

# Uživatelský manuál k jednotce MARS 8

Pavel Píša ( pisa@cmp.felk.cvut.cz )

8. října 2018

## Obsah

<b>1</b>	<b>Specifikace systému pro víceosou regulaci</b>	<b>1</b>
<b>2</b>	<b>Ovládání z PC po lince RS232</b>	<b>3</b>
2.1	Tvar příkazů a dotazů posílaných po RS232 . . . . .	3
2.2	Popis příkazů . . . . .	5
2.2.1	Příkazy najždění na polohu a referenční značku . . . . .	5
2.2.2	Příkazy řízení koordinovaných pohybů . . . . .	6
2.2.3	Nastavení parametrů regulátorů . . . . .	9
2.2.4	Podpora ladění odezvy regulátorů . . . . .	11
2.2.5	Nulování, zastavení a odpojení regulátorů . . . . .	12
2.2.6	Potvrzování přijetí a dokončení příkazů . . . . .	12
2.2.7	Joystick a klávesnice . . . . .	13
2.2.8	Status systému . . . . .	13
2.2.9	Přímé řízení rychlosti . . . . .	13
2.2.10	Řízení vstupů a výstupů . . . . .	14
2.2.11	Systémové příkazy . . . . .	16
2.3	Konfigurace . . . . .	18
2.3.1	Doporučené řešení Hard Home pro MARS8 . . . . .	18
2.4	Triggery událostí . . . . .	20
2.5	Komparátory polohy . . . . .	21
2.6	Rozšířené příkazy pro MARS8B . . . . .	22
<b>3</b>	<b>Ovládání přes IIC rozhraní</b>	<b>23</b>
<b>4</b>	<b>Manuální ovládání</b>	<b>23</b>
<b>5</b>	<b>Konektory a připojení</b>	<b>23</b>
<b>6</b>	<b>Výrobce</b>	<b>26</b>

## 1 Specifikace systému pro víceosou regulaci

Elektronická řídicí jednotka **MARS 8** je určena pro regulaci polohy až osmi stejnosměrných motorů s inkrementálními čidly přírůstku polohy a jedné indexové značky na otáčku motoru. Pracovní plocha ovládaného zařízení může být vymezena elektro-mechanickými koncovými přepínači zapojenými do výkonových větví motorů nebo čidly limitní polohy s logickými výstupy. Pro zpřesnění určení výchozí polohy je možné využít kombinace koncových spínačů, čidel ve volitelné kombinaci s indexovým výstupem.

Příkazy pro jednotku lze zadávat z volitelné lokální klávesnice nebo je lze zadávat z nadřazeného počítače přes rozhraní *RS-232* nebo volitelně přes rozhraní *RS-485*.

Jednotka může být propojena s dalšími zařízeními komunikací CAN. Jednotka může být doplněna vstupem pro analogový joystick.

