlő maximum 16 orsót kezel.

Az

SN-nel kezdődő változók az NC-től a PLC-be mennek (Bemenetek), míg az SP-vel kezdődő változók a PLC-től az NC-be (Kimenetek).

7.14.1 A bites orsóváltozók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
SN_RMPD	Orsón alapjel felfutott	SP_SEN	Jelkiadás az orsón engedélyezve
SN_NS	Orsó jeladóról mért fordulata elérte a fordulatszámparancsot N=Ns	SP_SSTRT	Orsó start
SN_N0	Orsó jeladóról mért fordulatszám 0 (N=0)	SP_PAR	Orsó alapjelkiadás paraméterről
SN_FLU	Orsó fordulatszáma ingadozik	SP_NEG	Orsó forgásirány M4 (negatív alapjel)
SN_FLOFF	Orsó fordulatszám ingadozás figyelés kikapcsolva	SP_OREQ	Orsón orientálás kérés
SN_LPCLSD	Az orsón a pozíciószabályozó hurok zárva.	SP_OSHRT	Orsó orientálás rövidebb úton (ha =0: a NEG előjele szerint orientál)
SN_ORIP	Orsó pozíciószabályozó hurok zárva, és orientálási pozíción	SP_SSYNCR	Orsó szinkronizálás kérés
SN_SINPOS	Orsó pozícióban	SP_PHSHFTR	Fázistolás kérés szinkronizáláskor
SN_SSYNA	Szolga orsó szinkronizálva	SP_POLYR	Nem használt
SN_PHSHFTR	Fázistolás megtörtént a szolga orsón	SP_SEND	Orsón jeladó figyelés tiltás
SN_SYNCPOS	Nem használt	SP_SMTNDP	Orsótengely mozgás tiltás pozitív irányban
SN_POLYA	Sokszögessztergáláshoz szinkronizálás a szolga orsón megtörtént	SP_SMTNDN	Orsótengely mozgás tiltás negatív irányban
SN_SMTNRP	Orsó tengely mozgás kérés pozitív irányban	SP_SDETCR	Orsókezelő kikapcsolás kérés
SN_SMTNRN	Orsó tengely mozgás kérés negatív irányban	SP_SLCLR	Hurok zárás kérés orientálás nélkül

7.14 Az orsókezelő változók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
SN_SRAPR	Gyorsmenet kérés	SP_SSROFF	Szolga orsón pozíció hurok nyitás
SN_SDETHA	Orsókezelő kikapcsolás nyugtázva	SP_SDISPD	Orsó kijelzés tiltása a képernyőn
SN_SALM	Hiba az orsón	SP_FEEDD	Orsó tengely előtoló mozgás tiltás
SN_RPE	Van érvényes referenciapont az orsón	SP_RAPD	Orsó gyorsmeneti mozgás tiltás
SN_SINDP	Az orsó indexpozíción	SP_TLCHIA	A szerszámcsere kérés nyugtázó jele
SN_TLCHI	Az orsóban lévő szerszám éltartama lejárt, szerszámcsere kérés	SP_TLCHA	A csoportcsere kérés nyugtázó jele
SN_TLCH	A teljes szerszámcsoporthoz éltartama lejárt, csoportcsere kérés	SP_TLSKP	Orsóban lévő szerszám törött
SN_TLSKPA	Szerszámtörés bejegyzés nyugtázása	SP_TLCD	Éltartamszámláló tiltása
SN_TLNL	Az orsóban lévő szerszám figyelmeztető éltartama lejárt	SP_OSW	Orsó az orientálási kapcsolón van

Az NC-től a PLC-be menő bites orsóváltozók

SN_RMPD: Orsón alapjel felfutott

Az orsókezelő a hajtásnak kiadott alapjelet, ha a fordulatszámot abszolút értékben növelni kell az N0665+n Rn S Ramp Up paraméter alapján felfuttatja, ha a fordulatszámot abszolút értékben csökkenteni kell az N0673+n Rn S Ramp Down paraméter alapján lefuttatja, ahol “n” a beváltott tartomány száma.

Amint az alapjel fel-, lefuttatást befejezte, az orsókezelő az SN_RMPD jelzőt 1-be állítja.

SN_NS: Orsó jeladóról mért fordulata elérte a fordulatszámparancsot N=N_s

Ha az orsó jeladóról mért fordulata az N0627 S N% és az N0628 S NW paraméterek meghatározta tűrésen belül elérte a kiadott fordulatszámparancsot, az orsókezelő az SN_NS jelzőt 1-be állítja.

Ha a fordulatszám abszolút értékben jó, de az SN_NS jel nem jön meg, az N0609 Spindle Encoder Config paraméterben meg kell fordítani a jeladó irányát (#1 ID, vagy #2 AD paraméteren), feltéve, hogy az orsó nem unipoláris: #5 UNI=0.

SN_N0: Orsó jeladóról mért fordulatszám 0 (N=0)

Ha az orsó jeladóról mért fordulata az N0629 S N0 paraméteren meghatározott tűrésen belül elérte a 0 értéket, az orsókezelő az SN_N0 jelzőt (és az SN_NS jelzőt is) 1-be állítja.

SN_FLU: Orsó fordulatszáma ingadozik

Ha az orsó jeladóról mért fordulata az N0632 S Fluct% és az N0633 S FluctW paraméterek meghatározta tűrésen kívül esik, az orsókezelő az SN_FLU jelzőt 1-be állítja.

SN_FLOFF: Orsó fordulatszám ingadozás figyelés kikapcsolva

Ha alkatrészprogramból G25 utasítással kikapcsoljuk az orsó fordulatszám ingadozás figyelését, az SN_FLOFF jelző 1 állapotot vesz fel.

SN_LPCLSD: Az orsón a pozíciószabályozó hurok zárva.

Ha az orsón a pozíciószabályozó hurok zárva van a jelzőt az orsókezelő 1-be írja. Ezek az esetek a következők:

- SP_OREQ orientálás, vagy SP_SLCLR hurokzárás parancs végrehajtása után,
- ha az orsó két orsó szinkronizálása után vagy szinkron mester, vagy szinkron szolga,
- ha az orsó G51.2 sokszögesztergálás mester, vagy szolga orsója,
- ha az orsó G81.8 elektronikus hajtómű bekapcsolása után mester orsó (fogaskerekek lefejtő marása).

SN_ORIP: Orsó pozíciószabályozó hurok zárva, és orientálási pozíción

Az orsókezelő a jelzőt 1-be írja, ha az orsón a pozíciószabályozó hurok zárva és az orsó a jeladó nullimpulzusától számítva az N0684 Spindle Grid Shift paraméteren megadott távolság N0743+n Rn S Inpos paraméteren megadott környezetében van. Ha N0684 Spindle Grid Shift=0, a nullimpulzus N0743+n Rn S Inpos paraméteren megadott környezetében.

SN_SINPOS: Orsó pozícióban

Az orsókezelő a jelzőt 1-be írja,

- SP_OREQ orientálás, vagy SP_SLCLR hurokzárás parancs végrehajtása után, ha az orsóra kiadott parancs pozíció és a jeladóról mért pozíció különbségének abszolút értéke a N0743+n Rn S Inpos paraméteren megadott értéknél kisebb.
- Ha az orsó két orsó szinkronizálása után *szinkron szolga*, vagy
- ha az orsó G51.2 *sokszögesztergálás szolga* orsója,
és a mester és szolga orsók ciklikus pozíciói különbségének (nullimpulzusainak pozíciókülönbsége az eltolásokat is figyelembe véve) abszolút értéke a N0743+n Rn S Inpos paraméteren megadott értéknél kisebb.

SN_SSyna: Szolga orsó szinkronizálva

Az SP_SSYNCR szinkronizálás kérés nyugtázó jelzője.

Ha egy orsón SP_SSYNCR paranccsal a PLC szinkronizálást kér egy másik orsóhoz, és a szinkronizálás megtörtént, az orsókezelő a jelzőt 1-be írja.

SN_PSHFTA: Fázistolás megtörtént a szolga orsón

Az SP_PSHFTR fázistolás kérés nyugtázó jelzője.

Ha a PLC a szinkronizálást fázistolással kéri SP_PSHFTR=1 feltétellel, és a szinkronizálás és fázistolás is megtörtént az orsókezelő az SN_PSHFTA jelzőt 1-be írja.

SN_SYNCPOS: Nem használt

SN_POLYA: Sokszögesztergáláshoz szinkronizálás a szolga orsón megtörtént

Ha az orsó G51.2 sokszögesztergálás szolga orsója, és a mesterorsóhoz történő szinkronizálás megtörtént az orsókezelő az SN_POLYA jelzőt 1-be írja.

SN_SMTNRP: Orsó tengely mozgás kérés pozitív irányban

SN_SMTNRN: Orsó tengely mozgás kérés negatív irányban

Az orsó indexelése esetén minden mozgás indítása előtt az interpolátor a megfelelő irányú mozgáskérés jelzőt 1-be kapcsolja.

Ha a PLC engedélyezi a mozgást a megfelelő irányban, az SP_SMTNDP=0, vagy az SP_SMTNDN=0 jelzőket kell beállítania.

Lásd még SN_SRAPR, SP_RAPD, SP_FEEDD jelzőket.

A jelzők például az orsók rögzítésére, oldására használhatók.

SN_SRAPR: Gyorsmenet kérés

A jelzőt az interpolátor az SN_SMTNRP, SN_SMTNRN mozgáskérés jelzőkkel együtt kezeli.

Ha SN_SRAPR=0, az interpolátor előtolással akar mozogni, és SP_FEEDD=0 (előtoló mozgás engedélyezve) és SP_RAPD=1 (gyorsmeneti mozgás tiltott) állapotot vár.

Ha SN_SRAPR=1, az interpolátor gyorsmenettel akar mozogni, és SP_FEEDD=1 (előtoló mozgás tiltott) és SP_RAPD=0 (gyorsmeneti mozgás engedélyezve) állapotot vár.

Lásd még SN_SMTNRP, SN_SMTNRN, SP_RAPD, SP_FEEDD jelzőket.

Ennek a jelzőnek a kezelését például előtoló és gyorsmeneti mozgások közötti áttétel-változtatásra lehet használni.

SN_SDETCHA: Orsókezelő kikapcsolás nyugtázva

SN_SDETCHA az SP_SDETCR orsó kikapcsolás kérés jel nyugtázó jelzője.

Az SP_SDETCR=1 orsó kikapcsolás kérésre az orsókezelő SN_SDETCR=1-et ad vissza. Az SP_SDETCR=0 bekapcsolás kérésre az orsókezelő SN_SDETCR=0-át ad vissza. Lásd: SP_SDETCR jelző leírását.

SN_SALM: Hiba az orsón

Ha az orsókezelő, az orsó bármely elemén (jeladó, EtherCAT-en keresztül működő hajtás, pozíciószabályozó kör) szervohibát észlel az SN_SALM jelzőt 1-be állítja.

Figyelem! A PLC program feladata a hibajelzésre reagálni, hajtást leállítani, vészállapotot kiváltani! Az NC nem kapcsolja ki a gépet automatikusan, csak a PLC parancsára!

SN_RPE: Van érvényes referenciapont az orsón

Az SN_RPE=1 állapotában az orsón már megtörtént a referenciapont felvétele (nullimpulzus helyzete ismert). Abszolút jeladók esetén a jelző mindig 1.

SN_SINDP: Az orsó indexpozíción

A jelző értéke 1, ha az orsó pozíciója az indexpozíció N0743+n Rn S Inpos paraméter által meghatározott sugarú körén belül van.

Indexpozíciónak nevezzük egy orsón a diszkrét, egymáshoz képest egyenlő távolságra lévő, valamilyen szempontból kitüntetett pozíciókat. Pl: ha az orsó rögzítéséhez egy rudat be lehet tolni egy, az orsóhoz erősített tárcsába 5 fokonként, akkor az orsó 5 fokonként indexelhető.

Az N0822 Basic Angle of Spnd. Pos. paraméteren beállíthatjuk az indexelési távolságot. Pl: a fenti orsórögzítés esetén 5-t.

SN_TLCHI: Az orsóban lévő szerszám éltartama lejárt, szerszámcsere kérés

A jelzőt az NC 1-be kapcsolja, ha az N2900 Tool M. Config paraméter #0 TMU=1 és az adott orsóban lévő szerszám

- éltartama lejárt, vagy
- a PLC bekapcsolja rá a törött SP_TLSKP jelzőt.

Az SN_TLCHI jelző bekapcsolva marad addig, amíg a PLC az SP_TLCHIA nyugtázó jelet be nem kapcsolja. Miután az NC az SN_TLCHI jelet törölte, a PLC-nek is törölnie kell a nyugtázó jelet.

Az SN_TLCHI jelző resetre *nem törlődik*.

SN_TLCH: A teljes szerszámcsoporthoz éltartama lejárt, csoportcsere kérés

A jelzőt az NC 1-be kapcsolja, ha az N2900 Tool M. Config paraméter #0 TMU=1 és a csoport (az azonos típuskódú szerszámok) összes szerszámára igaz, hogy

- az éltartama “Lejárt”, vagy
- “Törött”.

Az SN_TLCH jelző bekapcsolva marad addig, amíg a PLC az SP_TLCHA nyugtázó jelet be nem kapcsolja. Miután az NC az SN_TLCH jelet törölte, a PLC-nek is törölnie kell a nyugtázó jelet.

Az SN_TLCH jelző resetre *nem törlődik*.

SN_TLSKPA: Szerszámtörés bejegyzés nyugtázása

Ha a PLC szerszámtörést detektál és 1-be írja az SP_TLSKP jelzőt, az NC a táblázatban a szerszámmra bejegyzi a “törött” állapotot, majd 1-be írja az SN_TLSKPA nyugtázó jelet, ha az N2900 Tool M. Config paraméter #0 TMU=1.

Ezután kiadja az SN_TLCHI szerszámcseré kérés jelzőt.

SN_TLNL: Az orsóban lévő szerszám figyelmeztető élettartama lejárt

Ha az orsóban lévő szerszám figyelmeztető élettartama lejárt az NC 1 PLC ciklus elejéig a jelzőt 1-be írja, ha az N2900 Tool M. Config paraméter #0 TMU=1.

A PLC-től az NC-be menő bites orsóváltozók

SP_SEN: Jelkiadás az orsón engedélyezve

Valahányszor a PLC az orsóra valamilyen parancsot ad ki (forgatás, orientálás szinkronizálás, sokszögesztergálás), az SP_SEN jelző 1-be írásával engedélyeznie kell a jelkiadást.

Az orsó leállítása után a jelzőt ki kell kapcsolni.

SP_SSTRT: Orsó start

Az **SP_SSTRT=1** hatására az orsókezelő a beállított fordulatszámmal forgatja az orsót.

Az **SP_SSTRT=0** hatására az orsókezelő leállítja az orsó forgását.

Lásd még SP_PAR, SP_NEG jelzőket és SP_PRG regisztert.

SP_PAR: Orsó alapjelkiadás paraméterről

Az **SP_PAR=0** állapotban, SP_SSTRT=1 hatására az orsókezelő az SP_PRG regiszterben megadott fordulatszámmal forgatja az orsót.

Az **SP_PAR=1** állapotban, SP_SSTRT=1 hatására az orsókezelő az N0657+n Rn S Jog Speed paraméteren beállított fordulatszámmal forgatja az orsót.

SP_NEG: Orsó forgásirány M4 (negatív alapjel)

Az **SP_NEG=0** állapotban, SP_SSTRT=1 hatására az orsókezelő az orsót pozitív, M3 irányban forgatja.

Az **SP_NEG=1** állapotban, SP_SSTRT=1 hatására az orsókezelő az orsót negatív, M4 irányban forgatja.

Ha az orsó forgásiránya nem jó, az N0609 Spindle Encoder Config paraméterben meg kell fordítani #0 MD paraméteren a motor forgásirányát. (Ne az SP_NEG invertálásával fordítsunk forgásirányt.)

Az **SP_NEG=0** állapotban, SP_SSYNCR=1 szinkronizálás kérés hatására az orsókezelő az orsót az SP_MAST regiszterben megadott orsó forgásirányával megegyező irányban forgatja.

Az **SP_NEG=1** állapotban, SP_SSYNCR=1 szinkronizálás kérés hatására az orsókezelő az orsót az SP_MAST regiszterben megadott orsó forgásirányával ellentétes irányban forgatja. (Ellenorsó szinkronizálása.)

SP_OREQ: Orsón orientálás kérés

Az **SP_OREQ=1** hatására az orsókezelő

- ha az orsó forog,
 - lelassítja az orsó forgását az N0800+n Rn S OrientSpeed paraméteren megadott fordulatszámra,
 - zárja a pozíciószabályozó hurkot (SN_LPCLSD=1),
 - majd ráál a nullimpulzustól számított, az N0684 Spindle Grid Shift paraméteren megadott pozícióra (a nullimpulzusra, ha a paraméter 0)
 - bekapcsolja az SN_ORIP jelzőt.
- ha az orsó áll és SP_OSHRT=0,
 - az SP_NEG jelzőn meghatározott irányban az N0800+n Rn S OrientSpeed paraméteren megadott sebességgel ráál a nullimpulzustól számított, az N0684 Spindle Grid Shift paraméteren megadott pozícióra (a nullimpulzusra, ha a paraméter 0)
 - bekapcsolja az SN_ORIP jelzőt.
- ha az orsó áll SP_OSHRT=1, és a nullimpulzus helyzete már ismert, valamint az N0607 Spindle Config paraméter #4 ZOR=1
 - a közelebbi irányban az N0800+n Rn S OrientSpeed paraméteren megadott sebességgel ráál a nullimpulzustól számított, az N0684 Spindle Grid Shift paraméteren megadott pozícióra (a nullimpulzusra, ha a paraméter 0)
 - bekapcsolja az SN_ORIP jelzőt.

Az **SP_OREQ=0** hatására az orsókezelő nyitja a pozíciószabályozó hurkot és kikapcsolja az SN_LPCLSD jelzőt.

SP_OSHRT: Orsó orientálás rövidebb úton (ha =0: a NEG előjele szerint orientál)

Ha az orsó áll és SP_OSHRT=0, az SP_NEG jelzőn meghatározott irányban orientál.

Ha az orsó áll, az SP_OSHRT=1, és a nullimpulzus helyzete már ismert, valamint az N0607 Spindle Config paraméter #4 ZOR=1 a rövidebb úton orientál.

SP_SSYNCR: Orsó szinkronizálás kérés

A jelző két orsó szinkronizálására szolgál. A szinkronizálás a két orsó nullimpulzusának pozícióhelyes együttfutását jelenti forgó állapotban. A két nullimpulzus futhat együtt, vagy az N0685 Spindle Phase Shift paraméteren beállított távolságra egymástól. A szinkron szabályozás erősítését az N0784+n Rn S Synchr K paraméterrel szabályozhatjuk. A mester orsó leállítása után is a szolga követi a mestert. A szinkronizálást egy mester orsóhoz mindig a szolga orsó kéri.

Szinkronizálást használunk például akkor, amikor ellenorsós gépen az ellenorsóval, mint szinkron szolgával forgó állapotban át kell venni a darabot. A pozícióhelyesség azt jelenti, hogy ha az egyik oldalon a főorsó nullimpulzusához viszonyítva munkáltunk meg valamit a darabon, az ellenorsóban a másik oldalon ehhez képest végezhetünk megmunkálást.

Szinkronizálás kérés előtt a szolga orsó SP_MAST regiszterébe be kell írni a mester orsó indexét.

Ha az N0685 Spindle Phase Shift paraméteren beállított értékkel el akarjuk tolni a két nullimpulzust egymástól, a szinkronizálás kérés előtt állítsuk be az SP_PHSHFTR jelzőt.

Az **SP_SSYNCR=1** hatására az orsókezelő

- ha a mester orsó fordulatszáma nagyobb, mint az N0319+n Rn S Rapid n=1...8 paraméteren meghatározott érték, lelassítja azt a paraméteren meghatározott értékre,
- zárja a pozíciószabályozó hurkot a mester orsón,
- zárja a pozíciószabályozó hurkot a szolga orsón,
- felpörgeti a szolgát a mester fordulatszámára, a mesterhez képest az SP_NEG jelzőn meghatározott irányba,
- az SP_PHSHFTR jelző állásának megfelelően a két orsó nullimpulzusát a megfelelő távolságra hozza,
- bekapcsolja az N0784+n Rn S Synchr K paraméterrel beállítható együttfutás szabályozást,
- bekapcsolja a szolga orsón az SN_SSYNA nyugtázó jelzőt és az SN_SINPOS jelzőt, ha a két orsó szinkronhibájának abszolút értéke a N0743+n Rn S Inpos paraméteren megadott értéknél kisebb.

Az **SP_SSYNCR=0** hatására az orsókezelő

- a mesterorsón nyitja a pozíciószabályozó hurkot,
- a mesterorsón, ha forog, beállítja az SP_PRG regiszterben megadott fordulatot,
- a szolga orsón nyitja a pozíciószabályozó hurkot,
- kikapcsolja az SN_SSYNA és az SN_SINPOS jelzőket.

SP_PHSHFTR: Fázistolás kérés szinkronizáláskor

Az **SP_PHSHFTR=1** állapotban, **SP_SSYNCR=1** szinkronizálás kérés hatására a két orsó nullimpulzusát az N0685 Spindle Phase Shift paraméteren beállított távolságra

állítja be és bekapcsolja a szolga orsón az SN_SSYNA nyugtázó jelzőt és az SN_SINPOS jelzőt, ha a két orsó szinkronhibájának abszolút értéke a N0743+n Rn S Inpos paraméteren megadott értéknél kisebb.

Az **SP_PHSHFTR=0** állapotban, **SP_SSYNCR=1** szinkronizálás kérés hatására a két orsó nullimpulzusát szinkronizálja egymáshoz és bekapcsolja a szolga orsón az SN_SSYNA nyugtázó jelzőt és az SN_SINPOS jelzőt, ha a két orsó szinkronhibájának abszolút értéke a N0743+n Rn S Inpos paraméteren megadott értéknél kisebb.

SP_POLYR: Nem használt

SP_SEND: Orsón jeladó figyelés tiltás

Ha a jelzőt 1-be kapcsoljuk az orsókezelő nem figyel az orsóra szerelt jeladó hibáit (nem ad Jeladó hibajelzést).