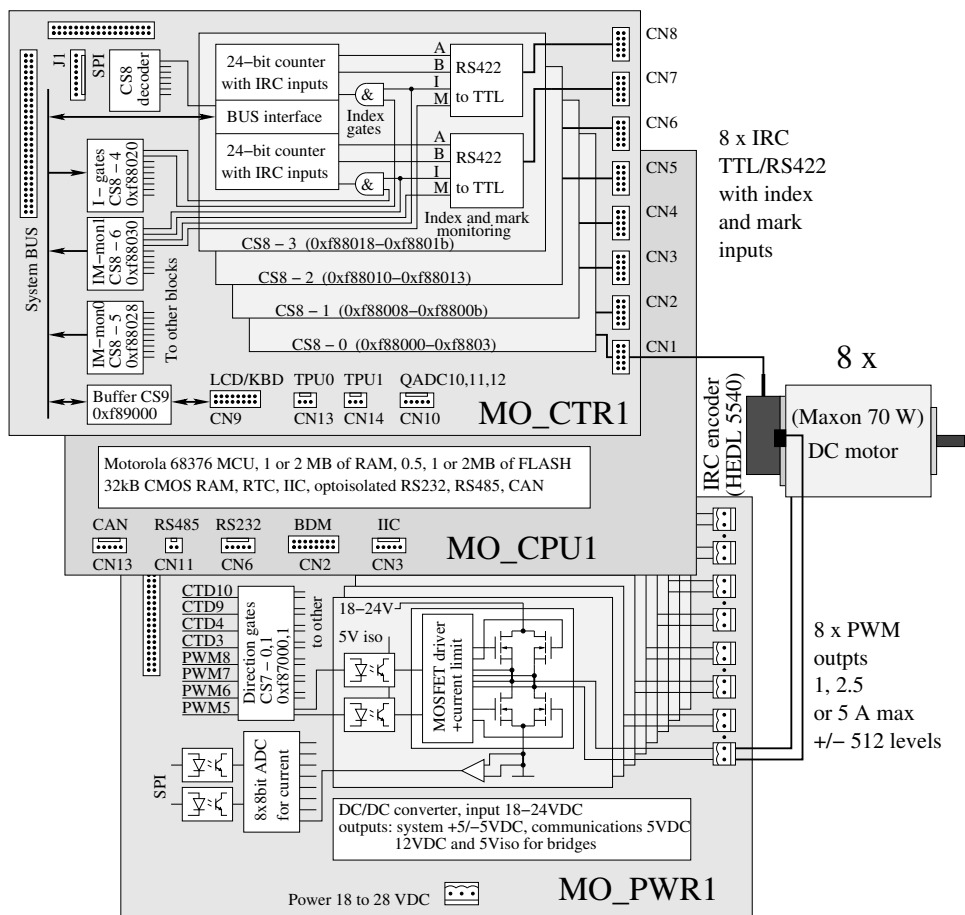
Mimo řízení motorů je možno využít množství logických vstupů a výstupů a až 12 analogových vstupů. Další periferie lze připojovat přes rozhraní SPI nebo  $I^2C$ . Dále je možno využít až 16 kanálů řízených časovacím koprocesorem. Jako digitální vstupy lze též číst indexové a koncové značky motorů.

### Technická data jednotky MARS 8:

veličina	hodnota	jednotky
Vnější napájecí napětí skříně	230±10%	VAC
Trvalý výkon vstupního transformátoru	150	W
Napájecí napětí elektroniky	24 (12 až 34)	VDC
Max. přípustný proud jedním motorem	5	A
- nadproudová ochrana	lokální elektronická pro každou osu+ měření a vyhodnocení proudu mikroprocesorem	
Vstup snímače polohy (IRC)	2 fázově posunuté signály + index + limit úrovně TTL nebo RS-422	
Max. frekvence IRC signálů	2	MHz
Vstupy snímačů joysticku	0 až 5	V
- předpokládaná centrální poloha	2.5	V
- citlivost vstupů joysticku	rozsah lze upravit pro konkrétní joystick odpory	
Jemné doladění citlivosti a nulové polohy joysticku	softwarově	
Generátor požadované polohy	lichoběžníkový průběh s nastavitelným zrychlením a maximální rychlostí konstantní rychlost s rozběhem a brzděním	
Regulace polohy motorů	vlečný PID regulátor s nelineárním tlumením	
Nastavování P, I a D konstant	softwarově	
Rozsah polohy motoru	±8000.000	po 0.001
- rozsah v periodách IRC signálu (4 fáze)	4000000	
Kalibrace absolutní polohy	automatické vyhledání volitelné kombinace koncových spínačů a indexu	

Pro přesné polohovací aplikace menších a středních rozměrů s vysokými nároky na přesnost je vhodná kombinace jednotky **MARS 8** s motory **MAXON RE 70 W / 42 VDC** s inkrementální čidly **HP HEDS 5540** s TTL výstupy nebo **HP HEDL 5540** s RS-422. Tato IRC čidla vysílají 500 period fázově posunutých signálů na otočku motoru a jsou doplněna indexovou značkou. Jednotka **MARS 8** je schopna měřit polohu s přesností 1/2000 otočky a provádět polohovou regulaci a najíždět na koncovou polohu s přesností ±1/2000 otočky.

Pro vymezení pracovní oblasti lze zapojit do výkonového výstupu na motory koncové spínače s diodami pro návrat z polohy za koncovým spínačem a pro rychlé



Obrázek 1: Blokové schéma desek elektroniky

zastavení při vyjetí z pracovní oblasti.

### Údržba.

Vlastní řídicí jednotka **MARS 8** nevyžaduje žádnou údržbu.