SP_SMTNDP: Orsótengely mozgás tiltás pozitív irányban

SP_SMTNDN: Orsótengely mozgás tiltás negatív irányban

Az SN_SMTNRP és az SN_SMTNRN mozgáskérés jelzőkre adott válaszjelek.

Minden orsó pozicionálás előtt (SP_OREQ=1, vagy SP_SLCLR=1 állapotban) az interpolátor a megfelelő irányú mozgáskérés jelzőt 1-be kapcsolja (SN_SMTNRP=1, SN_SMTNDN=1).

Az orsó addig nem mozog, amíg a PLC nem engedélyezi a mozgást a megfelelő irányban, az SP_SMTNDP=0, vagy az SP_SMTNDN=0 jelzőállással.

Az orsó megáll, ha a megfelelő irányú mozgástiltás jelzőt a PLC 1-be kapcsolja.

Lásd még SN_SRAPR, SP_RAPD, SP_FEEDD jelzőket.

A jelzők például orsók rögzítésére, oldására használhatók.

SP_SDETCR: Orsókezelő kikapcsolás kérés

Ha egy orsó működését ki kell kapcsolni, a megfelelő orsó SP_SDETCR jelzőjének 1-be írásával kérhetjük azt a rendszertől. Miután az orsókezelő leállította az orsó működését visszaadja az SN_SDETCHA=1 nyugtázó jelet. Visszakapcsoláskor az SP_SDETCR jelzőt 0-ba kell írni, és megvárni, amíg az orsókezelő SN_SDETCHA=0 állapotot ad vissza.

Ha az SN_SDETCHA jel 1 állapotban van az orsókezelő nem működik:

- törli az orsón a "van referenciapont" bejegyzést: SN_RPE=0,
- nem méri és nem tartja nyilván a jeladóról a pozíciót,
- nem zárja a pozíciószabályozó hurkot,
- semmilyen jelet nem ad ki az orsó hajtása felé,
- alkatrészprogramban és PLC-ben nem lehet az orsóra hivatkozni,

az orsó nem létezik.

Fordítva, ha az SP_SDETCR=0 és az NC kikapcsolja az SN_SDETCHA=0 jelzőt az orsó újra létezik.

Egy orsó akkor létezik, ha az

N0607 Spindle Config paraméteren #0 SEX értéke 1, és
AN_DETCH=0.

☞ **Figyelem!** Orsó ki, majd visszakapcsolását mindig a mondat pufferelesét megakadályozó funkcióban kell végezni (lásd Program paramétercsoport)

Példa:

Eszterga orsóját C tengellyé (körasztallá) kell alakítani, majd vissza orsóvá. Legyen a C tengely száma 3 (N0002 Axis Assign A3=1, N0100 Axis Name1 A3=C) az S1 orsó száma pedig 1. (N0605 Spindle Name2 S1=1)

Bekapcsolás után orsóként áll fel:

SP_SDETCR,#0=0, SN_SDETCR,#0=0
AP_DETCHR,#2=1, AN_DETCHR,#2=1

Ha át akarjuk kapcsolni az S1 orsót C tengellyé programozzunk egy megfelelő pufferelemző M funkciót és a jelzőket a következő módon állítsuk át:

SP_SDETCR,#0=1, SN_SDETCR,#0=1
AP_DETCHR,#2=0, AN_DETCHR,#2=0

Ettől kezdve ugyanarról az orsóhajtásról a jeladó jeleket nem az S1 orsókezelő veszi, hanem a C tengelykezelő. Az alapjelet nem az S1 orsókezelő adja ki az orsóhajtásnak, hanem a C tengely kezelő. A C tengelyt lehet mozgatni, C címre lehet alkatrészprogramban hivatkozni.

C tengely S1 orsóvá való visszaalakítása hasonló módon történik.

SP_SLCLR: Hurok zárás kérés orientálás nélkül

Az **SP_SLCLR=1** hatására az orsókezelő

- ha az orsó forog,
leállítja az orsó forgását,
zárja a pozíciószabályozó hurokot (SN_LPCLSD=1),
- ha az orsó áll zárja a pozíciószabályozó hurokot (SN_LPCLSD=1).

Az **SP_SLCLR=0** hatására az orsókezelő nyitja a pozíciószabályozó hurokot (SN_LPCLSD=0).

A funkció felhasználható pl. a merevszárú fűróciklus gyors végrehajtására, ha nem kell még egyszer ugyanabba a furatba betalálni. Lásd még: N0823 M Code for Closing S Loop paramétert és N1503 Drilling Cycles Config. paraméter #1 TSC bit.

SP_SSROFF: Szolga orsón pozíció hurok nyitás

Ha az orsó **szinkron szolga**, azaz az orsón SN_SSYNA=1, nyitja a pozíciószabályozó hurokot, a szolga orsó átveszi a mester orsó sebesség parancsjelet és azt küldi ki a szolga hajtásának.

Példa:

Az ellenorsót szinkronizáljuk a főorsóhoz darabátvétel miatt, majd az ellenorsó tokmány is megfogja a darabot. Ezután, valahányszor a fő- és az ellenorsón is zárva a

tokmány, a túlhatározottságot elkerülendő, az SP_SSROFF=1 beállítással nyissuk a szolga orsón a pozíciószabályozó hurkot. Ekkor a szolga orsó átveszi a mestertől a sebesség parancsjeleket és együtt mozognak. Lásd még DP_SILCK hajtásjelet.

SP_SDISPD: Orsó kijelzés tiltása a képernyőn

Ha egy orsón az N0607 Spindle Config paraméteren #0 SEX értéke 1, az orsó automatikusan megjelenik az FST kijelzőn.

Ha a jelző SP_SDISPD=1, az adott orsó eltűnik az FST kijelzőről.

Például, ha egy orsót az SP_SDETCR jelzővel kikapcsolunk a kijelzését is ki lehet kapcsolni.

SP_FEEDD: Orsó tengely előtoló mozgás tiltás

Az SN_SRAPR gyorsmeneti mozgáskérés jelzőre adott válaszjel.

Ha SN_SRAPR=0, az előtoló mozgás akkor indul el, ha a PLC az SP_FEEDD=0 jelet beállította.

Ha SP_FEEDD=1 az előtoló mozgás nem megy.

A jelzőt az SP_SMTNDP, SP_SMTNDN mozgástiltás jelzőkkel együtt kell kezelni.

Ennek a jelzőnek a kezelését például előtoló és gyorsmeneti mozgások közötti áttétel-változtatásra lehet használni.

SP_RAPD: Orsó gyorsmeneti mozgás tiltás

Az SN_RAPR gyorsmeneti mozgáskérés jelzőre adott válaszjel.

Ha SN_RAPR=1, a gyorsmeneti mozgás akkor indul el, ha a PLC az SP_RAPD=0 jelet beállította.

Ha SP_RAPD=1 a gyorsmeneti mozgás nem megy.

A jelzőt az SP_SMTNDP, SP_SMTNDN mozgástiltás jelzőkkel együtt kell kezelni.

Ennek a jelzőnek a kezelését például előtoló és gyorsmeneti mozgások közötti áttétel-változtatásra lehet használni.

SP_TLCHIA: A szerszámcserre kérés nyugtázó jele

Az SN_TLCHI jelző nyugtázó jele.

Ha a PLC a jelet 1-be kapcsolja az SP_TLCHIA jelet, az NC törli az SN_TLCHI jelzőt. Miután az NC az SN_TLCHI jelet törölte, a PLC-nek is törölnie kell a nyugtázó jelet.

SP_TLCHA: A csoportcsere kérés nyugtázó jele

Az SN_TLCH jelző nyugtázó jele.

Ha a PLC a jelet 1-be kapcsolja az SP_TLCHA jelet, az NC törli az SN_TLCH jelzőt. Miután az NC az SN_TLCH jelet törölte, a PLC-nek is törölnie kell a nyugtázó jelet.

SP_TLSKP: Orsóban lévő szerszám törött

Ha a PLC szerszámtörést detektál és 1-be írja az SP_TLSKP jelzőt, az NC a táblázatban a szerszámmra bejegyzzi a “törött” állapotot, majd 1-be írja az SN_TLSKPA nyugtázó jelet.

Ezután kiadja az SN_TLCHI szerszámcseré kérés jelzőt.

Miután az NC visszatda az SN_TLSKPA nyugtázó jelet a PLC-nek ki kell kapcsolni az SP_TLSKP jelzőt.

SP_TLCD: Éltartamszámláló tiltása

Ha a PLC a jelzőt 1-be kapcsolja, az adott orsóban lévő szerszám éltartamszámlálója leáll.

☞ **Figyelem!** Az

*SP_SSTRT,
SP_OREQ,
SP_SSYNCR,
SP_SLCLR*

jelzők egymást kizáró parancsokat jelentenek. Közülük mindig csak egy értéke lehet =1!

SP_OSW: Orsó az orientálási kapcsolón van.

Ha az N0609 Spindle Encoder Config paraméter #4 OSW=1, az orsó orientálás interface bemenetre történik. A kapcsoló jelének fogadását mindenképp a PLC program Int0 gyorsmoduljában kell végezni a mintavétel gyakorisága miatt. A kapcsoló állapotát az SP_OSW jelzőre kell másolni a PLC programban. Az orsókezelő az SP_OSW jelző 1 állapotára kezdi meg az orientálást.

A megoldás pontossága eseti, meghatározása mérést igényel.

7.14.2 A duplaszavas orsóváltozók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
SN_1	Az NC-től a PLC-nek átadott orsó kezelő bitek (DWORD)	SP_1	A PLC-től az NC-nek átadott orsó-kezelő bitek (DWORD)
SN_NCOM	Hajtásnak kiadott orsó fordulatszám alapjel ford/min-ben (DWORD)	SP_PRG	Orsó programozott fordulatszám (DWORD)
SN_NACT	Orsó jeladóról mért aktuális fordulatszáma (DWORD)	SP_ROT	Orsó forgásállapot kód (3, 4, ...) (DWORD)
		SP_RNG	Orsó tartománykód (11, 12, ..., 18) (DWORD)
		SP_MAST	Az orsó mester orsójának indexe (0,1,2...) (DWORD)
		SP_ASSIGN	Orsó csatorna számhoz rendelése (1,2,...) (DWORD)
		SP_ACTT	Az orsóban lévő szerszám száma (DWORD)

Az NC-től a PLC felé menő duplaszavas orsóváltozók

SN_NCOM: Hajtásnak kiadott orsó fordulatszám alapjel ford/min-ben (DWORD)

Az **SP_SSTRT jelző 1** állapotában az orsóhajtás felé kiadott fordulatszám parancsjel értéke ford/min dimenzióban.

Az SN_NCOM regiszter értéke,

- ha **SP_PAR=0**, az SP_PRG regiszterben megadott fordulatszám figyelembe véve a fel-, lefuttatást, szorozva az orsó override-dal (SP_SOVER), korlátozva az N0641+n Rn S Min és az N0649+n Rn S Max paraméterekkel, G96 állapotban korlátozva a G92 S paranccsal megadott fordulatszámmal és az N0688 Min Spindle Speed G96 paraméterrel.
- ha **SP_PAR=1**, az N0657+n Rn S Jog Speed paraméteren megadott fordulatszám. Csak a fel-, lefuttatás van rá hatással, de sem az orsó override, sem a korlátozó paraméterek nem érvényesülnek.

SN_NACT: Orsó jeladóról mért aktuális fordulatszáma (DWORD)

Ha az orsóra jeladó van szerelve, az orsó bármely állapotában az orsó aktuális fordulatszámát mutatja ford/min dimenzióban.

A PLC-től az NC felé menő duplaszavas orséváltozók

SP_PRG: Orsó programozott fordulatszám (DWORD)

A CN_SC regiszterben átadott S kód értékét írja a PLC a megfelelő SP_PRG regiszterbe. Dimenziója ford/min.

Konstans vágósebesség esetén, G96 állapotban (CN_CSURFS=1), a CN_CSPN regiszterbe írt, az NC által a vágósebességhez kiszámolt orsó fordulatszámot kell bemásolni a megfelelő SP_PRG regiszterbe.

SP_ROT: Orsó forgásállapot kód (3, 4, ...) (DWORD)

Az SP_ROT regiszterbe kell írni a szabványos forgáskódokat: 3: M3, 4: M4, 5: M5, 19: M19.

Ide kell írni az N0821 No. of M Code for Spnd. Pos paraméteren megadott hurokzárás orientálás nélkül kódot.

Ebbe a regiszterbe kell írni az N0689 Spindle M Low és az N0690 Spindle M High paramétereken megadott M kódokat is.

Az orsó forgásállapot kódja az FST képernyőn kerül kiírásra.

SP_RNG: Orsó tartománykód (11, 12, ..., 18) (DWORD)

Ha az orsón a tartományváltás M kódra történik, a 8 tartományra az M11, M12, ..., M18 kódokat kell használni.

Az orsó aktuális tartománykódját kell a regiszterbe írni, akkor is, ha a tartományt S kódra váltjuk. Csak 11, 12, ..., 18 írható a regiszterbe.

Az orsókezelő az SP_RNG kód alapján veszi figyelembe az orsók tartományfüggő paramétereit.

Kitöltése kötelező!

SP_MAST: Az orsó mester orsójának indexe (0,1,2...) (DWORD)

Orsók szinkronizálása esetén a szolga orsó kéri a szinkronizációt az SP_SSYNCR jelző beírásával. A szinkronizálást kérő szolga orsó SP_MAST regiszterében kell megadni a mester orsó indexét. (Orsó száma - 1.)

SP_ASSIGN: Orsó csatorna számhoz rendelése (1,2,...) (DWORD)

Ebben a regiszterben adhatjuk meg, hogy az adott orsó melyik csatornában működik.

A regiszterbe mindig a csatorna számát kell írni: az 1. csatorna esetén 1-et stb. Az orsóra alkatrész programból csak abban a csatornában hivatkozhatunk amelyik csatornához az orsót az SP_ASSIGN regiszter kijelölte.

Programfutás során szükség lehet arra, hogy egy orsó másik csatornába kerüljön, onnan lehessen megmunkálni vele. Az átirányítást M funkcióval végezheti a PLC program.

 **Figyelem!** Az SP_ASSIGN regiszter átírását csak a mondatpuffert ürítő funkcióban szabad végezni!

Kitöltése kötelező, bekapcsolás után inicializálni kell!

SP_ACTT: Az orsóban lévő szerszám száma (DWORD)

Ha az N2901 Search Config paraméter #7 TSP bitje 1, az aktuális szerszámszám kijelzése az FST ablakban az SP_ACTT orsónként indexelt regiszterből történik. A PLC program az adott orsóban lévő szerszám számát írja ide. Általában többszörös marógépeken lehet használni.

7.14.3 A lebegőpontos orsóváltozók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
		SP_SOVER	Orsó override: ha =1: 100% (double)

A PLC-től az NC felé menő lebegőpontos orsóváltozók

SP_SOVER: Orsó override: ha =1: 100% (double)

Orsónként rendelkezésre áll egy override regiszter. A regiszterbe lebegőpontosan kell az override értékét beírni. Ha például

SP_SOVER=0.32 jelentése 32%

SP_SOVER=1.0 jelentése 100%

NCT gépi kezelőpanel használata esetén az orsó override kapcsoló állását az MKSOVER regiszterből lehet venni, vagy az MB_SMAX, MB_S100, MB_SMAX gombokról származtatni.

Az orsó override alsó és felső korlátját a PLC programban kell beállítani!

 **Figyelem!** MKSOVER egész DWORD típusú, SP_SOVER pedig lebegőpontos double, tehát az override beállítása fixpontosból lebegőpontosba való konverziót (FLT utasítás) igényel. SP_SOVER indexelése 2-esével történik!

7.15 A csatorna kezelő változók

Az csatorna kezelő változók olyan az NC-től a PLC-be, vagy a PLC-ből az NC-be menő változók, amelyek csatornánként indexelődnek. Az összes itt közölt szimbólum az első (0-ás indexű) csatornára vonatkozik. A többi csatorna megfelelő változója indexelt címmel érhető el. A vezérlő maximum 8 csatornát kezel.

Az

CN-nel kezdődő változók az NC-től a PLC-be mennek (Bemenetek), míg az CP-vel kezdődő változók a PLC-től az NC-be (Kimenetek).

7.15.1 A bites csatornaváltozók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
CN_M1STB	1. M funkció beíró jel	CP_START	Start kérés
CN_M2STB	2. M funkció beíró jel	CP_STOP	Stop kérés
CN_M3STB	3. M funkció beíró jel	CP_JOG	Mozgatás üzemmód kérés
CN_M4STB	4. M funkció beíró jel	CP_INCR	Léptetés üzemmód kérés
CN_M5STB	5. M funkció beíró jel	CP_HNDL	Kézikerék üzemmód kérés
CN_M6STB	6. M funkció beíró jel	CP_REFP	Refpontfelvétel üzemmód kérés
CN_M7STB	7. M funkció beíró jel	CP_EDIT	Szerkesztés üzemmód kérés
CN_M8STB	8. M funkció beíró jel	CP_AUTO	Automata üzemmód kérés
CN_SSTB	S funkció beíró jel	CP_MDI	Kézi adatbeviteli üzemmód kérés
CN_TSTB	T funkció beíró jel	CP_JOGRAP	Jog mozgatás gyorsmenettel
CN_AUX1STB	1. Segédfunkció beíró jel	CP_TAXF	Szerszám irányú kézi mozgatás kérés
CN_AUX2STB	2. Segédfunkció beíró jel	CP_TRGAF	Szerszámra merőleges kézi mozgatás kérés
CN_AUX3STB	3. Segédfunkció beíró jel	CP_TTCRF	Szerszám középpont körüli kézi mozgatás kérés
CN_GSTB	Nem használt	CP_TBLB	Asztal szerinti kézi mozgatás kérés
CN_BKBUF	Feldolgozható mondat a pufferben	CP_INTDREQ	Nem használt
CN_STPREQ	Nem használt	CP_TLCM	Hosszkorrekció bemérés üzemmód be
CN_ALARM	Nem használt	CP_S2TS	S2-höz tartozó hossz bemérés választása
CN_OPMS	Nem használt	CP_WPCM	Munkadarab nullpont bemérés üzemmód be

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
CN_INTD	Függ állapot	CP_S2WS	S2-höz tartozó nullpont bemérő választása
CN_WTNG	Másik csatorna(ák) összeváró M kódjára vár	CP_AHND	Kézikerékkel történő nullponteltolás kérés
CN_ITFCHK	Nem használt	CP_WMCAXF	Alapelforgatással párhuzamos irányú előtolás kérés
CN_ITFALM	Nem használt		
CN_CSURFS	Konstans vágó sebesség (G96)		
CN_POLYT	Sokszögestergálás üzem (G51.2)		
CN_INCH	Méretadatok inchben programozva (G20)		
CN_HSHP	Nagysebességű nagy pontosságú üzem (G5.1)		
CN_CSACK	Nem használt		
CN_WPCNT	Megmunkált darab = elkészítendő darab		
CN_EGBMD	EGB mód (G81.8)		
CN_RTRFIN	Kikapás befejezve		
CN_IPSTP	Interpolátor áll	CP_SGLBK	Mondatonkénti üzem be
CN_IPEPTY	Interpolátor lejárt	CP_CNDSP	Feltételes stop be
CN_CBFR	Forgácsoló mondat előtolás kérés	CP_TEST	Teszt üzemmód kérés
CN_OVDIS	Override tiltva (G63)	CP_MLCK	Gép zárva üzemmód kérés
CN_THRD	Menetvágás (G33, G34)	CP_DRRUN	Száraz futás kérés
CN_THRDC	Menetvágó ciklus (G76, G78)	CP_BKRST	Mondat újra kérés
CN_TAP	Menetfűrés (G84)	CP_BKRET	Mondat vissza kérés
CN_RTAP	Merevszárú Menetfűrés (G84.2...)	CP_FLCK	Funkció zárva kérés
CN_REFPG	Programozott refpontfelvétel (G28)	CP_ABSOFF	Nem használt
CN_DWELL	Várakozás (G04)	CP_CNDBK_1	Feltételes mondat 1 be
CN_SKIP	Tapintás fut (G31)	CP_CNDBK_2	Feltételes mondat 2 be
CN_FREV	Fordulatonkénti előtolás (G95)	CP_CNDBK_3	Feltételes mondat 3 be
CN_POSCHK	Pozícióban jelre vár	CP_CNDBK_4	Feltételes mondat 4 be
CN_CHOP	Nem használt	CP_CNDBK_5	Feltételes mondat 5 be
CN_CHARP	Nem használt	CP_CNDBK_6	Feltételes mondat 6 be
CN_TLLE	A T kóddal hivatkozott szerszám éltartama lejárt	CP_CNDBK_7	Feltételes mondat 7 be

7.15 A csatorna kezelő változók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
		CP_CNDBK_8	Feltételes mondat 8 be
CN_START	Start állapot	CP_FIN	Kész jel, összes funkció végrehajtva
CN_STOP	Stop állapot	CP_RST	Nem használt
CN_JOG	Mozgatás üzemmód	CP_RSTREW	Nem használt
CN_INCR	Léptetetés üzemmód	CP_CSREQ	Nem használt
CN_HNDL	Kézikerék Üzemmód	CP_NOWT	Nem kell összeváró M kódra várni másik csatornától
CN_REFP	Refpontfelvétel üzemmód	CP_MINT	Megszakítási makró hívás
CN_EDIT	Szerkesztés üzemmód	CP_TMREN	Szabad felhasználású időmérő engedélyezése
CN_AUTO	Automata üzemmód	CP_TSBD	Nem használt
CN_MDI	Kézi adatbeviteli üzemmód	CP_EGBRRQ	Szerszám kikapás kérés EGB üzemben
CN_FLCK	Funkció zárva üzemmód	CP_M1ACK	1. M funkció végrehajtva
CN_TAXF	Szerszám irányú kézi mozgatás üzem	CP_M2ACK	2. M funkció végrehajtva
CN_TRGAF	Szerszámra merőleges kézi mozgatás üzem	CP_M3ACK	3. M funkció végrehajtva
CN_TTCRF	Szerszám középpont körüli kézi mozgatás üzem	CP_M4ACK	4. M funkció végrehajtva
CN_TBLB	Asztal szerinti kézi mozgatás üzem	CP_M5ACK	5. M funkció végrehajtva
CN_ABSOFF	Nem használt	CP_M6ACK	6. M funkció végrehajtva
CN_TEST	Teszt üzemmód	CP_M7ACK	7. M funkció végrehajtva
CN_MLCK	Gép zárva üzemmód	CP_M8ACK	8. M funkció végrehajtva
CN_DRRUN	Száraz futás üzem	CP_SACK	S funkció végrehajtva
CN_BKRST	Mondat újra állapot	CP_TACK	T funkció végrehajtva
CN_BKRET	Mondat vissza állapot	CP_AUX1ACK	1. segédfunkció végrehajtva
CN_AHND	Kézikerékkel történő nullponteltolás beváltva	CP_AUX2ACK	2. segédfunkció végrehajtva
CN_WMCAXF	Alapelforgatással párhuzamos irányú előtolás bekapcsolva	CP_AUX3ACK	3. segédfunkció végrehajtva
		CP_GACK	Nem használt
CN_1100	#1100 makrováltozó értéke (bit)	CP_HOLD	Mozgás állj minden tengelyre a csatornában

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
CN_1101	#1101 makrováltozó értéke (bit)	CP_CBFEN	Forgácsoló mondat előtolás engedélyezés
CN_1102	#1102 makrováltozó értéke (bit)	CP_FHNDL	Előtolás kézikerekről üzem be
CN_1103	#1103 makrováltozó értéke (bit)	CP_OVC	Nem használt
CN_1104	#1104 makrováltozó értéke (bit)	CP_HNDLS1	Előtolás az 1. kézikerekről
CN_1105	#1105 makrováltozó értéke (bit)	CP_HNDLS2	Előtolás a 2. kézikerekről
CN_1106	#1106 makrováltozó értéke (bit)	CP_HNDLS3	Előtolás a 3. kézikerekről
CN_1107	#1107 makrováltozó értéke (bit)	CP_HNDLS4	Előtolás a 4. kézikerekről
CN_1108	#1108 makrováltozó értéke (bit)	CP_LIM1DIS	1. paraméteres végállás tartomány tiltása
CN_1109	#1109 makrováltozó értéke (bit)	CP_LIM2DIS	2. paraméteres végállás tartomány tiltása
CN_1110	#1110 makrováltozó értéke (bit)	CP_LIM3DIS	3. paraméteres végállás tartomány tiltása
CN_1111	#1111 makrováltozó értéke (bit)	CP_LIMSEL	1B paraméteres végállás tartomány választása minden tengelyre
CN_1112	#1112 makrováltozó értéke (bit)	CP_CHOPON	Nem használt
CN_1113	#1113 makrováltozó értéke (bit)	CP_SGOEN	Második geometria korrekciós tábla engedélyezése
CN_1114	#1114 makrováltozó értéke (bit)	CP_SGOX	Második geometria korrekció figyelembe vétele az X tengelyen
CN_1115	#1115 makrováltozó értéke (bit)	CP_SGOY	Második geometria korrekció figyelembe vétele az Y tengelyen
CN_1116	#1116 makrováltozó értéke (bit)	CP_SGOZ	Második geometria korrekció figyelembe vétele az Z tengelyen
CN_1117	#1117 makrováltozó értéke (bit)	CP_OSGNX	X korrekció ellenkező előjellel (xkorrekció=-xkorrekció)
CN_1118	#1118 makrováltozó értéke (bit)	CP_OSGNY	Y korrekció ellenkező előjellel (ykorrekció=-ykorrekció)
CN_1119	#1119 makrováltozó értéke (bit)	CP_OSGNZ	Z korrekció ellenkező előjellel (zkorrekció=-zkorrekció)
CN_1120	#1120 makrováltozó értéke (bit)	CP_ROVLD	Gyorsmeneti mondatok átlapolásának kikapcsolása
CN_1121	#1121 makrováltozó értéke (bit)	CP_RLSOT3	Nem használt
CN_1122	#1122 makrováltozó értéke (bit)		
CN_1123	#1123 makrováltozó értéke (bit)		
CN_1124	#1124 makrováltozó értéke (bit)		
CN_1125	#1125 makrováltozó értéke (bit)		

7.15 A csatorna kezelő változók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
CN_1126	#1126 makrováltozó értéke (bit)		
CN_1127	#1127 makrováltozó értéke (bit)		
CN_1128	#1128 makrováltozó értéke (bit)		
CN_1129	#1129 makrováltozó értéke (bit)		
CN_1130	#1130 makrováltozó értéke (bit)		
CN_1131	#1131 makrováltozó értéke (bit)		
		CP_1000	#1000 makrováltozó (bit)
		CP_1001	#1001 makrováltozó (bit)
		CP_1002	#1002 makrováltozó (bit)
		CP_1003	#1003 makrováltozó (bit)
		CP_1004	#1004 makrováltozó (bit)
		CP_1005	#1005 makrováltozó (bit)
		CP_1006	#1006 makrováltozó (bit)
		CP_1007	#1007 makrováltozó (bit)
		CP_1008	#1008 makrováltozó (bit)
		CP_1009	#1009 makrováltozó (bit)
		CP_1010	#1010 makrováltozó (bit)
		CP_1011	#1011 makrováltozó (bit)
		CP_1012	#1012 makrováltozó (bit)
		CP_1013	#1013 makrováltozó (bit)
		CP_1014	#1014 makrováltozó (bit)
		CP_1015	#1015 makrováltozó (bit)
		CP_1016	#1016 makrováltozó (bit)
		CP_1017	#1017 makrováltozó (bit)
		CP_1018	#1018 makrováltozó (bit)
		CP_1019	#1019 makrováltozó (bit)
		CP_1020	#1020 makrováltozó (bit)
		CP_1021	#1021 makrováltozó (bit)
		CP_1022	#1022 makrováltozó (bit)
		CP_1023	#1023 makrováltozó (bit)
		CP_1024	#1024 makrováltozó (bit)
		CP_1025	#1025 makrováltozó (bit)
		CP_1026	#1026 makrováltozó (bit)
		CP_1027	#1027 makrováltozó (bit)
		CP_1028	#1028 makrováltozó (bit)

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
		CP_1029	#1029 makrováltozó (bit)
		CP_1030	#1030 makrováltozó (bit)
		CP_1031	#1031 makrováltozó (bit)

Az NC-től a PLC-be menő bites csatornaváltozók

CN_M1STB: 1. M funkció beíró jel

CN_M2STB: 2. M funkció beíró jel

CN_M3STB: 3. M funkció beíró jel

CN_M4STB: 4. M funkció beíró jel

CN_M5STB: 5. M funkció beíró jel

CN_M6STB: 6. M funkció beíró jel

CN_M7STB: 7. M funkció beíró jel

CN_M8STB: 8. M funkció beíró jel

Alkatrészprogramban egy mondatba maximum 8 különböző M kódot írhatunk, azaz egy mondatban 8 különböző M kódot tud az NC átadni a PLC-nek.

Ha az alkatrészprogramba M funkciót programoztak, a csatornakezelő

- a CN_MnC (n=1, 2, ..., 8) regiszterbe beírja az M kód értékét
- majd a **CN_MnSTB** beírójelet az NC **1 PLC ciklus időtartamra** 1-be írja.

A 8 különböző M kód értékét a vezérlő 8 különböző CN_MnC (n=1, 2, ..., 8) regiszterbe írja. A 8 átvadó regiszterhez 8 beíró jel CN_MnSTB (n=1, 2, ...8) tartozik. A beírójel hatására a kapott kód alapján a funkciót a PLC dekódolja.

Ha a kapott kód a gépen létező funkció takar, a beírójelnek megfelelő CP_MnACK nyugtázó jelzőt a PLC 0-ba állítja. A CP_MnACK nyugtázó jelzőt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

CN_SSTB: S funkció beíró jel

Ha az alkatrészprogramban S funkciót programoztak, a csatornakezelő

- a CN_SC regiszterbe beírja az S funkción programozott fordulatszám értékét,
- a CN_SSEL regiszterbe beírja a hivatkozott orsó számát (1...16)
- a CN_RNGREQ regiszterbe beírja a programozott fordulatszámhoz tartozó tartomány kódját (11...18),
- majd a **CN_SSTB** jelzőt **1 PLC ciklus időtartamra** 1-be írja.

A beírójel hatására a PLC a fenti regiszterekben írt kódokat dekódolja és a CP_SACK nyugtázó jelet 0-ba állítja. A CP_SACK nyugtázó jelzőt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

CN_TSTB: T funkció beíró jel

Ha az alkatrészprogramban T funkciót programoztak, a csatornakezelő

- a CN_TC regiszterbe beírja a T funkción programozott szerszám számát, eszterga csatornában a T címen hívott *korrekciószámot levágva*,
- a CN_MGZNO regiszterbe beírja, hogy a hivatkozott szerszám hányadik magazinban található,
- a CN_POTNO regiszterbe beírja, hogy a hivatkozott szerszám a megadott magazin hányadik zsebében van,
- megvizsgálja, hogy a hivatkozott szerszám, vagy szerszámcsoport éltartama lejárt-e, és ennek megfelelően beállítja a CN_TLLE jelzőt,
- majd a **CN_TSTB** jelzőt *1 PLC ciklus időtartamra* 1-be írja.

A beírójel hatására a PLC a fenti regiszterekbe írt kódokat dekódolja és a CP_TACK nyugtázó jelet 0-ba állítja. A CP_TACK nyugtázó jelzőt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

CN_AUX1STB: 1. Segédfunkció beíró jel

CN_AUX2STB: 2. Segédfunkció beíró jel

CN_AUX3STB: 3. Segédfunkció beíró jel

Az N1332+n Aux Fu Addr_n (n=1..3) paraméteren ki lehet jelölni az A, B, C, U, V, W címek közül három különbözőt. Ha alkatrészprogramban az itt megadott három segédfunkció címére hivatkozunk, a csatornakezelő

- a CN_AUX_nC (n=1...3) regiszterbe beírja a segédfunkció címén programozott értéket (mindig egész DWORD),
- majd a **CN_AUX_nSTB** (n=1...3) jelzőt *1 PLC ciklus időtartamra* 1-be írja.

A beírójel hatására a PLC a fenti regiszterekbe írt kódokat dekódolja és a CP_AUX_nACK (n=1...3) nyugtázó jelet 0-ba állítja. A CP_AUX_nACK nyugtázó jelzőt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

CN_GSTB: Nem használt

CN_BKBUF: Feldolgozható mondat a pufferben

A jelzőt az NC 1-be kapcsolja, ha

- automata üzemmódban (CN_AUTO=1) ki van jelölve egy program automatában történő futásra,
- MDI üzemmódban (CN_MDI=1) van,
- jog, inkrementális jog, vagy kézikérék üzemmódban (CN_JOG=1, CN_INCR=1, vagy CN_HNDL=1) lezárjuk egy mondat bevitelét, addig, amíg végre nincs a mondat hajtva.

CN_STPREQ: Nem használt

CN_ALARM: Nem használt

CN_OPMES: Nem használt

CN_INTD: Függ állapot

A jelzőt a csatornakezelő 1-be írja, ha automata üzemben a program futása meg lett szakítva. Megszakítás a következő esetekben lesz:

- programfutás közben kilépünk automata üzemmódból,
- automata üzemben vészállapot jön létre.

A megszakítási folyamat során a csatornakezelő

- letárolja a programvégrehajtás állapotát,
- a megszakítási pozíciót
- a funkció nyugtázó jelek (CP_MnACK, CP_SACK, CP_TACK, CP_AUXnACK) állapotát,

A megszakított program, visszalépve automata üzemmódba, folytatható a

CN_BKRST, CN_BKRET feltételek függvényében.

Azokat a funkciókat, amelyeknek a nyugtázó jele 0 volt a megszakításkor újra kiadja.

CN_WTNG: Másik csatorna(ák) összeváró M kódjára vár

Többcsatornás vezérlőkön szükség van arra, hogy az egyik csatornában futó alkatrész-program végrehajtását felfüggeszük addig, amíg egy vagy több másik csatorna a program végrehajtásában egy meghatározott pontra nem ér. Ezt nevezzük a csatornák szinkronizálásának, vagy összevárásának. A csatornák szinkronizálását M kódokkal lehet végezni. Az N2201 Waiting M Codes Min és az N2202 Waiting M Codes Max paramétereken kijelölhetünk egy M kód tartományt.

Az összevárást teljes egészében az NC végzi, azzal a PLC-nek nem kell foglalkozni.

Abban az esetben, ha egy csatorna összevárásban van az NC a CN_WTNG jelzőt 1-be írja.

CN_ITFCHK: Nem használt

CN_ITFALM: Nem használt

CN_CSURFS: Konstans vágó sebesség (G96)

Ha a csatorna G96 (konstans vágósebesség számítás) állapotban van,

- az NC 1-be írja a CN_CSURF jelzőt,
- PLC ciklusonként kiszámítja és a CN_CTSPN regiszterbe beírja a pillanatnyi koordinátaához tartozó orsó fordulatszámot,
- a G92 S utasítással programozott maximális fordulatszámot beírja a CN_NMAX regiszterbe.

A PLC program feladata, hogy a CN_CSURF=1 állapotban a CN_CTSPN regiszterben kapott értéket átmásolja a megfelelő (CP_SINP regiszter által meghatározott) orsó SP_PRG regiszterébe.

CN_POLYT: Sokszögesztergálás üzem (G51.2)

Ha a csatorna kezelő G51.2 sokszögesztergálás parancsot hajt végre a jelzőt 1-be kapcsolja. A jelzőt a G50.2 parancs kapcsolja ki.

Ha a jelző 1 állapotban van, a PLC-nek meg kell akadályoznia, hogy a CP_POLYSL regiszterben kijelölt orsó bármilyen funkciót hajtson végre.

CN_INCH: Méretadatok inchben programozva (G20)

A jelző =0 G21 metrikus adatbevitel esetén.

A jelző =1 G20 inch-es adatbevitel esetén.

A jelző és az N0104 Unit of Measure paraméter #0 IND bitjének függvényében kell a PLC-nek a CP_INC inkremens regiszterbe beírni a lépésmagyságot inkrementális jog üzemben. Példa:

- Ha IND=0, CN_INCH=0 és MB_I100-at nyomtak: CP_INC=0.1
- Ha IND=0, CN_INCH=1 és MB_I100-at nyomtak: CP_INC=0.254

CN_HSHP: Nagysebességű nagy pontosságú üzem (G5.1)

CN_CSACK: Nem használt

CN_WPCNT: Megmunkált darab = elkészítendő darab

A munkadarab számlálót a vezérlőn az Idők/Számlálók ablakban állíthatja be a kezelő.

Ha a megmunkált darabok száma elérte az elkészítendő darabok számát a vezérlő a CN_WPCNT jelzőt 1 PLC ciklus ideig 1-be írja.

A vezérlő az elkészült darabok számlálóját az N2305 Part Count M paraméteren megadott számú M funkcióra növeli. Pl: ha a paraméter =30, minden M30 funkcióra növeli a számlálót.

CN_EGBMD: EGB mód (G81.8)

Ha a csatorna kezelő G81.8 fogazási funkciót (elektronikus hajtómű) hajt végre 1-be kapcsolja a CN_EGBMD jelzőt. A jelzőt a G80.8 funkció (elektronikus hajtómű ki) törli. (Lásd: N1800 EGB Contr paramétercsoport.)

CN_RTRFIN: Kikapás befejezve

Ha a csatorna kezelő G81.8 fogazási funkciót (elektronikus hajtómű) hajt végre (CN_EGBMD=1) és a PLC fogazószerszám kikapást kér a CP_EGBRRQ=1 jelzővel, az NC az N1804 Retr. Dist. paraméteren meghatározott irányban és távolságra kihúzza

a szerszámot. A kihúzás végén a csatornakezelő 1-be írja az CN_RTRFIN jelzőt. Ezután a PLC-nek ki kell kapcsolnia CP_EGBRRQ=0 jelzőt.

CN_IPSTP: Interpolátor áll

A jelző az alábbi esetekben veszi fel a CN_IPSTP=1 állapotot:

- a csatornában nem fut program,
- a csatornában program fut, de az override értéke 0 CP_FOVER=0. Ez igaz G0 pozícionáló mondatokban is.
- a csatorna stop állapotba kerül CN_STOP=1 (pl: stop gomb hatására, mondatonkénti üzemben, stb.).

CN_IPEPTY: Interpolátor lejár

A jelző az alábbi esetekben veszi fel a CN_IPEPTY=1 állapotot:

- a csatornában nem fut program,
- a csatornában program fut, de az interpolátor áll, például azért, mert a csatorna funkciómondatot hajt végre.

CN_CBFR: Forgácsoló mondat előtolás kérés

A csatornakezelő az alábbi esetekben kéri a PLC-től az előtoló mozgás engedélyezését a CN_CBFR=1 jelző bekapcsolásával:

- G01 egyenes interpoláció,
- G02, G03 körinterpoláció,
- G33 menetvágás,
- minden ciklusban, amely a fenti modatok valamelyikét hajtja végre.

Az előtolás (mozgás) addig nem indul el, amíg a PLC a CP_CBFEN=1 jelzón azt nem engedélyezi, például amíg az orsó nem forog.

CN_OVDIS: Override tiltva (G63)

A csatornakezelő az alábbi esetekben jelzi az override és stop tiltva állapotot a PLC-nek a CN_OVDIS=1 jelző bekapcsolásával:

- G63 override tiltás be állapot (kikapcsolja: G61, G62, G64),
- G33 menetvágó mondatban (kikapcsolja egyéb interpoláció, pl G0, G1),
- G74, G84 menetfűró ciklusban, amíg a furatban dolgozik,
- G78 egyszerű menetvágó ciklusban, amikor a menetet vágja,
- G76 összetett menetvágó ciklusban, amikor a menetet vágja.

CN_THRD: Menetvágás (G33, G34)

A csatornakezelő a jelzőt 1-be kapcsolja CN_THRD=1, amikor

- G33 menetvágó mondatot hajt végre (kikapcsolja egyéb interpoláció, pl G0, G1),
- G78 egyszerű menetvágó ciklusban a menetet vágja,
- G76 összetett menetvágó ciklusban a menetet vágja.

CN_THRDC: Menetvágó ciklus (G76, G78)

A csatorna kezelő a jelzőt 1-be kapcsolja CN_THRDC=1, amikor

- G78 egyszerű menetvágó ciklusban a menetet vágja,
- G76 összetett menetvágó ciklusban a menetet vágja.

CN_TAP: Menetfúrás (G84)

A csatorna kezelő a jelzőt 1-be kapcsolja CN_TAP=1, amikor

- G74, G84 menetfúró ciklusban a furatban dolgozik.

CN_RTAP: Merevszárú Menetfúrás (G84.2...)

A csatorna kezelő a jelzőt 1-be kapcsolja CN_RTAP=1, amikor

- G84.2, G84.3 merevszárú menetfúró ciklusban a furatban dolgozik.

CN_REFPG: Programozott refpontfelvétel (G28)

A csatorna kezelő a jelzőt 1-be kapcsolja CN_REFPG=1, amikor

- G28 utasításban a referenciapontot veszi fel.

CN_DWELL: Várakozás (G04)

A csatorna kezelő a jelzőt 1-be kapcsolja CN_DWELL=1, amikor

- G4 mondatban várakozást hajt végre,
- fúróciklusban programozott várakozást végez.

CN_SKIP: Tapintás fut (G31)

A csatorna kezelő a jelzőt 1-be kapcsolja CN_SKIP=1, amikor

- G31 funkciót hajt végre.

CN_FREV: Fordulatonkénti előtolás (G95)

A csatorna kezelő a jelzőt 1-be kapcsolja CN_FREV=1, amikor

- fordulatonkénti (G95) előtolással mozog.

G94 esetén (percenkénti előtolás) a jelző CN_FREV=0. Pozícionáló mondatban (G0, G53) a jelzőt kikapcsolja még akkor is, amikor G95 állapot van.

CN_POSCHK: Pozícióban jelre vár

A csatorna kezelő a jelzőt 1-be kapcsolja CN_POSCHK=1,

- minden gyorsmeneti mozgás (G0, G53) végén, ha az N1337 Execution Config paraméter #0 PCH=1,
 - minden mozgásmondat végén, amelyikbe G9-et programoztak,
 - minden mozgásmondat végén, G61 pontos megállás állapotban,
- és megvárja, hogy a mozgásban részt vevő összes tengelyen a lemaradás értéke a N0516 Inpos paraméteren meghatározott ablakon belülre kerüljön. Ezután a CN_POSCHK jelzőt törli.

CN_CHOP: Nem használt

CN_CHARP: Nem használt

CN_TLLE: A T kóddal hivatkozott szerszám éltartama lejárt

Minden szerszámhíváskor (T kód) a csatorna kezelő megvizsgálja, hogy a hivatkozott szerszám, vagy szerszámcsoport éltartama lejárt-e, feltéve, hogy az N2900 Tool M. Config paraméter #0 TMU=1.

Ha az éltartam lejárt, az NC a CN_TSTB jellel párhuzamosan a CN_TLLE jelet 1-be írja.

CN_START: Start állapot

A CP_START start kérés nyugtázó jele.

Amikor a PLC a CP_START jelet bekapcsolja a csatorna kezelő megvizsgálja, hogy

- az adott üzemmódban (ezek: auto, MDI, jog, inkrementális jog, kézikérék) hajtható-e végre program, vagy egyedi mondat,
- van-e kijelölt program, vagy lezárt egyedi mondat végrehajtásra,
- nincs-e egyéb akadálya az indításnak, pl. hibajelzés.

Ha a fenti feltételek teljesülnek a csatorna kezelő a CN_START=1 állapotot adja vissza, amire a start gomb lámpája (pl. ML_START) bekapcsolható.

A csatorna kezelő a CN_START állapotot törli, ha az NC

- stop állapotba kerül (CN_STOP=1)
- a végrehajtásra kijelölt programot, vagy a lezárt egyedi mondatot végrehajtotta.

CN_STOP: Stop állapot

A CP_STOP stop kérés nyugtázó jele.

Amikor a PLC a CP_STOP jelet bekapcsolja a csatorna kezelő megvizsgálja, hogy

- az adott üzemmódban (ezek: auto, MDI, jog, inkrementális jog, kézikérék) van-e végrehajtás alatt program, vagy egyedi mondat,
- nincs-e egyéb akadálya a leállításnak, pl. override és stop tiltás állapot.