## 2 Ovládání z PC po lince RS232

Jednotka komunikuje s počítačem přes sériové asynchronní rozhraní. Požadované nastavení komunikačního portu je 9600 b/s, 8 bitů, žádná parita, 1 stopbit. Řízení toku dat je prováděno hardwarově signály **CST** a **RTS**. Rychlost komunikace je možné nastavovat z lokální klávesnice nebo příkazy po RS232 od 2400 b/s do 38400 b/s.

### 2.1 Tvar příkazů a dotazů posílaných po RS232

Příkaz se skládá z jména, operačního znaku a parametrů. Jednotlivé části mohou být odděleny mezerami.

**Jméno** je libovolná kombinace písmen a číslic začínající písmenem. V případě příkazů vztahujících se k jednotlivým motorům je jméno na konci doplněno o

znak motoru ( A, B, C až H) dále označený 'm'. Příkazy vstupů a výstupů obsahují číslo skupiny 'n', případně číslo trigeru 't' nebo komparátoru 'c'.

**Operační znak** definuje jestli se jedná o příkaz ( znak ':' ) nebo dotaz ( znak '?' ). Pro potvrzování může být použit znak '\\'.

**Parametr**, jehož význam je daný příkazem. Dále bude 'xxx.xxx' označováno desetinné číslo, 'xxx' celé číslo a 'x' číslice. Záporná čísla začínají znakem '-'.

Zprávy o dokončení operace nebo o chybě vysílané jednotkou MARS 8, které nejsou přímou odpovědí na předchozí dotaz, obsahují operační znak '!'. Řádky s hláškami pro ladění a informací o spuštění systému jsou uvozeny znakem '#'.

Některé příkazy nejsou v aktuální verzi software zatím implementovány. Tyto příkazy jsou zakomentovány v dvojitých závorkách [[ ]].

### Jména jednotlivých příkazů.

- Příkazy pro najíždění na polohu a referenční značku, více v odstavci 2.2.1  
Gm, GRm, APm, HHm, [[HH]], SETAPm
- Příkazy pro řízení koordinovaných pohybů, více v odstavci 2.2.2  
COORDGRP, COORDMV, COORDMVT
- Nastavení parametrů regulátorů, více v odstavci 2.2.3  
REGPm, REGIm, REGDm, REGS1m, REGS2m, REGMDm, REGMSm,  
REGACCM, REGMEEm, REGCFGm, REGDBGm, REGSFRQ, [[REGTY-  
PEm]]
- Podpora ladění odezvy regulátorů, více v odstavci 2.2.4  
REGDBGm, REGDBGHIS, REGDBGPRE, [[REGDBGGNS]], REGDBGGNR
- Nulování, zastavení a odpojení regulátorů, více v odstavci 2.2.5  
STOPm, STOP, PURGE, CLEARm, CLEAR, RELEASEm, RELEASE
- Potvrzování přijetí a dokončení příkazů, více v odstavci 2.2.6  
[[READY]], R, Rm, [[REPLY]]
- Joystick a klávesnice, více v odstavci 2.2.7  
[[KEYLOCK, JOYSTICKm, JOYSTICK, JOYOFFSm, JOYRESm, JOYHYSm,  
JOYCAL]]
- Status systému, více v odstavci 2.2.8  
STm, ST, STBSYBITS
- Přímé řízení rychlosti, více v odstavci 2.2.9  
SPDm, SPDTm
- Řízení vstupů a výstupů, více v odstavci 2.2.10  
PWMm, DIGO, DIGOn, DIGI, DIGIn, DIGM, DIGMn, TRIGt, CMPc, CM-  
PREPOc, CMPQUEm, CMPQUELISTm, CMPQUEFLUSHm, CMPQUEFROM-  
TRIGm, CMPQUEFROMTRIGFLUSH, CMPQUEFROMTRIGCHKSEQ, ADCx
- Systémové příkazy, více v odstavci 2.2.11  
VER, STAMP, ERRSTOP, IDLEREL, REGGOFLG, ECHO, RS323BAUD  
[[HEXLD, TEST, CFGNVSAVE, CFGDEFAULT]]

## 2.2 Popis příkazů

Následuje popis jednotlivých příkazů. Příkazy označené '\*opt' jsou k dispozici jen pro některé konfigurace systému.