Ha a fenti feltételek teljesülnek a csatorna kezelő leállítja az interpolációt és a CN_STOP=1 állapotot adja vissza, amire a stop gomb lámpája (pl. ML_STOP) bekapcsolható.

A csatorna kezelő a CN_STOP állapotot törli, ha az NC

- start állapotba kerül (CN_START=1)
- a végrehajtásra kijelölt programot, vagy a lezárt egyedi mondatot végrehajtotta.

CN_JOG: Mozgatás üzemmód

CN_INCR: Léptetetés üzemmód

CN_HNDL: Kézikerék Üzemmód

CN_REFP: Refpontfelvétel üzemmód

CN_EDIT: Szerkesztés üzemmód

CN_AUTO: Automata üzemmód

CN_MDI: Kézi adatbeviteli üzemmód

A CP_JOG, CP_INCR, CP_HNDL, CP_REFP, CP_EDIT, CP_AUTO és a CP_MDI üzemmód kérés jelzők nyugtázó jelei.

Amikor a PLC üzemmódváltást kér a CP_xxx jelzők valamelyikén, a csatornakezelő megvizsgálja, hogy

- abban az üzemmódban, amelyikben a csatorna van, van-e végrehajtás alatt program, vagy egyedi mondat,
- ha van programvégrehajtás, azonnal leállítja az interpolátort, vagy override és stop tiltás állapotban megvárja, hogy az állapot megszűnjön,
- ha egyedi mondatot hajt végre jog, inkrementális jok, kézikerék üzemmódban, vagy programot hajt végre MDI üzemmódban, törli a teljes végrehajtást,
- ha automata üzemmódban programot hajt végre beállítja a függ állapotot
CN_INTD=1,
- átlép a CP_xxx jelzőn kívánt üzemmódba és bekapcsolja a CN_xxx jelzőt.

A CN_xxx jelző 1 állapotában a PLC bekapcsolhatja a megfelelő üzemmód gomb lámpáját (pl. ML_xxx).

CN_FLCK: Funkció zárva üzemmód

A CP_FLCK funkció zárva kérés nyugtázó jele.

Amikor a PLC bekapcsolja a funkció zárva kérés jelet CP_FLCK=1, a csatornakezelő megvizsgálja, hogy

- a CN_FLCK jel állapota megváltoztatható-e, azaz a vezérlő nincs-e az automata (CN_AUTO=1), vagy MDI (CN_MDI=1) üzemmódok valamelyikében,
- ha nincs, a CN_FLCK jelző állapotát az ellenkezőjére változtatja.

A CN_FLCK jelző 1 állapotában a PLC bekapcsolhatja a gép zárva gomb lámpáját (pl: ML_FLCK).

A CN_FLCK jelző 1 állapotában a csatornakezelő semmilyen funkciót nem ad át a PLC-nek: nem adja ki a CN_MnSTB, CN_SSTB, CN_TSTB, CN_AUXnSTB beíró jeleket a PLC felé.

CN_TAXF: Szerszám irányú kézi mozgató üzemi

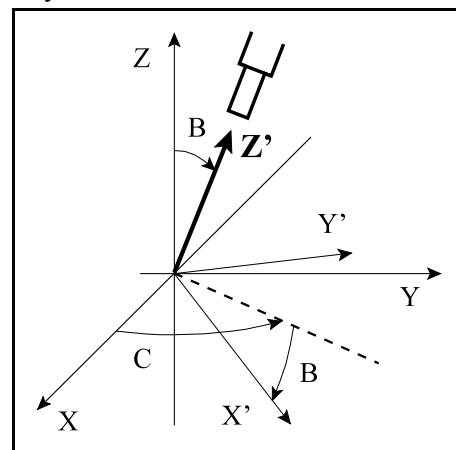
A CP_TAXF szerszámirányú kézi mozgató kérés nyugtázó jele. A jelzőt az NC 5 tengelyes *fej-fej, vagy fej-asztal* gépeken kapcsolja be, **jog, inkrementális jog és kézikerek üzemmódban**. Asztal-asztal gépeken nem működik.

A CN_TAXF=1 állapotban, az N3201 Tool Axis Direction **paraméteren megadott, szerszám irányú tengelyt kiválasztva** az N3204 No. of the First Rot. Ax. és az N3208 No. of the Second Rot. Ax. paramétereken meghatározott **forgó tengelyek szögállása függvényében mozgatja a szerszámot a szerszám irányában**.

Például:

Legyen a gépünk fej-fej azaz N3200 Mechanical Type=Head-Head, N3201 Tool Axis

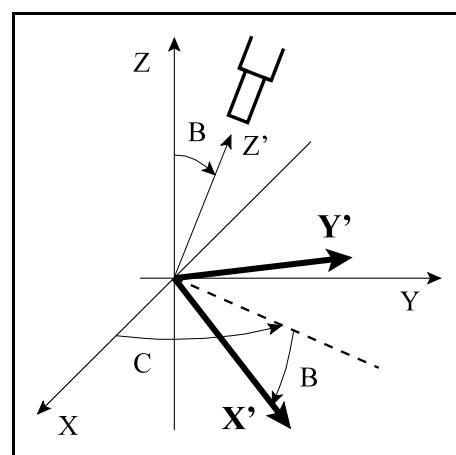
Direction=Z, azaz a szerszám a forgó tengelyek alaphelyzetében Z irányú párhuzamos, N3204 No. of the First Rot. Ax.=C és N3208 No. of the Second Rot. Ax.=B. Ekkor a **Z jog gomb megnyomására, vagy Z tengelyt választva a kézikerek mozgatóhoz**, a B és C tengelyek pozícióit figyelembe véve **mozog a szerszám irányában** akár mindhárom (X, Y, Z) tengely mentén.

**CN_TRGAF:** Szerszámra merőleges kézi mozgató üzemi

A CP_TRGAF szerszámra merőleges irányú kézi mozgató kérés nyugtázó jele. A jelzőt az NC 5 tengelyes *fej-fej, vagy fej-asztal* gépeken kapcsolja be, **jog, inkrementális jog és kézikerek üzemmódban**. Asztal-asztal gépeken nem működik. A CN_TRGAF=1 állapotban, az N3201 Tool Axis Direction **paraméteren megadott tengelyre merőleges irányok egyikét kiválasztva**, az N3204 No. of the First Rot. Ax. és az N3208 No. of the Second Rot. Ax. **paramétereken meghatározott forgó tengelyek szögállása függvényében mozgatja a szerszámot a szerszám tengelyére merőleges síkban**.

Például:

Legyen a gépünk fej-fej azaz N3200 Mechanical Type=Head-Head, N3201 Tool Axis Direction=Z, azaz a szerszám a forgó tengelyek alaphelyzetében Z irányú párhuzamos, N3204 No. of the First Rot. Ax.=C és N3208 No. of the Second Rot. Ax.=B. Ekkor az **X, vagy az Y jog gomb megnyomására, illetve X, vagy Y tengelyt választva, a kézikerek mozgatóhoz**, a B és C

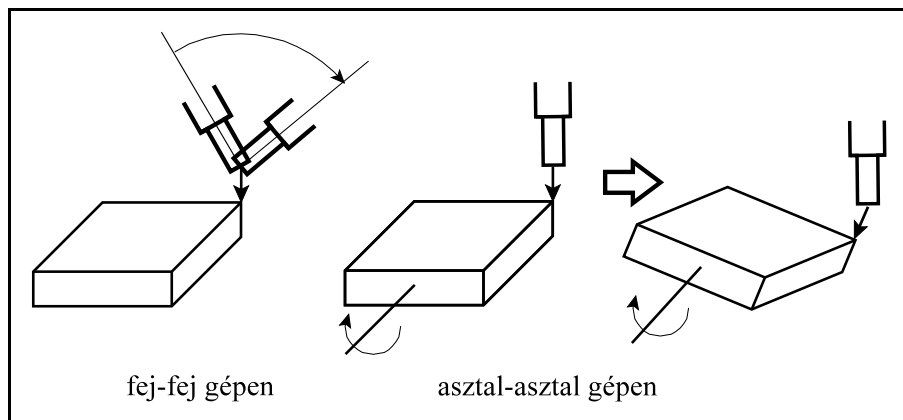


tengelyek pozícióit figyelembe véve **mozog a szerszámmra merőleges síkban** akár mindhárom (X, Y, Z) tengely mentén.

CN_TTCRF: Szerszám középpont körüli kézi mozgató üzemi

A CP_TTCRF szerszám középpont körüli kézi mozgató kérés nyugtázó jele. A jelzőt az NC **mindhárom 5 tengelyes géptípus esetén** visszaadja, **jog, inkrementális jog és kézikerek üzemi módban**.

A CN_TTCRF=1 állapotban, az N3204 No. of the First Rot. Ax., vagy az N3208 No. of the Second Rot. Ax. paramétereken kijelölt **fordó tengelyeket mozgatva úgy mozog, hogy a szerszám középpont helyzete a darab megfelelő pontjához képest változatlan marad**.



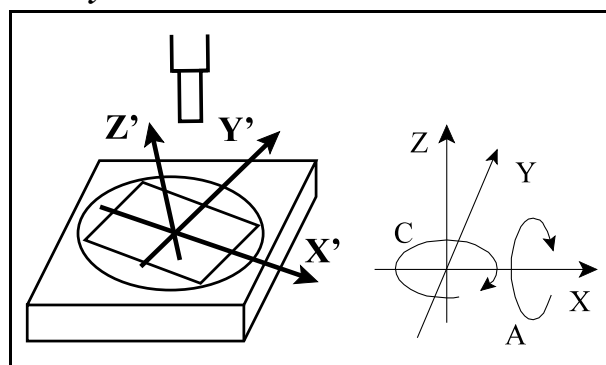
CN_TBLB: Asztal szerinti kézi mozgató üzemi

A CP_TBLB asztal szerinti kézi mozgató kérés nyugtázó jele. A jelzőt az NC 5 tengelyes **asztal-asztal, vagy fej-asztal** gépeken kapcsolja be, **jog, inkrementális jog és kézikerek üzemi módban**. Fej-fej gépeken nem működik.

A CN_TBLB=1 állapotban, az N3204 No. of the First Rot. Ax., és az N3208 No. of the Second Rot. Ax. **paramétereken meghatározott fordó tengelyek szögállása függvényében mozgatja a lineáris (X, Y, Z) tengelyeket** úgy, hogy az N3201 Tool Axis Direction paraméteren megadott tengelyt tekintve az asztalra merőleges iránynak, míg a másik két tengelyt az asztal síkjára eső irányoknak.

Például:

Legyen a gépünk asztal-asztal azaz N3200 Mechanical Type=Table-Table, N3201 Tool Axis Direction=Z, azaz a szerszám a fordó tengelyek alaphelyzetében Z irányral párhuzamos, N3204 No. of the First Rot. Ax.=A és N3208 No. of the



Second Rot. Ax.=C. Ekkor az *X, vagy az Y jog gomb megnyomására, illetve X, vagy Y tengelyt választva, a kézikerék mozgatóhoz*, az A és C tengelyek pozícióit figyelembe véve *mozog az asztal síkjában*. A *Z jog gomb megnyomására, vagy Z tengelyt választva a kézikerék mozgatóhoz*, az A és C tengelyek pozícióit figyelembe véve *mozog az asztal síkjára merőleges irányban*.

CN_ABSOFF: Nem használt

CN_TEST: Teszt üzemmód

A CP_TEST teszt üzemmód kérés nyugtázó jele.

Amikor a PLC bekapcsolja a teszt üzemmód kérés jelet CP_TEST=1, a csatornakezelő megvizsgálja, hogy

- a CN_TEST jel állapota megváltoztatható-e, azaz a vezérlő nincs-e az automata (CN_AUTO=1), vagy MDI (CN_MDI=1) üzemmódok valamelyikében,
- ha nincs, a CN_TEST jelző állapotát az ellenkezőjére változtatja.

A CN_TEST jelző 1 állapotában a PLC bekapcsolhatja a teszt üzemmód gomb lámpáját (pl: ML_TEST).

A CN_TEST jelző 1 állapotában a csatornakezelő

- végrehajtja az interpolációt, minden mondatot, az előtoló mondatokat (G1, G2, G3, G33) is, emelt előtolással, az override kapcsoló állásának függvényében exponenciálisan növelve a sebességet,
- semmilyen mozgásparancsot nem ad át a pozíciószabályozó körnek, így a tengelyek nem mozognak,
- semmilyen funkciót nem ad át a PLC-nek: nem adja ki a CN_MnSTB, CN_SSTB, CN_TSTB, CN_AUXnSTB beíró jeleket a PLC felé.

CN_MLCK: Gép zárva üzemmód

A CP_MLCK gép zárva üzemmód kérés nyugtázó jele.

Amikor a PLC bekapcsolja a gép zárva üzemmód kérés jelet CP_MLCK=1, a csatornakezelő megvizsgálja, hogy

- a CN_MLCK jel állapota megváltoztatható-e, azaz a vezérlő nincs-e az automata (CN_AUTO=1), vagy MDI (CN_MDI=1) üzemmódok valamelyikében,
- ha nincs, a CN_MLCK jelző állapotát az ellenkezőjére változtatja.

A CN_MLCK jelző 1 állapotában a PLC bekapcsolhatja a gép zárva üzemmód gomb lámpáját (pl: ML_MLCK).

A CN_MLCK jelző 1 állapotában a csatornakezelő

- végrehajtja az interpolációt, minden mondatot a programozott előtolással, az override kapcsolók (előtolás, gyorsmenet) figyelembe vételével,
- ha a száraz futás be van kapcsolva CN_DRRUN=1, az előtoló mondatokat (G1, G2, G3, G33) az N0305 Max Feed paraméteren meghatározott emelt előtolással hajtja végre,

- semmilyen mozgásparancsot nem ad át a pozíciószabályozó körnek, így a tengelyek nem mozognak,
- semmilyen funkciót nem ad át a PLC-nek: nem adja ki a CN_MnSTB, CN_SSTB, CN_TSTB, CN_AUXnSTB beíró jeleket a PLC felé.

CN_DRRUN: Száraz futás üzem

A CP_DRRUN száraz futás kérés nyugtázó jele.

Amikor a PLC bekapcsolja a száraz futás kérés jelet CP_DRRUN=1, a csatorna kezelő megvizsgálja, hogy

- a CN_DRRUN jel állapota megváltoztatható-e, azaz a vezérlő nincs-e az automata (CN_AUTO=1), vagy MDI (CN_MDI=1) üzemmódok valamelyikében,
- ha nincs, a CN_DRRUN jelző állapotát az ellenkezőjére változtatja.

A CN_DRRUN jelző 1 állapotában a PLC bekapcsolhatja a száraz futás üzemmód gomb lámpáját (pl: ML_DRRUN).

A CN_DRRUN jelző 1 állapotában a csatorna kezelő

- az előtoló mondatokat (G1, G2, G3, G33) az N0305 Max Feed paraméteren meghatározott emelt előtolással hajtja végre,
- minden mozgásparancsot kiad a pozíciószabályozó körnek, így a tengelyek mozognak, feltéve, hogy nincs a gép zárva (CN_MLCK=0),
- minden funkciót átad a PLC-nek, feltéve, hogy a gép sincs zárva (CN_MLCK=0), és a funkció sincs zárva (CN_MLCK=0).

CN_BKRST: Mondat újra állapot

A CP_BKRST mondat újra kérés nyugtázó jele.

Amikor a PLC bekapcsolja a mondat újra kérés jelet CP_BKRST=1, a csatorna kezelő megvizsgálja, hogy

- van-e automata végrehajtásban megszakított program, azaz CN_INTD=1,
- ha igen bekapcsolja a CN_BKRST jelzőt,

A CN_BKRST jelző 1 állapotában a PLC bekapcsolhatja a mondat újra gomb lámpáját (pl: ML_BKRST).

Automata üzemmódban start hatására a csatorna kezelő

- kiadja a PLC-nek azokat a funkciókat, amiket a megszakítás pillanatáig még nem hajtott végre a beírójelekkel az átadóregisztereken keresztül,
- visszaáll a megszakított mondat kezdőpontjára,
- majd innen folytatja a megmunkálást.

CN_BKRET: Mondat vissza állapot

A CP_BKRET mondat vissza kérés nyugtázó jele.

Amikor a PLC bekapcsolja a mondat vissza kérés jelet CP_BKRET=1, a csatorna kezelő megvizsgálja, hogy

- van-e automata végrehajtásban megszakított program, azaz CN_INTD=1,

- ha igen bekapcsolja a CN_BKRET jelzőt,
A CN_BKRET jelző 1 állapotában a PLC bekapcsolhatja a mondat vissza gomb lámpáját (pl: ML_BKRET).
- Automata üzemmódban start hatására a csatornakezelő
 - kiadja a PLC-nek azokat a funkciókat, amiket a megszakítás pillanatáig még nem hajtott végre a beírójelekkel az átadóregisztereken keresztül,
 - visszaáll a megszakított mondat megszakítási pozíciójára,
 - majd innen folytatja a megmunkálást.

CN_AHND: Kézikerékkel történő nullponteltolás beváltva

A CP_AHND kézikerekkel történő nullponteltolás kérés nyugtázó jele. Automata és Kézi adatbeviteli (MDI) üzemmódban lehet beváltani., akár start állapotban is. Ekkor a kézikerek üzemmódban megszokott módon a tengely és lépésnagyság választást a PLC-ben engedélyezni kell. A lépésnagyságot, biztonsági okokból célszerű korlátozni 0.001, esetleg 0.01 mm lépésre. A kézikerek hatására a kiválasztott tengely mozog, az elmozdulást nem az abszolút pozícióban, hanem a Nullpont tábla Kézikerék sorában veszi figyelembe. A kézikerekkel bevitt nullponteltolás M30-ra és reset hatására törlődik.

CN_WMCA XF: Alapelforgatással párhuzamos irányú előtolás bekapcsolva

Jog, Inkrementális jog, vagy Kézikerék üzemmódban a PLC a CP_WMCA XF jelzőn kérheti, hogy a tengelyek kézi mozgatása az eredeti irányok, vagy az alapelforgatás szerint történjen. A CN_WMCA XF jelző a kérés nyugtázójele. Ha a jelző
 =0: a kézi mozgatás az eredeti tengelyirányok szerint,
 =1: a kézi mozgatás az elforgatott tengelyirányok szerint történik.

CN_1100: #1100 makrováltozó értéke (bit)

CN_1101: #1101 makrováltozó értéke (bit)

CN_1102: #1102 makrováltozó értéke (bit)

CN_1103: #1103 makrováltozó értéke (bit)

CN_1104: #1104 makrováltozó értéke (bit)

CN_1105: #1105 makrováltozó értéke (bit)

CN_1106: #1106 makrováltozó értéke (bit)

CN_1107: #1107 makrováltozó értéke (bit)

CN_1108: #1108 makrováltozó értéke (bit)

CN_1109: #1109 makrováltozó értéke (bit)

CN_1110: #1110 makrováltozó értéke (bit)

CN_1111: #1111 makrováltozó értéke (bit)

CN_1112: #1112 makrováltozó értéke (bit)

CN_1113: #1113 makrováltozó értéke (bit)

CN_1114: #1114 makrováltozó értéke (bit)

CN_1115: #1115 makrováltozó értéke (bit)
CN_1116: #1116 makrováltozó értéke (bit)
CN_1117: #1117 makrováltozó értéke (bit)
CN_1118: #1118 makrováltozó értéke (bit)
CN_1119: #1119 makrováltozó értéke (bit)
CN_1120: #1120 makrováltozó értéke (bit)
CN_1121: #1121 makrováltozó értéke (bit)
CN_1122: #1122 makrováltozó értéke (bit)
CN_1123: #1123 makrováltozó értéke (bit)
CN_1124: #1124 makrováltozó értéke (bit)
CN_1125: #1125 makrováltozó értéke (bit)
CN_1126: #1126 makrováltozó értéke (bit)
CN_1127: #1127 makrováltozó értéke (bit)
CN_1128: #1128 makrováltozó értéke (bit)
CN_1129: #1129 makrováltozó értéke (bit)
CN_1130: #1130 makrováltozó értéke (bit)
CN_1131: #1131 makrováltozó értéke (bit)

A felhasználó alkatrészprogramból makrováltozóknak történő értékadással beállíthat, illetve törölhet bites PLC jelzőket. Csatornánként 32 jelzőbit áll a felhasználó rendelkezésére a PLC-vel történő kommunikáció számára: #1100, #1101, ..., #1131.

Például az alkatrészprogramba írt

#1109=1

utasítás beállítja a CN_1109=1 PLC jelzőt. A

#1109=0

utasítás törli a CN_1109=0 PLC jelzőt.

A PLC-től az NC-be menő bites csatornaváltozók

CP_START: Start kérés

Amikor a kezelő megnyomja a start gombot, a PLC programnak meg kell vizsgálnia, hogy a gép oldaláról nincs-e akadálya a start kiváltásának. Ha nincs, a CP_START start kérés jelzőt 1-be írja, amivel a csatorna kezelőtől kéri a start állapotot.

A CP_START=1 állapotában a csatorna kezelő megvizsgálja, hogy indítható-e megmunkálás a csatornában. Ha igen a CN_START=1 állapottal nyugtázza a kérést.

NCT gépi kezelőpanel használata esetén az MB_START jelző adja a start gomb állapotát.

CP_STOP: Stop kérés

Amikor a kezelő megnyomja a stop gombot a PLC programnak meg kell vizsgálnia, hogy a gép oldaláról nincs-e akadálya a stop kiváltásának. Ha nincs, a CP_STOP stop kérés jelzőt 1-be írja, amivel a csatorna kezelőtől kéri a stop állapotot.

A CP_STOP=1 állapotában a csatorna kezelő megvizsgálja, hogy leállítható-e a megmunkálás a csatornában. Ha igen a CN_STOP=1 állapottal nyugtázza a kérést.

NCT gépi kezelőpanel használata esetén az MB_STOP jelző adja a stop gomb állapotát.

CP_JOG: Mozgatás üzemmód kérés

CP_INCR: Léptetés üzemmód kérés

CP_HNDL: Kézikerék üzemmód kérés

CP_REFP: Refpontfelvétel üzemmód kérés

CP_EDIT: Szerkesztés üzemmód kérés

CP_AUTO: Automata üzemmód kérés

CP_MDI: Kézi adatbeviteli üzemmód kérés

Amikor a kezelő megnyomja valamelyik üzemmódváltó gombot, a PLC programnak meg kell vizsgálnia, hogy a gép oldaláról nincs-e akadálya az üzemmódváltásnak. Ha nincs, a lenyomott üzemmód gombhoz tartozó CP_xxx üzemmód kérés jelzőt 1-be írja.

A CP_xxx=1 állapotában a csatorna kezelő megvizsgálja, hogy beváltható-e a kért üzemmód. Ha igen a megfelelő üzemmód jelzőjének CN_xxx=1 állapotával nyugtázza a kérést.

NCT gépi kezelőpanel használata esetén az MB_JOG, MB_INCR, MB_HNDL, MB_REFP, MB_EDIT, MB_AUTO és az MB_MDI jelzők adják az üzemmódváltó gombok állapotát.

CP_JOGRAP: Jog mozgató gyorsmenettel

Amikor a kezelő megnyomja a jog gyorsmenet gombot, a PLC programnak meg kell vizsgálnia, hogy a gép oldaláról nincs-e akadálya a tengelyek gyorsmenettel történő mozgatásának. Ha nincs, a CP_JOGRAP Jog mozgató gyorsmenettel jelzõt 1-be írja.

A CP_JOGRAP=1 állapotában a tengelykezelő a csatornához tartozó tengelyeket

- mozgató (jog) üzemben gyorsmenettel mozgatja, ha a megfelelő irányválasztó gombbal együtt van nyomva,
- programvégrehajtás közben a programozott F elõtölést az N0313 Feed Mult paraméteren meghatározott szorzószámmal megnöveli.

NCT gépi kezelőpanel használata esetén az MB_JOGRAP jelző adja a jog gyorsmenet gomb állapotát. A gomb lámpáját (ML_JOGRAP) a CP_JOGRAP jelző kezeli.

CP_TAXF: Szerszám irányú kézi mozgató kérés

Gombnyomás hatására a PLC a CP_TAXF jelző bekapcsolásával kéri az üzemmódot.

Az NC a CN_TAXF jelző bekapcsolásával nyugtázza a kérést. Az NC a CN_TAXF jelzõt csak valamelyik kézi üzemmódban kapcsolja be. A szerszám irányú kézi mozgató részletes leírását lásd a CN_TAXF jelző leírásánál.

CP_TRGAF: Szerszámmal merőleges kézi mozgató kérés

Gombnyomás hatására a PLC a CP_TRGAF jelző bekapcsolásával kéri az üzemmódot. Az NC a CN_TRGAF jelző bekapcsolásával nyugtázza a kérést. Az NC a CN_TRGAF jelzõt csak valamelyik kézi üzemmódban kapcsolja be. A szerszámmal merőleges kézi mozgató részletes leírását lásd a CN_TRGAF jelző leírásánál.

CP_TTCRF: Szerszám középpont körüli kézi mozgató kérés

Gombnyomás hatására a PLC a CP_TTCRF jelző bekapcsolásával kéri az üzemmódot. Az NC a CN_TTCRF jelző bekapcsolásával nyugtázza a kérést. Az NC a CN_TTCRF jelzõt csak valamelyik kézi üzemmódban kapcsolja be. A Szerszám középpont körüli kézi mozgató részletes leírását lásd a CN_TTCRF jelző leírásánál.

CP_TBLB: Asztal szerinti kézi mozgató kérés

Gombnyomás hatására a PLC a CP_TBLB jelző bekapcsolásával kéri az üzemmódot. Az NC a CN_TBLB jelző bekapcsolásával nyugtázza a kérést. Az NC a CN_TBLB jelzõt csak valamelyik kézi üzemmódban kapcsolja be. A Szerszám középpont körüli kézi mozgató részletes leírását lásd a CN_TBLB jelző leírásánál.

CP_INTDREQ: Nem használt

CP_TLCM: Hosszkorrekció bemérés üzemmód be

A hosszkorrekció bemérés üzem olyan eszterga csatornában használható, amelyekre szerszámbemérőt szereltek. Az hosszkorrekció bemérés üzemmódot a PLC a CP_TLCM jelző 1-be írásával kapcsolja be.

A jelző bekapcsolása történhet a **bemérőkar kihajtásakor**, vagy egy tetszőleges, a PLC program által meghatározott **nyomógomb** hatására.

Amikor a PLC a CP_TLCM bitet bekapcsolja

- a csatorna automatikusan Jog üzembe kapcsol: CN_JOG=1. Addig az üzemmódot nem lehet elhagyni, amíg a CP_TLCM jelző állapota IGAZ.
- A jog gombbal történő mozgás előtolását az interpolátor ilyenkor a N0319 T Meas Feed paraméterről veszi. Egyszerre csak egy tengelyen lehet mozogni, azaz kizárja több tengely egyidejű mozgását.
- a kijelző oldal beváltja a jelző csatornához tartozó Eltolás képernyőt a Korrekció bemérés bekapcsolt állapotával.

CP_S2TS: S2-höz tartozó hossz bemérő választása

Egy csatorna max. 2 szerszámbemérő jelét tudja kezelni. Két szerszámbemérő használható pl. ellenorsós esztergákon.

A mérőrendszer a CP_S2TS PLC bit állása alapján választja ki melyik tapintót kell használnia. Ha CP_S2TS értéke

- =0: a N3012 Sensor Input of Tool Setter S1 paraméteren kiválasztott tapintót használja,
- =1: a N3013 Sensor Input of Tool Setter S2 paraméteren kiválasztott tapintót használja.

CP_WPCM: Munkadarab nullpont bemérés üzemmód be

A munkadarab nullpont bemérés üzem eszterga csatornában használható és általában a darab homlokfelületének Z irányú eltolását méri. A munkadarab nullpont bemérés üzemmódot a PLC a CP_WPCM jelző 1-be írásával kapcsolja be.

A jelző bekapcsolása történhet a tapintó bekapcsolásakor, vagy egy tetszőleges, a PLC program által meghatározott nyomógomb hatására.

Amikor a PLC a CP_WPCM bitet bekapcsolja

- az NC oldal automatikusan Jog üzembe kapcsol: CN_JOG=1. Addig az üzemmódot nem lehet elhagyni, amíg a CP_WPCM jelző állapota IGAZ.
- A jog gombbal történő mozgás előtolását az interpolátor ilyenkor a N0319 T Meas Feed paraméterről veszi. Egyszerre csak egy tengelyen lehet mozogni, azaz kizárja több tengely egyidejű mozgását.
- a kijelző oldal beváltja a jelző csatornához tartozó Eltolás képernyőt a Nullpont bemérés bekapcsolt állapotával.

CP_S2WS: S2-höz tartozó nullpont bemérő választása

Egy csatorna max. 2 munkadarab nullpont bemérő jelét tudja kezelni. Két munkadarab nullpont bemérő használható pl. ellenorsós esztergákon.

A mérőrendszer a CP_S2WS PLC bit állása alapján választja ki melyik tapintót kell használni. Ha CP_S2WS értéke

=0: a N3014 Sensor Input of Workpiece Setter S1 paraméteren kiválasztott tapintót használja,

=1: a N3015 Sensor Input of Workpiece Setter S2 paraméteren kiválasztott tapintót használja.

Ez utóbbit akkor, ha az egyrevolveres ellenorsós gépre minden orsóhoz szerelnek egy nullpontbemérő tapintót.

CP_AHND: Kézikerékkel történő nullponteltolás kérés

Gombnyomás hatására a PLC a CP_AHND jelző bekapcsolásával kéri az üzemmódot. Az NC a CN_AHND jelző bekapcsolásával nyugtázza a kérést. Az NC a CN_AHND jelzőt csak Automata, vagy Kézi adatbeviteli (MDI) üzemmódban kapcsolja be. A Kézikerékkel történő nullponteltolás részletes leírását lásd a CN_AHND jelző leírásánál.

CP_WMCA XF: Alapelforgatással párhuzamos irányú előtolás kérés

Jog, Inkrementális jog, vagy Kézikerék üzemmódban a PLC kérheti, hogy a tengelyek kézi mozgatása az alapelforgatás szerint történjen:

=0: kézi mozgatás kérés az eredeti tengelyirányok szerint,

=1: kézi mozgatás kérés az elforgatott tengelyirányok szerint.

CP_SGLBK: Mondatonkénti üzem be

A PLC programnak, ha megnyomják a mondatonkénti üzem gombot a CP_SGLBK jelző értékét ellenkezőjére kell váltania.

A CP_SGLBK=1 állapotban a csatornakezelő minden mondat végrehajtása után stop állapotot vesz fel CN_STOP=1, azaz leállítja a megmunkálást.

NCT gépi kezelőpanel használata esetén az MB_SGLBK jelző adja a mondatonkénti üzem gomb állapotát. A gomb lámpáját (ML_SGLBK) a CP_SGLBK jelző kezeli.

CP_CNDSP: Feltételes stop be

A PLC programnak, ha megnyomják a feltételes stop gombot a CP_CNDSP jelző értékét ellenkezőjére kell váltania.

Ha a program végrehajtása M01 kódra fut a csatornakezelő megvizsgálja a CP_CNDSP jelző állapotát. CP_CNDSP=1 állapotban leállítja a megmunkálást és stop állapotot vesz fel CN_STOP=1. Ellenkező esetben CP_CNDSP=0 nincs leállítás. NCT gépi kezelőpanel használata esetén az MB_CNDSP jelző adja a feltételes stop gomb állapotát. A gomb lámpáját (ML_CNDSP) a CP_CNDSP jelző kezeli.

CP_TEST: Teszt üzemmód kérés

Amikor a kezelő megnyomja a teszt üzem gombot, a PLC program a CP_TEST teszt üzemmód kérés jelzőt 1-be írja, amivel a csatorna kezelőtől kéri a teszt üzemet.

A CP_TEST=1 állapotban a csatorna kezelő megvizsgálja, hogy bekapcsolható-e a teszt üzem a csatornában. Ha igen a CN_TEST=1 állapottal nyugtázza a kérést.

NCT gépi kezelőpanel használata esetén az MB_TEST jelző adja a teszt gomb állapotát.

CP_MLCK: Gép zárva üzemmód kérés

Amikor a kezelő megnyomja a gép zárva gombot, a PLC program a CP_MLCK gép zárva üzemmód kérés jelzőt 1-be írja, amivel a csatorna kezelőtől kéri a gép zárva üzemet.

A CP_MLCK=1 állapotban a csatorna kezelő megvizsgálja, hogy bekapcsolható-e a gép zárva üzem a csatornában. Ha igen a CN_MLCK=1 állapottal nyugtázza a kérést.

NCT gépi kezelőpanel használata esetén az MB_MLCK jelző adja a gép zárva gomb állapotát.

CP_DRRUN: Száraz futás kérés

Amikor a kezelő megnyomja a száraz futás gombot, a PLC program a CP_DRRUN száraz futás kérés jelzőt 1-be írja, amivel a csatorna kezelőtől kéri a száraz futás üzemet.

A CP_DRRUN=1 állapotban a csatorna kezelő megvizsgálja, hogy bekapcsolható-e a száraz futás üzem a csatornában. Ha igen a CN_DRRUN=1 állapottal nyugtázza a kérést.

NCT gépi kezelőpanel használata esetén az MB_DRRUN jelző adja a száraz futás gomb állapotát.

CP_BKRST: Mondat újra kérés

Amikor a kezelő megnyomja a mondat újra gombot, a PLC program a CP_BKRST mondat újra kérés jelzőt 1-be írja, amivel a csatorna kezelőtől kéri a mondat újraindítását.

A CP_BKRST=1 állapotban a csatorna kezelő megvizsgálja, hogy bekapcsolható-e a mondat újraindítás a csatornában. Ha igen a CN_BKRST=1 állapottal nyugtázza a kérést.

NCT gépi kezelőpanel használata esetén az MB_BKRST jelző adja a mondat újra gomb állapotát.

CP_BKRET: Mondat vissza kérés

Amikor a kezelő megnyomja a mondat vissza gombot, a PLC program a CP_BKRET mondat vissza kérés jelzőt 1-be írja, amivel a csatorna kezelőtől kéri a megszakítási pontra való visszatérést.

A CP_BKRET=1 állapotban a csatorna kezelő megvizsgálja, hogy bekapcsolható-e a mondatba való visszapoziicionálás a csatornában. Ha igen a CN_BKRET=1 állapottal nyugtázza a kérést.

NCT gépi kezelőpanel használata esetén az MB_BKRET jelző adja a mondat vissza gomb állapotát.

CP_FLCK: Funkció zárva kérés

Amikor a kezelő megnyomja a funkció zárva gombot, a PLC program a CP_FLCK funkció zárva kérés jelzőt 1-be írja, amivel a csatorna kezelőtől kéri a funkciókiadás zárását.

A CP_FLCK=1 állapotban a csatorna kezelő megvizsgálja, hogy bekapcsolható-e a funkció zárva üzem a csatornában. Ha igen a CN_FLCK=1 állapottal nyugtázza a kérést.

NCT gépi kezelőpanel használata esetén az MB_FLCK jelző adja a funkció zárva gomb állapotát.

CP_ABSOFF: Nem használt

CP_CNDBK_1: Feltételes mondat 1 be

CP_CNDBK_2: Feltételes mondat 2 be

CP_CNDBK_3: Feltételes mondat 3 be

CP_CNDBK_4: Feltételes mondat 4 be

CP_CNDBK_5: Feltételes mondat 5 be

CP_CNDBK_6: Feltételes mondat 6 be

CP_CNDBK_7: Feltételes mondat 7 be

CP_CNDBK_8: Feltételes mondat 8 be

Csatornánként 8 különböző, mondatkihagyási feltételt lehet beállítani. Ha az alkatrész-programban egy mondat a /n (n=1, ..., 8) utasítással kezdődik a csatorna kezelő a mondat végrehajtása előtt megvizsgálja, hogy az n-edik feltétel teljesül-e, vagyis a CP_CNDBK_n jelző állapota 1-e. Ha

- CP_CNDBK_n=0 a mondatot végrehajtja,
- CP_CNDBK_n=1 a mondatot nem hajtja végre, a következőre lép.

NCT gépi kezelőpanel használata esetén az MB_CNDBK jelző adja a feltételes mondat gomb állapotát. A gomb lámpáját (ML_CNDBK) a PLC programozó által a gombhoz kiválasztott CP_CNDBK_n jelző kezeli.

CP_FIN: Kész jel, összes funkció végrehajtva

A PLC program a CP_FIN=1 állapottal jelzi a csatorna kezelőnek, hogy az összes funkciót végrehajtotta. CP_FIN=1 állapotban a csatorna kezelő

- start állapotban előveszi a következő mondatot a pufferből és végrejtja,
- mondatonkénti végrehajtás esetén stop állapotot vesz fel, és startra vár,

– program végén, azaz üres puffér esetén, törli a start állapotot.

A PLC-ben a CP_FIN jelet akkor lehet bekapcsolni, ha az összes funkció nyugtázó jel (CP_MnACK, CP_SACK, CP_TACK, CP_AUXnACK) állapota 1 és a PLC program oldaláról nincs egyéb ok arra, hogy az alkatrészprogram végrehajtását felfüggesztjük.

CP_RST: Nem használt

CP_RSTREW: Nem használt

CP_CSREQ: Nem használt

CP_NOWT: Nem kell összeváró M kódra várni másik csatornától

Többcsatornás vezérlőkön szükség van arra, hogy az egyik csatornában futó alkatrészprogram végrehajtását felfüggesztjük addig, amíg egy vagy több másik csatorna a program végrehajtásában egy meghatározott pontra nem ér. Ezt nevezzük a csatornák szinkronizálásának, vagy összevárásának. A csatornák szinkronizálását M kódokkal lehet végezni. Az N2201 Waiting M Codes Min és az N2202 Waiting M Codes Max paramétereken kijelölhetünk egy M kód tartományt.

Az összevárást teljes egészében az NC végzi, azzal a PLC-nek nem kell foglalkozni. Ha egy programot úgy írtunk meg, hogy másik csatorna, vagy csatornák szinkronizációs M kódjára várjon, de a programot csak önmagában, a többi csatornában futó programhoz történő szinkronizálás nélkül akarjuk futtatni, a csatornakezelő

– CP_NOWT=1 állapotban átugorja a szinkronizációs M kódokat, azaz nem vár másik csatornára.

CP_MINT: Megszakítási makró hívás

CP_TMREN: Szabad felhasználású időmérő engedélyezése

A vezérlő Idők/számlálók ablakának Időzítő sorában kiolvasható egy idő, amely egy szabadfelhasználású időzítő értékét mutatja nap/óra/perc/másodperc/msec formában.

A szabadfelhasználású időzítőt a PLC a

- CP_TMREN=1 állapottal indítja,
- CP_TMREN=0 állapottal leállítja.

Az időzítő értéke átírható/kiolvasható

- kezelőpanelről,
- alkatrészprogramból, #3001 makrováltozón keresztül.

CP_TSBD: Nem használt

CP_EGBRRQ: Szerszám kikapás kérés EGB üzemben

Ha a csatornakezelő G81.8 fogazási funkciót (elektronikus hajtómű) hajt végre 1-be kapcsolja a CN_EGBMD jelzőt.

A CN_EGBMD jelző 1 állapotában a PLC kérheti a szerszám kikapását a CP_EGBRRQ jelző 1-be írásával. Ekkor az NC

- az N1804 Retr. Dist. paraméteren meghatározott irányban és távolságra kihúzza a szerszámot,
- majd a kihúzás végén a csatornakezelő 1-be írja az CN_RTRFIN jelzőt.

Ezután a PLC-nek ki kell kapcsolnia CP_EGBRRQ=0 jelzőt.

CP_M1ACK: 1. M funkció végrehajtva

CP_M2ACK: 2. M funkció végrehajtva

CP_M3ACK: 3. M funkció végrehajtva

CP_M4ACK: 4. M funkció végrehajtva

CP_M5ACK: 5. M funkció végrehajtva

CP_M6ACK: 6. M funkció végrehajtva

CP_M7ACK: 7. M funkció végrehajtva

CP_M8ACK: 8. M funkció végrehajtva

Alkatrészprogramban egy mondatba maximum 8 különböző M kódot írhatunk, azaz egy mondatban 8 különböző M kódot tud az NC átadni a PLC-nek.

Ha az alkatrészprogramba M funkciót programoztak, a csatornakezelő a CN_MnSTB beírójelet 1 PLC ciklus időtartamra 1-be írja. A beírójel hatására a kapott kód alapján a funkciót a PLC dekódolja.

Ha a kapott kód a gépen létező funkció takar, a PLC

- CP_MnACK=0 törli a beírójelnek megfelelő nyugtázó jelzőt.

A funkció végrehajtása után a PLC

- CP_MnACK=1 beírja a nyugtázó jelzőt.

A CP_MnACK jelzők állapota alapján tudja nyilván a csatornakezelő, hogy a végrehajtás alatt álló mondatból mely M funkciók vannak végrehajtva. A programot megszakítva, majd újraindítva a végre nem hajtott funkciókat újra kiadja a csatornakezelő a PLC-nek.

A nyugtázó jelek állapotát bekapcsolás után a PLC programnak kell inicializálni (CP_MnACK=1).

CP_SACK: S funkció végrehajtva

Ha az alkatrészprogramban S funkciót programoztak, a csatornakezelő a CN_SSTB jelzőt 1 PLC ciklus időtartamra 1-be írja. A beírójel hatására a PLC

- CP_SACK=0 törli az S funkció nyugtázó jelzőt

A funkció végrehajtása után a PLC

- CP_SACK=1 beírja a nyugtázó jelzőt.

A CP_SACK jelzők állapota alapján tatja nyilván a csatorna kezelő, hogy a végrehajtás alatt álló mondatból végrehajtotta-e az S funkciót. A programot megszakítva, majd újraindítva a végre nem hajtott S funkciót újra kiadja a csatorna kezelő a PLC-nek. A nyugtázó jel állapotát bekapcsolás után a PLC programnak kell inicializálni (CP_SACK=1).

CP_TACK: T funkció végrehajtva

Ha az alkatrészprogramban T funkciót programoztak, a csatorna kezelő a CN_TSTB jelzőt 1 PLC ciklus időtartamra 1-be írja. A beírójel hatására a PLC

- CP_TACK=0 törli a T funkció nyugtázó jelzőt.

A funkció végrehajtása után a PLC

- CP_TACK=1 beírja a nyugtázó jelzőt.

A CP_TACK jelzők állapota alapján tatja nyilván a csatorna kezelő, hogy a végrehajtás alatt álló mondatból végrehajtotta-e a T funkciót. A programot megszakítva, majd újraindítva a végre nem hajtott T funkciót újra kiadja a csatorna kezelő a PLC-nek.

A nyugtázó jel állapotát bekapcsolás után a PLC programnak kell inicializálni (CP_TACK=1).

CP_AUX1ACK: 1. segédfunkció végrehajtva

CP_AUX2ACK: 2. segédfunkció végrehajtva

CP_AUX3ACK: 3. segédfunkció végrehajtva

Ha az alkatrészprogramban segéd funkciót programoztak, a csatorna kezelő a segéd-funkcióhoz tartozó CN_AUXnSTB jelzőt 1 PLC ciklus időtartamra 1-be írja. A beírójel hatására a PLC

- CP_AUXnACK=0 törli a megfelelő segédfunkció nyugtázó jelzőjét.

A funkció végrehajtása után a PLC

- CP_AUXnACK=1 beírja a megfelelő nyugtázó jelzőt.

A CP_AUXnACK jelzők állapota alapján tatja nyilván a csatorna kezelő, hogy a végrehajtás alatt álló mondatból végrehajtotta-e az adott segédfunkciót. A programot megszakítva, majd újraindítva a végre nem hajtott segédfunkciókat újra kiadja a csatorna kezelő a PLC-nek.

A nyugtázó jelek állapotát bekapcsolás után a PLC programnak kell inicializálni (CP_AUXnACK=1).

CP_GACK: Nem használt

CP_HOLD: Mozgás állj minden tengelyre a csatornában

Ha a PLC a jelzőt 1-be kapcsolja a csatorna kezelő a csatornához tartozó összes tengely mozgását feltétel nélkül leállítja.