### 2.2.1 Příkazy najíždění na polohu a referenční značku

Jméno	Op	Parametry	Funkce
Gm	:	xxx.xxx	Najetí na absolutní polohu
		<-8000,8000.000>	

Najede motorem 'm' na absolutní polohu 'xxx.xxx'.

Jméno	Op	Parametry	Funkce
GRm	:	xxx.xxx	Najetí na relativní polohu
		<-8000,8000.000>	

Relativní pohyb motoru 'm' o vzdálenost 'xxx.xxx'. Za referenční polohu je brána aktuální požadovaná poloha nebo po nastavení bitu 1 příkazem **REGGFLG** koncová poloha předcházejícího příkazu **Gm** nebo **GRm**. Příkaz používá modifikovaný generátor lichoběžníkového profilu, který nevyhodnocuje směr pohybu podle absolutní hodnoty, což umožňuje využití relativního pohybu pro rotační polohování bez omezení způsobeného rozsahem 24-bitového čítače polohy.

Jméno	Op	Parametry	Funkce
APm	?		Aktuální poloha

Vrací aktuální polohu motoru 'm' ve formátu 'xxx.xxx'.

Jméno	Op	Parametry	Funkce
HHm	:		Referenční poloha 'm'

Nalezne referenční značku a vynuluje odečet polohy pro motor 'm'.

Jméno	Op	Parametry	Funkce
SETAPm	:	xxx.xxx	Nastaví odečet polohy 'm'
		<-8000,8000.000>	

Pro vybranou osu 'm' dojde k nastavení hardwarového odečtu polohy na zadanou hodnotu. Před provedením příkazu musí být pro daný motor vypnuta regulace, jinak příkaz není proveden a je vrácena chyba.

### 2.2.2 Příkazy řízení koordinovaných pohybů

Jméno	Op	Parametry	Funkce
COORDGRP	:	m1,m2,...	Výběr motorů pro koordinovaný pohyb
		A..H	

Nastaví seznam motorů 'm1,m2,...', které mohou být řízeny koordinovaně příkazy **COORDMV**. Pokud je seznam motorů prázdný, dojde k zrušení předchozích nastavení. Pro úspěšné provedení příkazu **COORDGRP** je nutné splnit následující podmínky: motory, které jsou uvedeny v seznamu nesmí být v době vykonání příkazu ve stavu průběhu příkazu nebo v chybovém stavu, viz 2.2.8; žádný motor se nesmí v seznamu opakovat; pro současnou verzi software musí být motory uvedeny ve vzestupném pořadí.

Jméno	Op	Parametry	Funkce
COORDMV	:	xxx,xxx,...	Koordinovaný pohyb v motorových souřadnicích
		<-8000,8000.000>,...	

Příkaz přidá do fronty další požadovanou polohu motorů definovaných příkazem **COORDGRP:m1,m2,...** a zároveň se zadaná poloha stane novou koncovou polohou celého pohybu. Pokud jsou již ve frontě požadovaných poloh dříve zadané polohy, dojde po přidání polohy k přepočítání průjezdových rychlostí jednotlivých zlomových poloh. Plánování je provedeno vždy tak, aby pro žádný motor nedošlo k překročení pro něj definované maximální rychlosti a maximálního zrychlení a aby pohyb skončil v poslední zadané poloze. Výpočet bere v úvahu pro každý přidávaný úsek pohybu hodnoty rychlosti a zrychlení platné v okamžiku přidání nové koncové polohy. Počet volných pozic ve frontě je v současné verzi software nastaven na 200, přičemž při poklesu volných pozic pod 50 je ve stavu systému **ST?** nastaven příznak naplnění fronty. Je-li signalizováno naplnění fronty je potřeba s dalším plněním počkat do té doby, než je z fronty vyraženo dostatečné množství již provedených úseků pohybu. Příkaz **COORDMV** je přijat pouze tehdy, když jednotlivé motory definované příkazem **COORDGRP** již vykonávají předchozí koordinovaný příkaz, nebo pokud nejsou zaneprázdněny příkazem z jiné skupiny příkazů. Není-li některá z těchto podmínek splněna příkaz je ignorován a je vrácen řádek **ERROR**. Pokud jsou všechny podmínky splněny dojde po přijetí příkazu k aktivaci koordinovaného pohybu. Po dobu zaneprázdnění koordinovaným pohybem není možné zadat pro zúčastněné motory jiný výkonný příkaz (například **HH** nebo **Gm**). Všechny takovéto příkazy jsou ignorovány a je vrácena chyba. Koordinovaný pohyb je možné ukončit příkazem **STOP** nebo lze zastavovat jednotlivé motory, pak ostatní motory dokončí již naplánované pohyby. V případě, že jsou zastaveny všechny zúčastněné motory nebo dojde na některém z nich k chybě, dojde k deaktivaci koordinátoru a zbývající položky ve frontě jsou zrušeny.