A CP_STOP jelzőtől abban különbözik, hogy míg az override és stop tiltás állapotban (G63) a stop hatástalan, addig a CP_HOLD ilyenkor is hatásos.

A fentiek miatt menetvágáskor, vagy menetfűráskor (G74, G84) az orsó leállításakor a CP_HOLD jelzőt kell használni az előtolás megállítására, illetve az előtolást az orsó leállításával lehet megállítani a CP_HOLD jelzőn keresztül.

CP_CBFEN: Forgácsoló mondat előtolás engedélyezés

A csatorna kezelő a PLC-től az előtoló mozgás engedélyezését a CN_CBFR=1 jelző bekapcsolásával kéri.

Az előtolás (mozgás) addig nem indul el, amíg a PLC a CP_CBFEN=1 jelzőn azt nem engedélyezi. Az előtolás engedélyezése különböző feltételekhez köthető a PLC program részéről, például orsó forgáshoz forgácsoló gépeken, lézer vagy láng bekapcsolásához vágógépeken stb.

CP_FHNDL: Előtolás kézikérékről üzem be

Ha a CP_FHNDL jelző be van kapcsolva programvégrehajtás alatt, az interpolátor az adott csatornában nem a programozott előtolásnak megfelelően (G94, vagy G95 F) mozog, hanem a kézikérékről jövő impulzusok szerint. A mozgás sebessége függ

- az inkrementválasztás nagyságától
- a kézikerek forgatás sebességétől.

A CP_HNDLSn jelzők segítségével kell kiválasztani, hogy melyik kézikereket használja a csatorna.

A funkció felhasználható pl. többcsatornás gépeken a programok belövésére ütközés-vizsgálat céljára.

CP_OVC: Nem használt

CP_HNDLS1: Előtolás az 1. kézikérékről

CP_HNDLS2: Előtolás a 2. kézikérékről

CP_HNDLS3: Előtolás a 3. kézikérékről

CP_HNDLS4: Előtolás a 4. kézikérékről

Ha az előtolás kézikérékről üzem be van kapcsolva (CP_FHNDL=1) a fenti jelzőkön lehet a csatorna kezelőnek kijelölni, hogy a lehetséges 4 kézikerek közül melyiket használja a funkcióra a megfelelő CP_HNDLSn jelző beírásával.

CP_LIM1DIS: 1. paraméteres végállás tartomány tiltása

CP_LIM2DIS: 2. paraméteres végállás tartomány tiltása

CP_LIM3DIS: 3. paraméteres végállás tartomány tiltása

A vezérlőben tengelyenként 3 különböző, paraméteren beállítható végállástartományt lehet engedélyezni az N1000 Range Enable paraméter #0 RE1, #1 RE2, #2 RE3 bitjein.

A PLC program egy adott csatornához tartozó összes tengelyen kikapcsolhatja a 3 beállítható végállástartomány valamelyikét a megfelelő CP_LIMnDIS jelző 1-be írásával.

☞ **Figyelem!** Ekkor a vezérlő nem kezeli a paraméteres végállásokat!

CP_LIMSEL: 1B paraméteres végállás tartomány választása minden tengelyre

Csatornánkénti paraméteres végállás választás PLC-ből a csatorna minden tengelyére.

Ha az

N1001 StrkCont paraméter #4 ABA=1

az adott csatornában a CP_LIMSEL PLC jelző hatásos. A CP_LIMSEL jelző mondja meg, hogy az 1-es tartományban melyik végállás paramétercsoport

CP_LIMSEL=0: 1A,

CP_LIMSEL=1: 1B

legyen érvényes a csatorna összes tengelyére, mindkét irányban.

A végállás választás változtatását a tengelyek álló állapotában kell végezni.

☞ **Lásd még:** AP_LIMSELP és AP_LIMSELN jelzőket.

CP_CHOPON: Nem használt

CP_SGOEN: Második geometria korrekciós tábla engedélyezése

A jelző kizárólag eszterga csatornában hatásos.

Kiterített szerszámozást használó esztergákon célszerű bevezetni egy második geometriai korrekciós tárat. A második geometriai korrekciós tár ugyanolyan hosszú, mint az első, és a geometriai korrekcióra való hivatkozás során a második geometriai korrekciós tárban tárolt értékeket, meghatározott feltételek teljesülése esetén, hozzá kell adni az első tárban tárolt értékekhez.

A második geometriai korrekciós tárban a szerszámtartók gépi koordináta rendszerben elfoglalt helyzetét X, Y, Z-t lehet megadni. Ezzel lehetővé válik, hogy az első geometriai korrekciós tárban a szerszám valódi X, Y, Z hosszát adjuk meg, azaz az első geometriai tárban a külső szerszámbemérőn mért kinyúlási értékeket közvetlen meg lehet adni. A figyelembe vett korrekció értéke tehát:

korrekció = 1. geometriai korrekció + 2. geometriai korrekció + kopás korrekció

Ha a N1414 Comp. Config on Lathes paraméter #4 SGC bitje:

=0: a második geometriai korrekciós tár nem létezik, nem kerül kijelzésre,

=1: a második geometriai korrekciós tár létezik, kijelzésre kerül, és a korrekciók értékének figyelembe vételét PLC jelzők szabályozzák.

A CP_SGOEN második geometria korrekciós tábla engedélyezése jelző

=0 állapotában a második geometriai korrekciót nem veszi figyelembe,

=1 állapotában a második geometriai korrekciót figyelembe veszi

a csatorna kezelő a korrekció T kóddal történő lehívásakor.

Lásd még CP_SGOX, CP_SGOY, CP_SGOZ jelzőket.

A CP_SGOEN használható, ha például egy eszterga csatornán belül van revolverfej is, kiterített szerszámozás is, és a revolverfejben lévő szerszámokra nem kell, a kiterített szerszámokra le kell hívni a 2. geometriai korrekciót.

CP_SGOX: Második geometria korrekció figyelembe vétele az X tengelyen

CP_SGOY: Második geometria korrekció figyelembe vétele az Y tengelyen

CP_SGOZ: Második geometria korrekció figyelembe vétele az Z tengelyen

Ha a 2. geometriai korrekció figyelembe vétele CP_SGOEN=1 engedélyezve van a csatornában, a csatorna X, Y, Z tengelyein külön engedélyezhető figyelembe vételük:

CP_SGOX=1 az X tengelyen,

CP_SGOY=1 az Y tengelyen,

CP_SGOZ=1 a Z tengelyen

engedélyezi a 2. geometriai korrekció figyelembe vételét.

Lásd még CP_OSGNX, CP_OSGNY, CP_OSGNZ jelzőket.

CP_OSGNX: X korrekció ellenkező előjellel (xkorrekció=-xkorrekció)

CP_OSGNY: Y korrekció ellenkező előjellel (ykorrekció=-ykorrekció)

CP_OSGNZ: Z korrekció ellenkező előjellel (zkorrekció=-zkorrekció)

Ha a 2. geometriai korrekció figyelembe vétele CP_SGOEN=1 engedélyezve van a csatornában, és az adott tengelyen is a CP_SGOX, CP_SGOY, CP_SGOZ jelzőkkel, akkor a korrekció figyelembe vételének irányát (előjelét) is szabályozhatjuk tengelyenként:

CP_OSGNX=1 az X tengelyen ellenkező előjellel,

CP_OSGNY=1 az Y tengelyen ellenkező előjellel,

CP_OSGNZ=1 a Z tengelyen ellenkező előjellel

veszi figyelembe a korrekciót:

korrekció = -(1. geometriai korrekció + 2. geometriai korrekció + kopás korrekció)

Akkor használható, ha valamelyik tengelyre szerelt szerszám iránya megegyezik a tengely pozitív irányával.

CP_ROVLD: Gyorsmeneti mondatok átlapolásának kikapcsolása

A jelző akkor használható, amikor az N0407 Acc Contr paraméter #0 ROL bitjén a gyorsmeneti mozgások átlapolása be van kapcsolva.

Ha az átlapolást bizonyos esetekben ki kell kapcsolni, a PLC program a

CP_ROVLD=1 jelző bekapcsolásával megteheti azt.

CP_RLSOT3: Nem használt

CP_1000: #1000 makrováltozó (bit)

CP_1001: #1001 makrováltozó (bit)

CP_1002: #1002 makrováltozó (bit)

CP_1003: #1003 makrováltozó (bit)

CP_1004: #1004 makrováltozó (bit)

CP_1005: #1005 makrováltozó (bit)

CP_1006: #1006 makrováltozó (bit)
CP_1007: #1007 makrováltozó (bit)
CP_1008: #1008 makrováltozó (bit)
CP_1009: #1009 makrováltozó (bit)
CP_1010: #1010 makrováltozó (bit)
CP_1011: #1011 makrováltozó (bit)
CP_1012: #1012 makrováltozó (bit)
CP_1013: #1013 makrováltozó (bit)
CP_1014: #1014 makrováltozó (bit)
CP_1015: #1015 makrováltozó (bit)
CP_1016: #1016 makrováltozó (bit)
CP_1017: #1017 makrováltozó (bit)
CP_1018: #1018 makrováltozó (bit)
CP_1019: #1019 makrováltozó (bit)
CP_1020: #1020 makrováltozó (bit)
CP_1021: #1021 makrováltozó (bit)
CP_1022: #1022 makrováltozó (bit)
CP_1023: #1023 makrováltozó (bit)
CP_1024: #1024 makrováltozó (bit)
CP_1025: #1025 makrováltozó (bit)
CP_1026: #1026 makrováltozó (bit)
CP_1027: #1027 makrováltozó (bit)
CP_1028: #1028 makrováltozó (bit)
CP_1029: #1029 makrováltozó (bit)
CP_1030: #1030 makrováltozó (bit)
CP_1031: #1031 makrováltozó (bit)

A felhasználó alkatrészprogramból makrováltozókon keresztül lekérdezhet bites PLC jelzőket. Csatornánként 32 jelzőbit áll a felhasználó rendelkezésére a PLC-vel történő kommunikáció számára: #1000, #1001, ..., #1031.

Például az alkatrészprogramba írt

IF #1025EQ1 GOTO30

utasítás akkor ugrik az N30 mondatra, amikor a CP_1025 jelző 1.

7.15.2 A duplaszavas csatornaváltozók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
CN_1	Az NC-ből PLC-nek átadott csatorna kezelő bitek DWORD1	CP_1	A PLC-ből NC-nek átadott csatorna kezelő bitek DWORD1
CN_2	Az NC-ből PLC-nek átadott csatorna kezelő bitek DWORD2	CP_2	A PLC-ből NC-nek átadott csatorna kezelő bitek DWORD2
CN_3	Az NC-ből PLC-nek átadott csatorna kezelő bitek DWORD3	CP_3	A PLC-ből NC-nek átadott csatorna kezelő bitek DWORD3
CN_1132	#1132 makrováltozó értéke (DWORD)	CP_4	A PLC-ből NC-nek átadott csatorna kezelő bitek DWORD4
CN_M1C	1. M funkció kódja (DWORD)	CP_JOGFD	Jog előtolás kiválasztása táblázatból (DWORD)
CN_M2C	2. M funkció kódja (DWORD)	CP_SINP	Megmunkálást végző orsó száma (1,2,...) (DWORD)
CN_M3C	3. M funkció kódja (DWORD)	CP_CSAX	Nem használt
CN_M4C	4. M funkció kódja (DWORD)	CP_POLYSL	Sokszögesztergálás szolgáló orsójának száma (1,2,...) (DWORD)
CN_M5C	5. M funkció kódja (DWORD)	CP_ACTT	Az aktuális szerszám száma (DWORD)
CN_M6C	6. M funkció kódja (DWORD)	CP_MGR1	A PLC-ből kijelezni kívánt 1. M-kód (DWORD)
CN_M7C	7. M funkció kódja (DWORD)	CP_MGR2	A PLC-ből kijelezni kívánt 2. M-kód (DWORD)
CN_M8C	8. M funkció kódja (DWORD)	CP_MGR3	A PLC-ből kijelezni kívánt 3. M-kód (DWORD)
CN_SC	S funkció kódja (DWORD)	CP_MGR4	A PLC-ből kijelezni kívánt 4. M-kód (DWORD)
CN_SSEL	Orsó száma, amire az S kód vonatkozik (DWORD)	CP_MGR5	A PLC-ből kijelezni kívánt 5. M-kód (DWORD)
CN_RNGREQ	S kódhoz tartozó tartomány száma (DWORD)	CP_MGR6	A PLC-ből kijelezni kívánt 6. M-kód (DWORD)
CN_TC	T funkció kódja (DWORD)	CP_MGR7	A PLC-ből kijelezni kívánt 7. M-kód (DWORD)
CN_AUX1C	1. Segédfunkció kódja (DWORD)	CP_MGR8	A PLC-ből kijelezni kívánt 8. M-kód (DWORD)
CN_AUX2C	2. Segédfunkció kódja (DWORD)	CP_MGR9	A PLC-ből kijelezni kívánt 9. M-kód (DWORD)
CN_AUX3C	3. Segédfunkció kódja (DWORD)	CP_MGR10	A PLC-ből kijelezni kívánt 10. M-kód (DWORD)

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
CN_CTSPN	G96 esetén az orsó fordulatszáma (DWORD)	CP_MGR11	A PLC-ből kijelezni kívánt 11. M-kód (DWORD)
CN_NMAX	G96 esetén a max. fordulatszám értéke (G92 S_) (DWORD)	CP_MGR12	A PLC-ből kijelezni kívánt 12. M-kód (DWORD)
CN_MGZNO	T kódhoz tartozó magazin száma (DWORD)	CP_MGR13	A PLC-ből kijelezni kívánt 13. M-kód (DWORD)
CN_POTNO	T kódhoz tartozó zsebszám (DWORD)	CP_MGR14	A PLC-ből kijelezni kívánt 14. M-kód (DWORD)
		CP_MGR15	A PLC-ből kijelezni kívánt 15. M-kód (DWORD)
		CP_MGR16	A PLC-ből kijelezni kívánt 16. M-kód (DWORD)
		CP_1032	#1032 makrováltozó (DWORD)
		CP_OFFSNO	A szerszámbemérésnél használandó korrekció száma (DWORD)

Az NC-től a PLC felé menő duplaszavas csatornaváltozók

CN_1132: #1132 makrováltozó értéke (DWORD)

A felhasználó alkatrészprogramból makrováltozóknak történő értékadással beállíthat, illetve törölhet bites PLC jelzőket. Csatornánként 32 jelzőbit áll a felhasználó rendelkezésére a PLC-vel történő kommunikáció számára: #1100, #1101, ..., #1131. Ezeket a PLC jelzőket duplaszavasan is írhatja a #1132 makrováltozón keresztül.

A

#1132=128

utasítás a CN_1107=1 jelzőt írja be.

A PLC ugyanígy, duplaszavasan is lekérdezheti a CN_1100, CN_1101, ... CN_1131 biteket a CN_1132 változón keresztül.

CN_M1C: 1. M funkció kódja (DWORD)

CN_M2C: 2. M funkció kódja (DWORD)

CN_M3C: 3. M funkció kódja (DWORD)

CN_M4C: 4. M funkció kódja (DWORD)

CN_M5C: 5. M funkció kódja (DWORD)

CN_M6C: 6. M funkció kódja (DWORD)

CN_M7C: 7. M funkció kódja (DWORD)

CN_M8C: 8. M funkció kódja (DWORD)

Alkatrészprogramban egy mondatba maximum 8 különböző M kódot írhatunk, azaz egy mondatban 8 különböző M kódot tud az NC átadni a PLC-nek.

Ha az alkatrészprogramba M funkciót programoztak, a csatornakezelő

- a CN_MnC (n=1, 2, ..., 8) regiszterbe beírja az M kód értékét
- majd a CN_MnSTB beírójelet az NC 1 PLC ciklus időtartamra 1-be írja.

A 8 különböző M kód értékét a vezérlő 8 különböző **CN_MnC** (n=1, 2, ..., 8) regiszterbe írja. A 8 átdó regiszterhez 8 beíró jel CN_MnSTB (n=1, 2, ...8) tartozik. A beírójel hatására a kapott kód alapján a funkciót a PLC dekódolja.

Ha a kapott kód a gépen létező funkció takar, a beírójelnek megfelelő CP_MnACK nyugtázó jelzőt a PLC 0-ba állítja. A CP_MnACK nyugtázó jelzőt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

CN_SC: S funkció kódja (DWORD)

Ha az alkatrészprogramban S funkciót programoztak, a csatornakezelő

- a CN_SC regiszterbe beírja az S funkcion programozott fordulatszám értékét,
- a CN_SSEL regiszterbe beírja a hivatkozott orsó számát (1...16)
- a CN_RNGREQ regiszterbe beírja a programozott fordulatszámhoz tartozó tartomány kódját (11...18),
- majd a CN_SSTB jelzőt 1 PLC ciklus időtartamra 1-be írja.

A beírójel hatására a PLC a fenti regiszterekbe írt kódokat dekódolja és a CP_SACK nyugtázó jelet 0-ba állítja. A CP_SACK nyugtázó jelzőt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

A CN_SC regiszterben kapott kódot a funkció végrehajtása során a CN_SSEL regiszter által kijelölt orsó SP_PRG regiszterébe kell írni.

CN_SSEL: Orsó száma, amire az S kód vonatkozik (DWORD)

Ha az alkatrészprogramban S funkciót programoztak, a csatornakezelő

- a CN_SC regiszterbe beírja az S funkcion programozott fordulatszám értékét,
- a CN_SSEL regiszterbe beírja a hivatkozott orsó számát (1...16)
- a CN_RNGREQ regiszterbe beírja a programozott fordulatszámhoz tartozó tartomány kódját (11...18),
- majd a CN_SSTB jelzőt 1 PLC ciklus időtartamra 1-be írja.

A beírójel hatására a PLC a fenti regiszterekbe írt kódokat dekódolja és a CP_SACK nyugtázó jelet 0-ba állítja. A CP_SACK nyugtázó jelzöt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

A csatorna kezelő a CN_SSEL regiszterben beállított orsó számát az alkatrészprogramban történő hivatkozás szerint állítja be.

Az orsókra alkatrészprogramból S címen, vagy kiterjesztett név esetén S+2 karakteres címen hivatkozhatunk. Kiterjesztett címeken történő hivatkozás esetén az N0605 Spindle Name2 és az N0606 Spindle Name3 paramétereken adhatjuk meg a cím 2. és 3. karakterét. Az első karakter mindig kötelezően S. Ha az utolsó karakter szám, adatmegadásnál = jelet kell használni.

Ha a gépen több orsó is van és nem akarunk kiterjesztett címeket használni, hivatkozhatunk egy adott orsóra S és P címen, ahol S-en adjuk meg a fordulatszámot, P-n az orsó számát, amire az S kód vonatkozik.

A csatorna kezelő a programozott Sxx, vagy Ss Pp kód alapján megállapítja, hogy hányadik orsóra történt hivatkozás az alkatrészprogramban, és a hivatkozott orsó számát írja be a CN_SSEL regiszterbe.

Kiterjesztett címek használata esetén legyen a 3. orsó címe S3.

S3=1000

programozása esetén a megfelelő csatorna átadó regiszterei a következő adatokat kapják:

CN_SC=1000

CN_SSEL=3

S és P cím programozása esetén a fenti hivatkozás:

S1000 P3

Az átadóregiszterekbe ugyanazok az értékek kerülnek.

Az N0604 Default Spindle paraméteren csatornánként kijelölhetünk egy orsót, amire S címen hivatkozhatunk. Ha pl. a paraméter értéke 2, a 2. orsóra hivatkozhatunk S2 és S címen is. Mindkét esetben a CN_SSEL=2-t adja át a csatorna kezelő.

A PLC program a **CN_SSEL** regiszterben meghatározott orsóra ad ki minden parancsot

- M3, M4, M5, M19-et,
- az N0823 M Code for Closing S Loop paraméteren beállított hurokzárási parancsot,
- a szolga orsóra vonatkozó szinkronizálási parancsot (M kódot),
- a sokszögesztergálás szolga orsójának kiválasztására szolgáló M kódot.

Egyszerre több orsó is forgatható egy csatornában programból. Ekkor írjunk:

N10 S1=1000 M3

N20 S2=1500 M4

Ha egy orsót le kell állítania programból, írjunk:

N100 S1=0 M5

N110 S2=0 M5

CN_RNGREQ: S kódhoz tartozó tartomány száma (DWORD)

Ha az alkatrészprogramban S funkciót programoztak, a csatorna kezelő

- a CN_SC regiszterbe beírja az S funkción programozott fordulatszám értékét,
- a CN_SSEL regiszterbe beírja a hivatkozott orsó számát (1...16)
- a CN_RNGREQ regiszterbe beírja a programozott fordulatszámhoz tartozó tartomány kódját (11...18),
- majd a CN_SSTB jelzőt 1 PLC ciklus időtartamra 1-be írja.

A beírójel hatására a PLC a fenti regiszterekbe írt kódokat dekódolja és a CP_SACK nyugtázó jelet 0-ba állítja. A CP_SACK nyugtázó jelzőt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

A csatorna kezelő kikódolja, hogy a programozott fordulatszám az adott orsó melyik tartományába esik:

- Ha a tartomány határok átfedik egymást, mint általában esztergákon, a CN_RNGREQ regiszter nem használható (ekkor is történik kikódolás). Ekkor a tartományváltást M kódra (kötelezően M11, M12, ..., M18-ra) kell tenni a PLC programban, hogy a felhasználó eldönthesse, melyik tartomány optimális a forgcsolás szempontjából.
- Ha a tartomány határok nem fedik át egymást, mint általában marógépeken, használható a CN_RNGREQ regiszter.

Ebben az esetben a tartományváltás a PLC programban elintézhető külön M funkciók programozása nélkül is, nem kell M funkciókat bevezetni.

A csatorna kezelő a **CN_RNGREQ** regisztert a következőképp kódolja ki:

ha $S \leq N0650 R1 S Max$ paraméter értéke: akkor **CN_RNGREQ=11**,

ha $S \leq N0651 R2 S Max$ paraméter értéke: akkor **CN_RNGREQ=12**,

és így tovább.