Jméno	Op	Parametry	Funkce
COORDMVT	:	mintim,xxx,xxx,...	Koordinovaný pohyb s časováním
		<0,10000000>	
		<-8000,8000.000>,...	

Příkaz přidá do fronty další požadovanou polohu motorů definovaných příkazem **COORDGRP:m1,m2,...** a spustí pohyb. Od příkazu **COORDMV** se liší tím, že pro každý úsek je zadán parametrem **mintim** minimální čas v milisekundách, po který bude pohyb trvat.

Jméno	Op	Parametry	Funkce
COORDRELMVT	:	<b>mintim,xxx,xxx,...</b>	Relativní koordinovaný pohyb s časováním
		<0,10000000>	
		<-8000,8000.000>,...	

Příkaz přidá do fronty požadavek na relativní pohyb motorů definovaných příkazem **COORDGRP:m1,m2,...** a spustí pohyb.

Jméno	Op	Parametry	Funkce
COORDSPLINET	:	<b>mintim,order,a11,a12,a13...,a21,a22,...</b>	Koordinovaný pohyb po spline úseku
		<0,10000000>	Minimální čas v milisekundách pro dosažení příštího bodu
		<1, 5>	Řád polynomu, první řád odpovídá COORDRELMVT
		<-8000,8000.000>,...	

Příkaz přidá do fronty požadavek na pohyb podle polynomu n-tého řádu a spustí pohyb. Jedná se o motory vybrané příkazem **COORDGRP:m1,m2,...**. Pro každý motor je výpočet okamžité polohy určen vztahem

$$rp_m = ep_m + \sum_{i=1}^{order} a_{m,i} \cdot p$$

kde  $rp_m$  určuje v daném okamžiku požadovanou polohu pro motor  $m$ ,  $ep_m$  představuje koncovou polohu motoru  $m$  po dokončení pohybu podle minulého úseku. Parametry polynomu  $a_{m,i}$  pak určují průběh polohy v daném úseku. Přitom posice v daném úseku je určena parametrem  $m \in \langle 0, 1 \rangle$ .

Jméno	Op	Parametry	Funkce
COORDAP	?		Výše aktuální polohy v rámci interpolace
		msec_time,seg_sqn,seg_par,p1,p2,...	Čas měřený jednotkou od spuštění jednotky, číslo úseku v sekvenci příkazů, aktuální hodnota parametru v rámci úseku a polohy jednotlivých motorů

Zjištění aktuální polohy. Protože parametr, čas i sekvenční číslo je po inkrementech počítané až do plných 32 bitů, kdy dojde po naplnění k dokončení úseku a u času a sekvenčního čísla k přetečení do nuly, tak je po urychlení přenosu a zmenšení

rozsahu využití zkrácení vypisovaných hodnot u času a sekvenčního čísla maskováním a přenosem je spodních bitů hodnoty, měřený jednotkou od spuštění jednotky je omezen na interval  $\langle 0, CMDTIMEMASK \rangle$ , číslo úseku v sekvenci příkazů je omezeno na interval  $\langle 0, COORDSQNMASK \rangle$  a aktuální hodnota parametru v rámci úseku je omezena na horních  $n$  nejvýznamnějších bitů do intervalu  $\langle 0, 2^{COORDPARBITS} - 1 \rangle$ .

$m \in \langle 0, 1 \rangle$ .

Jméno	Op	Parametry	Funkce
COORDAPS	?		Vyšle aktuální polohu a rychlosti motorů v rámci interpolace
		msec_time, seg_sqn, seg_par, p1, p2, ..., s1, s2, ...	Čas měřený jednotkou od spuštění jednotky, číslo úseku v sekvenci příkazů, aktuální hodnota parametru v rámci úseku, polohy jednotlivých motorů a rychlosti jednotlivých motorů

Vrácená hodnota obsahuje navíc okamžité rychlosti jednotlivých motorů.