CN_TC: T funkció kódja (DWORD)

Ha az alkatrészprogramban T funkciót programoztak, a csatorna kezelő

- a CN_TC regiszterbe beírja a T funkción programozott szerszám számát, eszterga csatornában a T címen hívott **korrekciószámot levágva**,
- a CN_MGZNO regiszterbe beírja, hogy a hivatkozott szerszám hányadik magazinban található,
- a CN_POTNO regiszterbe beírja, hogy a hivatkozott szerszám a megadott magazin hányadik zsebében van,
- megvizsgálja, hogy a hivatkozott szerszám, vagy szerszámcsoporthoz tartozó élettartama lejárt-e, és ennek megfelelően beállítja a CN_TLLE jelzőt,
- majd a CN_TSTB jelzőt 1 PLC ciklus időtartamra 1-be írja.

A beírójel hatására a PLC a fenti regiszterekbe írt kódokat dekódolja és a CP_TACK nyugtázó jelet 0-ba állítja. A CP_TACK nyugtázó jelzőt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

A CN_TC regiszterbe mindig az alkatrészprogramban hivatkozott szerszám száma (típuszáma) kerül, még a szerszámkezelő táblázat használata esetén is, amikor több, ugyanolyan típusszámú szerszám lehet a magazinban. Ekkor a CN_MGZNO és a CN_POTNO regiszterekből olvasható ki a beváltandó szerszám tartózkodási helye.

CN_AUX1C: 1. Segédfunkció kódja (DWORD)

CN_AUX2C: 2. Segédfunkció kódja (DWORD)

CN_AUX3C: 3. Segédfunkció kódja (DWORD)

Az N1332+n Aux Fu Addr_n (n=1..3) paraméteren ki lehet jelölni az A, B, C, U, V, W címek közül három különbözőt. Ha alkatrészprogramban az itt megadott három segédfunkció címére hivatkozunk, a csatorna kezelő

- a **CN_AUXnC** (n=1...3) regiszterbe beírja a segédfunkció címén programozott értéket (mindig egész DWORD),

- majd a CN_AUXnSTB (n=1...3) jelzőt 1 PLC ciklus időtartamra 1-be írja.

A beírójel hatására a PLC a fenti regiszterekbe írt kódokat dekódolja és a

CP_AUXnACK (n=1...3) nyugtázó jelet 0-ba állítja. A CP_AUXnACK nyugtázó jelzőt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

CN_CTSPN: G96 esetén az orsó fordulatszáma (DWORD)

Ha a csatorna G96 (konstans vágósebesség számítás) állapotban van,

- az NC 1-be írja a CN_CSURF jelzőt,

- PLC ciklusonként kiszámítja és a CN_CTSPN regiszterbe beírja a pillanatnyi koordinátaához tartozó orsó fordulatszámot,

- a G92 S utasítással programozott maximális fordulatszámot beírja a CN_NMAX regiszterbe.

A PLC program feladata, hogy a CN_CSURF=1 állapotban a **CN_CTSPN** regiszterben kapott értéket átmásolja a megfelelő (CP_SINP regiszter által meghatározott) orsó **SP_PRG** regiszterébe.

CN_NMAX: G96 esetén a max. fordulatszám értéke (G92 S_) (DWORD)

Ha a csatorna G96 (konstans vágósebesség számítás) állapotban van,

- az NC 1-be írja a CN_CSURF jelzőt,

- PLC ciklusonként kiszámítja és a CN_CTSPN regiszterbe beírja a pillanatnyi koordinátaához tartozó orsó fordulatszámot,

- a G92 S utasítással programozott maximális fordulatszámot beírja a **CN_NMAX** regiszterbe.

Az SP_PRG regiszterbe írt fordulatszámot a PLC programnak nem kell korlátoznia a CN_NMAX értékkel, azt az orsókezelő az alapjel kiadásakor figyelembe veszi.

CN_MGZNO: T kódhoz tartozó magazin száma (DWORD)

CN_POTNO: T kódhoz tartozó zsebszám (DWORD)

Ha az alkatrészprogramban T funkciót programoztak, a csatorna kezelő

- a CN_TC regiszterbe beírja a T funkció programozott szerszám számát, eszterga csatornában a T címen hívott *korrekciószámot levágva*,
- a CN_MGZNO regiszterbe beírja, hogy a hivatkozott szerszám hányadik magazinban található,
- a CN_POTNO regiszterbe beírja, hogy a hivatkozott szerszám a megadott magazin hányadik zsebében van,
- megvizsgálja, hogy a hivatkozott szerszám, vagy szerszámcsoport éltartama lejárt-e, és ennek megfelelően beállítja a CN_TLLE jelzőt,
- majd a CN_TSTB jelzőt 1 PLC ciklus időtartamra 1-be írja.

A beírójel hatására a PLC a fenti regiszterekbe írt kódokat dekódolja és a CP_TACK nyugtázó jelet 0-ba állítja. A CP_TACK nyugtázó jelzőt csak a funkció teljes végrehajtása után szabad 1-be kapcsolni.

Ha az N2900 Tool M. Config paraméter #0 TMU=1 bitje be van állítva a gépen a szerszámkezelő táblázat működik. Ekkor a CN_MGZNO és CN_POTNO regisztereket kezelni kell!

A szerszámkezelési funkciót az alábbi esetekben kell a vezérlésen élesíteni:

- a szerszámokra éltartam vizsgálatot akarunk alkalmazni,
- a magazinban tárolt szerszámokra az alkatrészprogramból nem helykód szerint, hanem szerszámkód alapján kívánunk hivatkozni,
- ha a gépen alkalmazott szerszámcsoport random tárkezelést igényel.

A szerszámkezelő táblázatba írhatjuk be a vezérlésben alkalmazott szerszámok T kódját, azaz a típusszámukat, amelyre az alkatrészprogramban hivatkozunk.

Ha több azonos műveletet ellátó szerszám vesz részt az éltartamkezelésben, az azonos műveletet végző szerszámokat ugyanazzal a típuszámmal írjuk be a táblázatba. Alkatrészprogramban T címen a típusszámra kell hivatkozni, és a szerszámkezelő dönti el, melyik típusszámú szerszámot veszi elő. Általában a legkisebb, még le nem járt éltartamú szerszámot. A szerszámkezelő táblázat globális, azaz minden csatornára közös.

A szerszámkezelő maximum 4 magazint kezel. Ezen kívül minden, a szerszámgépen definiált orsót, amibe szerszám kerülhet mint orsómagazint lehet definiálni az N2922 Spindle Magazines paraméteren. Az orsómagazinok mellé, az előbbi paraméteren, definiálhatunk egy-egy készenléti magazint a karban lévő szerszámok tárolására.

A CN_TC regiszterbe mindig az alkatrészprogramban hivatkozott szerszám száma (típuszáma) kerül, még a szerszámkezelő táblázat használata esetén is, amikor több, ugyanolyan típusszámú szerszám lehet a magazinban.

Szerszámkezelő táblázat használata esetén a hivatkozott típusszám mellé (CN_TC regiszter) a szerszámkezelő beírja a

CN_MGZNO regiszterbe, hogy a hivatkozott szerszám hányas számú magazinban található, illetve a

CN_POTNO regiszterbe, hogy a szerszám az adott magazin melyik zsebében van.

A **CN_MGZN** regiszter *értelmezése*:

CN_MGZN=0 jelentése: a hivatkozott szerszám nincs egyik magazinban sem,

CN_MGZN=1, 2, 3, 4 jelentése: a szerszám az 1., 2., 3., 4. magazinban van,

CN_MGZN=10, 20, 30, ... jelentése: a szerszám az 1., 2., 3., ... orsóban van,

CN_MGZN=11, 21, 31, ... jelentése: a szerszám az 1., 2., 3., ... orsóhoz tartozó készenléti magazinban (karban) van van.

A **CN_POTNO** regiszter *értelmezése*:

ha CN_MGZN=0, nincs jelentése,

ha CN_MGZN=1, 2, 3, 4, akkor CN_POTNO=1, 2, 3, ... a 30 Tool Management paramétercsoportban az adott magazin hosszára beállított érték,

ha CN_MGZN=10, 20, 30, ..., akkor CN_POTNO=1, (1., 2., 3., ... orsóban van)

ha CN_MGZN=11, 21, 31, ..., akkor CN_POTNO=1. (1, 2, 3, ... orsóhoz tartozó karban van).

A PLC-től az NC felé menő duplaszavas csatornaváltozók**CP_JOGFD:** Jog előtolás kiválasztása táblázatból (DWORD)

Ha az N0316 Jog F Contr paraméteren a JFT opció van választva (a paraméter értéke 1), a jog üzemmódban az előtolás értékét a csatornakezelő a CP_JOGFD PLC regiszter értéke alapján állítja be, exponenciális függvény szerint. Az előtolás mindig 1/min dimenzióban kerül kiadásra.

A CP_JOGFD regiszter értékét az override kapcsoló állásához kell kötni. NCT gépi kezelőpanel esetén az MKFOVER regiszter értékéhez.

Az alábbi táblázat CP_JOGFD regiszter értékének függvényében 0-tól 15-ig mutatja az előtolás értékét, ha JFT paraméter van beállítva. Nagyobb CP_JOGFD értékekre a sor folytatódik.

CP_JOGFD	G21 mm/min	G20 in/min	körtengely °/min
0	0	0	0
1	2	0.08	0.4
2	3.2	0.12	0.64
3	5	0.2	1
4	7.9	0.3	1.58
5	12.6	0.5	2.52
6	20	0.8	4
7	32	1.2	6.4
8	50	2	10
9	79	3	15.8
10	126	5	25.2
11	200	8	40
12	320	12	64
13	500	20	100
14	790	30	158
15	1260	50	252

CP_SINP: Megmunkálást végző orsó száma (1,2...) (DWORD)

A PLC programnak a CP_SINP regiszterben csatornánként ki kell jelölnie az aktív orsó számát: **CP_SINP**=1, 2, 3, ...

Az aktív orsó jeladójáról veszi a *csatornakezelő* a jeladó impulzusokat

- G95 fordulatonkénti előtolás,
- G33 menetvágás során.
- G84.2, G84.3 merevszárú menetfűrés esetén az aktív orsóról várja az SN_LPCLSD hurok zárva jelet, és az aktív orsóval fűr,
- G51.2 sokszögesztergálás esetén a mester orsó mindig az aktív orsó, illetve

A **PLC program** mindig a CP_SINP regiszterben megadott aktív orsó

- forgásállapota alapján engedélyezi forgácsoló mondatban az előtolást a CP_CBFEN jelzővel,
- G96 konstans vágósebesség számítás esetén CN_CSURF=1 állapotban a CN_CTSPN regiszterben kapott értéket átmásolja a megfelelő **CP_SINP** regiszter által meghatározott aktív orsó **SP_PRG** regiszterébe.

Több csatornához is lehet ugyanazt az orsót rendelni. Például egy 2x2 tengelyes esztergán mindkét csatornához a bal oldali orsót rendelve, ugyanazon a darabon dolgozhat mindkét szánrendszer fordulatonkénti előtolással. Az alkatrész program írója eldöntheti, hogy a konstans vágósebesség számítást melyik csatorna X tengelye alapján vegye a vezérlő figyelembe (a szükséges csatornában programozza a G96 kódot).

Azt, hogy az aktív orsó melyik legyen és hogyan kell kiválasztani, a gép építője dönti el. Az aktív orsó kijelölését M kódokkal lehet megadni. Például:

M31 kód S1 aktív,
M32 kód S2 aktív stb.

☞ **Figyelem!** Az aktív orsót kijelölő M funkciót összevárós (pufferürítő) funkcióra kell kijelölni, mert a kód végrehajtásának hatása van a mondatelőkészítésre.

CP_CSAX: Nem használt

CP_POLYSL: Sokszögesztergálás szolga orsójának száma (1,2,...) (DWORD)

A CP_POLYSL regiszterbe kell írni csatornánként annak az orsónak a számát, amelyik a sokszögesztergálás szolga orsója lesz, azaz amelyikben a szerszám forog.

A sokszögesztergálás mesterorsóját a CP_SINP regiszter jelöli ki.

A G51.2 sokszögesztergálás parancs kiadása előtt M funkcióval a PLC programnak

- ki kell választania a szolga orsót a CP_POLYSL regiszter megadásával
- az orsókezelőt SP_SEN=1 engedélyezni kell
- az orsóhajtást be kell kapcsolni.

CP_ACTT: Az aktuális szerszám száma (DWORD)

Ha az N2901 Search Config paraméter #7 TSP bitje 0, az aktuális szerszámszám kijelzése az FST ablakban a CP_ACTT csatornánként indexelt regiszterből történik.

A PLC program a beváltott szerszám számát írja ide.

CP_MGR1: A PLC-ből kijeleszni kívánt 1. M-kód (DWORD)

CP_MGR2: A PLC-ből kijeleszni kívánt 2. M-kód (DWORD)

CP_MGR3: A PLC-ből kijeleszni kívánt 3. M-kód (DWORD)

CP_MGR4: A PLC-ből kijeleszni kívánt 4. M-kód (DWORD)

CP_MGR5: A PLC-ből kijeleszni kívánt 5. M-kód (DWORD)

CP_MGR6: A PLC-ből kijeleszni kívánt 6. M-kód (DWORD)

CP_MGR7: A PLC-ből kijeleszni kívánt 7. M-kód (DWORD)

CP_MGR8: A PLC-ből kijelezni kívánt 8. M-kód (DWORD)

CP_MGR9: A PLC-ből kijelezni kívánt 9. M-kód (DWORD)

CP_MGR10: A PLC-ből kijelezni kívánt 10. M-kód (DWORD)

CP_MGR11: A PLC-ből kijelezni kívánt 11. M-kód (DWORD)

CP_MGR12: A PLC-ből kijelezni kívánt 12. M-kód (DWORD)

CP_MGR13: A PLC-ből kijelezni kívánt 13. M-kód (DWORD)

CP_MGR14: A PLC-ből kijelezni kívánt 14. M-kód (DWORD)

CP_MGR15: A PLC-ből kijelezni kívánt 15. M-kód (DWORD)

CP_MGR16: A PLC-ből kijelezni kívánt 16. M-kód (DWORD)

Az N1341 M GR Low 1, ..., N1356 M GR Low 16 és az N1357 M GR High 1, ..., N1372 M GR High 16 paramétereken 16 különböző M kód csoport jelölhető ki. A Low jelzésű paraméterre a csoport legkisebb számú, a High jelzésű paraméterre a csoport legnagyobb számú kódját kell írni.

A paramétereket úgy kell megadni, hogy a kijelölt M kód sávok egymást kizáró gépállapot jelző kódok legyenek.

A mondatbeolvasó az M kódokat úgy szűri, hogy az egy csoportba tartozó M kódok közül csak egy lehet az adott mondatban, egyébként *Ellentmondó M-kódok* hibaüzenetet ad.

A paraméteren beállított értékeket mondatkeresés során is figyelembe veszi a vezérlő, az M kódok kigyűjtésénél. Az egy csoportba tartozó M kódok közül csak az utolsónak megadott kódot gyűjti ki. Tekintsük a következő M kódokat:

M51: tokmány zárás S1 orsón

M52: tokmány nyitás S1 orsón

M53: tokmány nyitás forgó orsónál S1 orsón

Állítsuk be a paramétert a következőképp:

N1341 M GR Low 1=51

N1357 M GR High 1=53

Szorítás állapot:

M54: belső szorítás S1

M55: külső szorítás S1

A paraméterek:

N1342 M GR Low 2=54

N1358 M GR High 2=55

Mondatkeresés során M51, M52, M53 kódok közül csak az utolsónak programozottat gyűjti ki és hajtja végre a kezelővel. Ugyanez vonatkozik az M54, M55 csoportra is. A megfelelő gépállapotok közül mindig csak egy kerül kiírásra a CP_MGR1, 2 regiszterekben, pl. a tokmány zárása után a regiszterek tartalma:

CP_MGR1=51 – tokmány zárva S1 orsón és

CP_MGR2=54 – belső szorítás S1 orsón.

Az M GR Low n, M GR High n pár megfelel a CP_MGRn regiszternek. A paraméterekre azoknak az M kód csoportoknak a kezdő- és végszámát kell beírni, amelyek a PLC program a

- **CP_MGR1, ..., CP_MGR16** regiszterekbe beír, és amelyek az M kódok ablakban kijelzésre kerülnek.

CP_1032: #1032 makrováltozó (DWORD)

A felhasználó alkatrészprogramból makrováltozókon keresztül lekérdezhet bites PLC jelzőket. Csatornánként 32 jelzőbit áll a felhasználó rendelkezésére a PLC-vel történő kommunikáció számára: #1000, #1001, ..., #1031. Ezeket a változókat le lehet kérdezni duplaszavasán is a #1032 makrováltozón keresztül.

Például az alkatrészprogramba írt

```
IF #1032EQ16 GOTO30
```

utasítás akkor ugrik az N30 mondatra, amikor a #1004=1 (CP_1004 jelző 1), a többi 0.

CP_OFFSNO: A szerszámbemérésnél használandó korrekció száma (DWORD)

Tapintóval történő hosszkorrekció/munkadarab nullpont kézi bemérése során a hosszkorrekciós regisztert, amibe a tapintó mérési eredménye kerül/ahonnan az eltolás kiszámításához a korrekciót veszi, ha az N3016 Tool/WP Setter Config paraméter #0 ONS bitjének értéke:

=0: a kezelő kézzel választja ki a kezelőpanelen,

=1: a PLC választja ki a CP_OFFSNO regiszterben.

Ha például a kezelő a szerszámokat úgy akarja bemérni, hogy a szerszámszámmal egyezzen meg a korrekció száma (pl. T1212) a CP_OFFSNO regiszterbe a PLC program írja be az aktuális szerszám számát. Így szerszámváltás után a bemérés ablak az aktuális szerszám számával megegyező korrekcióra ugrik, nem kell kézzel kiválasztani a bemérendő korrekciót.

7.15.3 A lebegőpontos csatornaváltozók

Bemenetek		Kimenetek	
Szimbólum	Leírás	Szimbólum	Leírás
CN_1133	#1133 makrováltozó értéke (double)	CP_INC	Incrementális jog és kézikerék lépés nagysága (double)
CN_1134	#1134 makrováltozó értéke (double)	CP_FOVER	Előtolás override: ha =1: 100% (double)
CN_1135	#1135 makrováltozó értéke (double)	CP_ROVER	Gyorsmenet override: ha =1: 100% (double)
CN_GC	Nem használt	CP_COV	Nem használt
CN_G1DAT	Nem használt	CP_1033	#1033 makrováltozó (double)
CN_G2DAT	Nem használt	CP_1034	#1034 makrováltozó (double)
CN_G3DAT	Nem használt	CP_1035	#1035 makrováltozó (double)

Az NC-től a PLC felé menő lebegőpontos csatornaváltozók**CN_1133:** #1133 makrováltozó értéke (double)**CN_1134:** #1134 makrováltozó értéke (double)**CN_1135:** #1135 makrováltozó értéke (double)

A felhasználó alkatrészprogramból a #1133, #1134, #1135 makrováltozóknak történő értékadással lebegőpontos adatokat adhat át a PLC számára csatornánként. Ezeket az adatokat a PLC program a CN_1133, CN_1134, CN_1135 regiszterekből közvetlenül ki tudja olvasni.

Az alkatrészprogramba írt

#1134=167.832

utasítás a CN_1134-es PLC regiszterbe írja a 167.832 számot.

CN_GC: Nem használt**CN_G1DAT:** Nem használt**CN_G2DAT:** Nem használt**CN_G3DAT:** Nem használt

A PLC-től az NC felé menő lebegőpontos csatornaváltozók**CP_INC:** Incrementális jog és kézikerek lépés nagysága (double)

Inkrementális jog (CN_INCR=1), kézikerek (CN_HNDL=1) üzemmódokban, vagy kézikerekkel vezérelhető előtolás (CP_FHNDL=1) esetén a CP_INC regiszterbe kell írni csatornánként a lépés nagyságát lebegőpontosan.

A lépésnagyság kiválasztása NCT gépi kezelőpanel használata esetén az

MB_I1, MB_I10, MB_I100, MB_I1000

gombokról történik.

A lépésnagyság kiválasztása NCT kihelyezett kézikerek esetén a

HB_I1, HB_I10, HB_I100, HB_I1000

jelzőkről történik.

A CP_INC regiszterbe mindig lebegőpontos számot kell írni. A beírandó szám mértékegységét az N0104 Unit of Measure paraméter #0 IND bitje határozza meg, amely a kimeneti mértékrendszert mutatja. (Azt a mértékrendszert, amiben a pozíciómérés történik.)

Ha CP_INC=0.01 és

IND=0: CP_INC=0.01 jelentése 0.01 mm

IND=1: CP_INC=0.01 jelentése 0.01 inch

A vezérlőben alkalmazott bemeneti mértékrendszer függvényében az adatokat konvertálni kell. A konverzióhoz a CN_INCH jelző használható. Ha a jelző

CN_INCH= 0: G21 metrikus adatbevitel

CN_INCH=1: G20 inch-es adatbevitel

van érvényben.

Például MB_I10 gombot (0.01) nyomtak:

– **IND=0** (metrikus mérés) **CN_INCH=0** (G21 metrikus adatbevitel) esetén

CP_INC=0.01 (lépésnagyság **0.01 mm**) írandó

– **IND=0** (metrikus mérés) **CN_INCH=1** (G20 inch-es adatbevitel) esetén

CP_INC=0.0254 (lépésnagyság 0.0254 mm = 0.0254/25.4 = **0.01 inch**)

írandó.

CP_FOVER: Előtolás override: ha =1: 100% (double)

Az előtolás override értéke csatornánként. Lebegőpontos szám. Ha például

CP_FOVER=1.0 jelentése 100%

CP_FOVER=1.427 jelentése 142.7%

NCT gépi kezelőpanel használata esetén az override kapcsoló állását az MKFOVER regiszterből kell venni.

Az előtolás override felső korlátját a PLC programban kell beállítani!

☞ **Figyelem!** MKFOVER egész DWORD típusú, CP_FOVER pedig lebegőpontos double, tehát az override beállítása fixpontosból lebegőpontosba való konverziót (FLT utasítás) igényel. CP_FOVER indexelése 2-esével történik!

CP_ROVER: Gyorsmenet override: ha =1: 100% (double)

A gyorsmenet override értéke csatornánként. Lebegőpontos szám. Ha például

CP_ROVER=0.272 jelentése 27.2%

CP_ROVER=1.0 jelentése 100%

NCT gépi kezelőpanel használata esetén a gyorsmenet override értékét az előtolás override kapcsoló állásából az MKFOVER regiszterből is lehet venni.

A gyorsmenet override felső korlátja 100%, ennél tovább a csatornakezelő nem engedi!

☞ **Figyelem!** MKFOVER egész DWORD típusú, CP_ROVER pedig lebegőpontos double, tehát az override beállítása fixpontosból lebegőpontosba való konverziót (FLT utasítás) igényel. CP_ROVER indexelése 2-esével történik!

CP_COV: Nem használt

CP_1033: #1033 makrováltozó (double)

CP_1034: #1034 makrováltozó (double)

CP_1035: #1035 makrováltozó (double)

A PLC program a CP_1033, CP_1034, CP_1035 regiszterekbe történő írással lebegőpontos adatokat adhat át közvetlenül az alkatrészprogram számára. Az alkatrészprogram a #1033, #1034, #1035 makrováltozókon keresztül az adatokat felhasználhatja.

Az alkatrészprogramba írt

#100=#1134

utasítás a CP_1134-es PLC regiszterbe írt lebegőpontos számot írja a #100 makrováltozóba.

Betűrendes index:

. (pont) operátor.....	15	AN_REFEND.....	272
, (vessző) operátor.....	18	AN_REFP1.....	272
: operátor.....	17	AN_REFP2.....	272
! operátor.	24	AN_REFP3.....	272
+F.....	75	AN_REFP4.....	272
-F.....	76	AN_RPE.....	271
* operátor.	25	AN_SPRPNA.....	273
*F.....	77	AN_SPRPNM.	274
/F.....	78	AN_SYNCA.....	273
# operátor.	24	AN_SYNCM.	274
#\$ operátor.	25	AND.....	66
@.	23	ANINPUTS.	239, 240
ACOS.....	85	AP_DECSW.....	277
ActPosAx.....	117	AP_DETCHR.....	275
ActPosSp.	119	AP_DIARAD.....	282
ADD.....	70	AP_DISPD.....	279
ALR.	101	AP_EFD.....	284
ALRF.	101	AP_END.....	277
alulcsordulás jelző.	28	AP_FEEDD.	283
AN_AXALM.	271	AP_FLWU.	277
AN_BEPTY.....	274	AP_GOR.....	283
AN_DETCHA.....	270	AP_JOGN.....	277
AN_EGBS.	273	AP_JOGP.....	277
AN_GOA.....	275	AP_LCK.....	279
AN_IEPTY.....	275	AP_LIMN.....	278
AN_INDP.....	274	AP_LIMP.....	278
AN_INPOS.....	270	AP_LIMSELN.....	279
AN_LUBR.....	271	AP_LIMSELP.....	279
AN_MIRA.....	275	AP_MIRR.....	284
AN_MIXA.....	273	AP_MIXR.....	280
AN_MIXM.....	274	AP_MTNDN.....	278
AN_MTNRN.....	271	AP_MTNDP.....	278
AN_MTNRP.....	271	AP_OPNR.....	276
AN_OPNA.....	270	AP_PARKR.....	279
AN_OTN.....	272	AP_PLCR.....	283
AN_OTP.....	272	AP_RAPD.....	278
AN_PARKA.....	273	AP_RES.....	283
AN_PLCA.....	274	AP_RPE.....	284
AN_RAPR.....	271	AP_SPRPNR.....	281

AP_SSLOP.	282	CN_1117.	318
AP_SYNCR.	280	CN_1118.	318
áramúterv.	8	CN_1119.	318
ARTL.	62	CN_1120.	318
ARTR.	62	CN_1121.	318
ASHL.	60	CN_1122.	318
ASHR.	60	CN_1123.	318
ASIN.	84	CN_1124.	318
ATAN.	86	CN_1125.	318
átvitel jelző.	28	CN_1126.	318
AxesTime.	121	CN_1127.	318
báziscím.	13	CN_1128.	318
BCD.	25, 91	CN_1129.	318
BIN.	90	CN_1130.	318
binárisan kódolt decimális.	25	CN_1131.	318
byte.	12	CN_1132.	333
CANErr.	121	CN_1133.	344
CEQ.	97	CN_1134.	344
CGE.	97	CN_1135.	344
CGT.	97	CN_ABSOFF.	315
ChannelsTime.	120	CN_AHND.	317
CLE.	97	CN_ALRM.	307
CLT.	97	CN_AUTO.	312
CMP.	96	CN_AUX1C.	337
CN_1100.	317	CN_AUX1STB.	306
CN_1101.	317	CN_AUX2C.	337
CN_1102.	317	CN_AUX2STB.	306
CN_1103.	317	CN_AUX3C.	337
CN_1104.	317	CN_AUX3STB.	306
CN_1105.	317	CN_BKBUF.	306
CN_1106.	317	CN_BKRET.	316
CN_1107.	317	CN_BKRST.	316
CN_1108.	317	CN_CBFR.	309
CN_1109.	317	CN_CHARP.	311
CN_1110.	317	CN_CHOP.	311
CN_1111.	317	CN_CSACK.	308
CN_1112.	317	CN_CSURFS.	307
CN_1113.	317	CN_CTSPN.	337
CN_1114.	317	CN_DRRUN.	316
CN_1115.	318	CN_DWELL.	310
CN_1116.	318	CN_EDIT.	312

CN_EGBMD.	308	CN_POSCHK.	310
CN_FLCK.	312	CN_POTNO.	338
CN_FREV.	310	CN_REFP.	312
CN_G1DAT.	344	CN_REFPG.	310
CN_G2DAT.	344	CN_RNGREQ.	336
CN_G3DAT.	344	CN_RTAP.	310
CN_GC.	344	CN_RTRFIN.	308
CN_GSTB.	306	CN_SC.	334
CN_HNDL.	312	CN_SKIP.	310
CN_HSHP.	308	CN_SSEL.	334
CN_INCH.	308	CN_SSTB.	305
CN_INCR.	312	CN_START.	311
CN_INTD.	307	CN_STOP.	311
CN_IPEPTY.	309	CN_STPREQ.	306
CN_IPSTP.	309	CN_TAP.	310
CN_ITFALM.	307	CN_TAXF.	313
CN_ITFCHK.	307	CN_TBLB.	314
CN_JOG.	312	CN_TC.	336
CN_M1C.	334	CN_TEST.	315
CN_M1STB.	305	CN_THRD.	309
CN_M2C.	334	CN_THRDC.	310
CN_M2STB.	305	CN_TLLE.	311
CN_M3C.	334	CN_TRGAF.	313
CN_M3STB.	305	CN_TSTB.	306
CN_M4C.	334	CN_TTCRF.	314
CN_M4STB.	305	CN_WMCAXF.	317
CN_M5C.	334	CN_WPCNT.	308
CN_M5STB.	305	CN_WTNG.	307
CN_M6C.	334	CNCBufferCount.	121
CN_M6STB.	305	CNE.	97
CN_M7C.	334	CNT.	52, 54
CN_M7STB.	305	CommandAx.	117
CN_M8C.	334	CommandSp.	119
CN_M8STB.	305	CompenValAx.	119
CN_MDI.	312	ComPosAx.	116
CN_MGZNO.	338	ComPosSp.	119
CN_MLCK.	315	ComVelAx.	117
CN_NMAX.	337	ComVelSp.	119
CN_OPMES.	307	COS.	82
CN_OVDIS.	309	CP_1000.	330
CN_POLYT.	308	CP_1001.	330

CP_1002.....	330	CP_BKRET.....	323
CP_1003.....	330	CP_BKRST.....	323
CP_1004.....	330	CP_CBFEN.....	328
CP_1005.....	330	CP_CHOPON.....	329
CP_1006.....	331	CP_CNDBK_1.....	324
CP_1007.....	331	CP_CNDBK_2.....	324
CP_1008.....	331	CP_CNDBK_3.....	324
CP_1009.....	331	CP_CNDBK_4.....	324
CP_1010.....	331	CP_CNDBK_5.....	324
CP_1011.....	331	CP_CNDBK_6.....	324
CP_1012.....	331	CP_CNDBK_7.....	324
CP_1013.....	331	CP_CNDBK_8.....	324
CP_1014.....	331	CP_CNDSP.....	322
CP_1015.....	331	CP_COV.....	346
CP_1016.....	331	CP_CSAX.....	341
CP_1017.....	331	CP_CSREQ.....	325
CP_1018.....	331	CP_DRRUN.....	323
CP_1019.....	331	CP_EDIT.....	319
CP_1020.....	331	CP_EGBRRQ.....	325
CP_1021.....	331	CP_FHNDL.....	328
CP_1022.....	331	CP_FIN.....	324
CP_1023.....	331	CP_FLCK.....	324
CP_1024.....	331	CP_FOVER.....	345
CP_1025.....	331	CP_GACK.....	327
CP_1026.....	331	CP_HNDL.....	319
CP_1027.....	331	CP_HNDLS1.....	328
CP_1028.....	331	CP_HNDLS2.....	328
CP_1029.....	331	CP_HNDLS3.....	328
CP_1030.....	331	CP_HNDLS4.....	328
CP_1031.....	331	CP_HOLD.....	327
CP_1032.....	343	CP_INC.....	345
CP_1033.....	346	CP_INCR.....	319
CP_1034.....	346	CP_INTDREQ.....	320
CP_1035.....	346	CP_JOG.....	319
CP_ABSOFF.....	324	CP_JOGFD.....	340
CP_ACTT.....	341	CP_JOGRAP.....	320
CP_AHND.....	322	CP_LIM1DIS.....	328
CP_AUTO.....	319	CP_LIM2DIS.....	328
CP_AUX1ACK.....	327	CP_LIM3DIS.....	328
CP_AUX2ACK.....	327	CP_LIMSEL.....	329
CP_AUX3ACK.....	327	CP_M1ACK.....	326

CP_M2ACK.....	326	CP_SACK.	326
CP_M3ACK.....	326	CP_SGLBK.	322
CP_M4ACK.....	326	CP_SGOEN.	329
CP_M5ACK.....	326	CP_SGOX.	330
CP_M6ACK.....	326	CP_SGOY.	330
CP_M7ACK.....	326	CP_SGOZ.....	330
CP_M8ACK.....	326	CP_SINP.....	340
CP_MDI.	319	CP_START.	319
CP_MGR1.	341	CP_STOP.....	319
CP_MGR10.	342	CP_TACK.	327
CP_MGR11.	342	CP_TAXF.....	320
CP_MGR12.	342	CP_TBLB.....	320
CP_MGR13.	342	CP_TEST.....	323
CP_MGR14.	342	CP_TLCM.	321
CP_MGR15.	342	CP_TMREN.....	325
CP_MGR16.	342	CP_TRGAF.	320
CP_MGR2.	341	CP_TSBD.....	325
CP_MGR3.	341	CP_TTCRF.....	320
CP_MGR4.	341	CP_WMCAXF.....	322
CP_MGR5.	341	CP_WPCM.....	321
CP_MGR6.	341	DA-i/I EE.	241
CP_MGR7.	341	DANI.	240
CP_MGR8.	342	DEG.	94
CP_MGR9.	342	DIFD.....	38
CP_MINT.....	325	DIFU.....	37
CP_MLCK.	323	DirectPlcBit.	116
CP_NOWT.....	325	DirectPlcDouble.....	116
CP_OFFSNO.	343	DirectPlcInt.....	116
CP_OSGNX.....	330	DIV.....	73
CP_OSGNY.....	330	DN_BVERR.....	247
CP_OSGNZ.	330	DN_CHERR1.....	248
CP_OVC.....	328	DN_CMEERR.	247
CP_POLYSL.	341	DN_CURERR.	247
CP_REFP.	319	DN_CWDER1.	248
CP_RLSOT3.....	330	DN_ECTERR.....	248, 254
CP_ROVER.	346	DN_EDERR1.....	247, 254
CP_ROVLD.....	330	DN_EDERR2.....	247
CP_RST.	325	DN_ENA.	245
CP_RSTREW.....	325	DN_ERR.....	249, 254
CP_S2TS.	321	DN_ERROR.....	247, 254
CP_S2WS.....	322	DN_FOLERR.....	248

DN_HALERR.	247	FL_CY.	28
DN_HASERR.	248	FL_ER.	28
DN_INC.	245 , 254	FL_GT.	29
DN_OVHERR.	248	FL_LT.	29
DN_PDPINT.	248	FL_OF.	28
DN_PRGERR.	248	FL_UF.	28
DN_PRM1.	245	FLEQ.	97
DN_PRM2.	245	FLGE.	97
DN_PRMERR.	248	FLGT.	97
DN_RDY.	245	FLLE.	97
DN_SRTERR.	248	FLLT.	97
DN_STAT.	249 , 254	FLNE.	97
DP_CTRL.	249 , 254	FLT.	92
DP_EMG.	246	FolErrAx.	117
DP_ENA1.	245	FolErrSp.	119
DP_ENA2.	245	Főprogram.	21
DP_ERRCLR.	247 , 254	FUP.	276 , 277
DP_MOD1.	246	HardwareTime.	120
DP_MOD2.	246	HB_AXIS4.	231
DP_POSLCK.	246	HB_AXIS5.	231
DP_PRM1.	246	HB_AXIS6.	231
DP_PRM2.	246	HB_AXIS7.	231
DP_SILCK.	247	HB_AXIS8.	231
DS-i/I EE.	241	HB_AXISX.	231
Dupla szó.	11	HB_AXISY.	231
dupla szó lekérdezése.	31	HB_AXISZ.	231
dupla szó negált lekérdezése.	33	HB_B12.	232
DWORD.	11	HB_B13.	232
EGBcCurrAx.	118	HB_I1.	231
EGBvTarAx.	118	HB_I10.	231
egyenlő jelző.	29	HB_I100.	231
Előjelbit.	24 , 26	HB_I1000.	231
eltolás.	13	hiba jelző.	28
END.	105	HWMOVE.	232
ENDAT.	250	I16.	234
ETPC.	236	I16S.	234
EXP.	87	I32.	234
exponens.	26	IEEE754.	26
FCMP.	96	Denormalizált szám.	26
felfutó él.	23	NaN (Not a Number).	27
FIX.	93	Nem Szám.	27

Normalizált szám.....	26	MB_MLCK.	228
Nulla.	27	MB_REFP.	228
Impulzusképzés felfutó élre.	37	MB_S100.	230 , 299
Impulzusképzés lefutó élre.....	38	MB_SGLBK.....	228
IN_1.	239	MB_SMAX.	230 , 299
IN_1ENn.....	239	MB_SMIN.	230
indexelt címzés.....	13	MB_START.....	228
Int0.	21	MB_STOP.	228
JME.	106	MB_TEST.	228
JMP.....	106	Measured.	120
kisebb, mint jelző.....	29	Mechanical Type.	313 , 314
Kitevő.....	26	Memória bitjének beállítása.....	35
komment.....	42 , 43	Memória bitjének törlése:.....	36
lebegőpontos szám.....	25	memóriabit írása.....	34
lefutó él.....	23	memóriabit lekérdezése.	30
% operátor.....	23	memóriabit negált írása.	35
létradiagram.	8 , 41	memóriabit negált lekérdezése.....	32
létrafok.	41	mintavételi idő.	9
LOG.	88	MK15.	224
M GR High n.	342	MK19.	224
MACR.	113	MKFOVER.....	229
MACW.....	114	MKSOVER.....	230
Mantissza.	26	ML_AUTO.....	228
MB_AUTO.....	228	ML_BKRET.....	228
MB_BKRET.....	228	ML_BKRST.....	228
MB_BKRST.....	228	ML_CNDBK.	229
MB_CNDBK.	229	ML_CNDSP.....	229
MB_CNDSP.....	229	ML_DRRUN.	228
MB_DRRUN.	228	ML_EDIT.....	228
MB_EDIT.....	228	ML_FLCK.	228
MB_FLCK.	228	ML_HNDL.....	228
MB_HNDL.....	228	ML_INCR.....	228
MB_I1.....	229	ML_JOG.....	228
MB_I10.....	229	ML_JOGn.....	229
MB_I100.....	229	ML_JOGRAP.....	228
MB_I1000.....	229	ML_MDI.	228
MB_INCR.	228	ML_MLCK.....	228
MB_JOG.....	228	ML_REFP.....	228
MB_JOGn.....	229	ML_SGLBK.....	228
MB_JOGRAP.....	228	ML_START.....	228
MB_MDI.	228	ML_STOP.	228

ML_TEST.	228	MSGF.	101
modul.	21	MUL.	72
MOV.	45	MVN.	45
MOVCMD.	109 , 274 , 283	MW.	124
MOVF.	46	N_ACTMSG.	268
MR.	123	N_CLRMSG.	265
MR, MW funkció kódok		N_FIRSTCC.	264
10.	127	N_MONDIS.	264
100.	162	N_MONST.	264
101.	163	N_MSG.	256
102.	165	N_MSG0.	265
103.	169	N_MSG1.	265
104.	173	N_MSG2.	265
105.	176	N_MSG3.	265
106.	177	N_MSG4.	265
107.	180	N_MSG5.	265
108.	182	N_MSG6.	265
11.	128	N_MSG7.	265
201.	206	N_MSG8.	265
202.	212	N_MSG9.	265
300.	189	N_MSGA.	265
303.	191	N_NCREADY.	264
304.	194	N_NVECAT.	266
306.	196	N_NVRAMOK.	264
307.	198	N_OFF.	264
308.	200	N_ON.	264
309.	202	N_P100MS.	264
31.	130	N_P1M.	264
32.	132	N_P1S.	264
34.	134	N_P2MS.	264
35.	136	N_P2T.	264
40.	138	N_PDB1.	261
41.	203	N_PDW1.	261
60.	139	N_Pij.	261
61.	140	N_PTEDT.	266
62.	141	N_SIMU.	266
63.	142	N_SW.	257
70.	144	N_SW0.	258
81.	147	N_SWN.	257
90.	149	N_SWP.	258
MSG.	101	N_TLEDT.	265

N_TLMD.	265	P_SVRC.	268
N_TLSRCH.	265	P_TLTAB.	267
N_TLSV.	265	P_TRCTR.	267
NActSp.	120	P_WOFFS.	267
nagyobb, mint jelző.	29	PitchAx.	118
NCommandSp.	120	PLC Doublen.	261
NCTDriveMess.	121	PlcBit.	116
NEG.	65	PlcCycle.	121
Negált relétekercs.	35	PlcDouble.	116
NSetSp.	120	PlcInt.	116
nyitóérintkező.	32, 33	PlcTime.	121
O16.	234	PosErrAx.	117
O8R.	234	PosErrSp.	119
O8RM.	234	PWR.	79
operátor		RAD.	95
. (pont).	15	RealTime.	120
, (vessző).	13	Relétekercs.	34
!.	24	REM.	101
*.	25	Remark.	42
%.	23	REMF.	101
#.	24	RET.	107
#\$.	25	ROT.	56
@.	23	RotaryAx.	121
OR.	66	RST.	36
oszlop.	41	SBN.	107
P_CHSEL.	268	SBS.	107
P_COMPG.	267	SCP.	115
P_COMPW.	267	SEC.	43
P_DIR.	267	SENS.	238
P_HnAS.	232	SET.	35
P_HOLD0.	266	SHTR.	59
P_MAC.	267	SIN.	81
P_MONREQ.	266	SN_FLOFF.	287
P_MSG.	256	SN_FLU.	287
P_PAR.	268	SN_LPCLSD.	287
P_PLC.	268	SN_N0.	287
P_PRGE.	267	SN_NACT.	297
P_PTTAB.	268	SN_NCOM.	297
P_RUNAUT.	267	SN_NS.	286
P_RUNMDI.	267	SN_ORIP.	287
P_SHTDNREQ.	266	SN_PHSHFTA.	288

SN_POLYA.....	288	SP_SSTRT.....	290
SN_RMPD.....	286	SP_SSYNCR.....	292
SN_RPE.....	289	SP_TLCD.....	296
SN_SALM.....	289	SP_TLCHA.....	295
SN_SDETHA.....	288	SP_TLCHIA.....	295
SN_SINDP.....	289	SP_TLSKP.....	296
SN_SINPOS.....	287	SQRT.....	80
SN_SMTNRN.....	288	státuszbitek.....	28
SN_SMTNRP.....	288	FL_CY.....	28
SN_SRAPR.....	288	FL_EQ.....	29
SN_SSYNA.....	288	FL_ER.....	28
SN_SYNCPOS.....	288	FL_GT.....	29
SN_TLCH.....	289	FL_LT.....	29
SN_TLCHI.....	289	FL_OF.....	28
SN_TLNL.....	290	FL_UF.....	28
SN_TLSKPA.....	290	StraightnessAx.....	118
sor.....	41	SUB.....	71
SP_ACTT.....	299	SyncErrAx.....	118
SP_ASSIGN.....	298	SyncErrSp.....	120
SP_FEEDD.....	295	SyncVSlaveSp.....	120
SP_MAST.....	298	SyncVTargetSp.....	120
SP_NEG.....	290	szimbolikus hozzáférés.....	12
SP_OREQ.....	291	szó.....	12
SP_OSHRT.....	291	TachAx.....	117
SP_OSW.....	296	TachRealAx.....	118
SP_PAR.....	290	TachRealSp.....	121
SP_PHSHFTR.....	292	TachSp.....	119
SP_POLYR.....	293	TAN.....	83
SP_PRG.....	298	TN_INPn1.....	237
SP_RAPD.....	295	TN_INPn2.....	237
SP_RNG.....	298	TN_INPn3.....	237
SP_ROT.....	298	TN_TSn.....	237
SP_SDETHR.....	293	TOFFD.....	49
SP_SDISPD.....	295	TOND.....	48
SP_SEN.....	290	Tool Axis Direction.....	313 , 314
SP_SEND.....	293	TP_OUTn1.....	237
SP_SLCLR.....	294	TP_OUTn2.....	237
SP_SMTNDN.....	293	TPLC.....	9
SP_SMTNDP.....	293	TPULSE.....	50
SP_SOVER.....	299	TSliceErr.....	121
SP_SSROFF.....	294	TTLAI.....	250

ECAT-TACHO.....	250
ECAT-TTLASM.....	250
TTLCAN.	250
túlcsordulás jelző.	28
Velocity.	122
WORD.	12
XOR.	66
záróelem.	41
záróérintkező.	30 , 31