Jméno	Op	Parametry	Funkce
COORDRP	?		Vyšle aktuální referenční (požadované) polohy v rámci interpolace
		msec_time, seg_sqn, seg_par, p1, p2, ...	Čas měřený jednotkou od spuštění jednotky, číslo úseku v sekvenci příkazů, aktuální hodnota parametru v rámci úseku a požadované polohy jednotlivých motorů

Vrácené hodnoty odpovídají aktuálním požadovaným polohám motorů v rámci právě zpracovávaného úseku/požadavku z fronty.

Jméno	Op	Parametry	Funkce
COORDEP	?		Vyšle aktuální referenční (požadované) polohy v rámci interpolace
		msec_time, seg_sqn, seg_par, p1, p2, ...	Čas měřený jednotkou od spuštění jednotky, číslo úseku v sekvenci příkazů, aktuální hodnota parametru v rámci úseku a požadované polohy jednotlivých motorů

Vrácené hodnoty odpovídají plánovaným koncovým polohám motorů po provedení všech příkazů z fronty požadavků.



Jméno	Op	Parametry	Funkce
COORDEP	?		Vyšle aktuální referenční (požadované) polohy v rámci interpolace
		msec_time,seg_sqn,seg_par,p1,p2,...	Čas měřený jednotkou od spuštění jednotky , číslo úseku v sekvenci příkazů , aktuální hodnota parametru v rámci úseku a koncové polohy motorů

Vrácené hodnoty odpovídají plánovaným koncovým polohám motorů po provedení všech příkazů z fronty požadavků.

Jméno	Op	Parametry	Funkce
CMDTIMEMASK	:	time_mask	Omezení/maskování výpisu času
		<0,4294967295>	

Jméno	Op	Parametry	Funkce
COORDSQNMASK:		sqn_mask	Omezení/maskování výpisu sekvence čísla úsek
		<0,4294967295>	

Jméno	Op	Parametry	Funkce
COORDPARBITS	:	par_bits	Formát výpisu parametru pozice v rámci úseku na horních <b>par_bits</b> bitů.
		<0,32>	

### 2.2.3 Nastavení parametrů regulátorů

Jméno	Op	Parametry	Funkce
REGPm	: ?	xxx	Proporcionální kon. pro 'm'
		<0,255>	
REGIm	: ?	xxx	Integrační konstanta pro 'm'
		<0,255>	
REGDm	: ?	xxx	Derivační konstanta pro 'm'
		<0,255>	
REGS1m	: ?	xxx	1. pomocná konstanta 'm'
		<0,255>	
REGS2m	: ?	xxx	2. pomocná konstanta 'm'
		<0,255>	

Příkazy umožňují nastavení a čtení konstant regulátoru motoru 'm'. Přesný význam parametrů a rozsah povolených hodnot závisí na zvoleném typu regulátoru pro daný motor. Pro **PID** slouží parametry **S1** a **S2** k potlačení necitlivosti výstupních budičů.

Jméno	Op	Parametry	Funkce
REGMSm	: ?	xxx	Maximální rychlost pro 'm'
		<0,30000>	

Nastaví maximální rychlost pro motor 'm' ve formátu 'xxx'. Rychlost se použije pro příkazy najetí na polohu a jako limit rychlosti při řízení joystickem. Hodnota je zadávána přímo v přírůstku IRC násobeného 256 za periodu vzorkování.

Jméno	Op	Parametry	Funkce
REGACcm	?	xxx	Zrychlení pro motor 'm'
		<0,30000>	

Nastaví zrychlení pro motor 'm' ve formátu 'xxx'. Hodnota zrychlení se použije pro příkazy najetí na polohu při lichoběžníkovém profilu rychlosti. Při konfiguraci s lichoběžníkovým profilem rychlosti se hodnota používá pro zpomalení při příkazu **STOP** a přerušení pohybu dalším příkazem **Gm**, **GRm**. Zrychlení se dále používá při řízení rychlosti při příkazech **SPDm** a **SPDTm**. Hodnota odpovídá přírůstku rychlosti za periodu vzorkování.

Jméno	Op	Parametry	Funkce
REGMEEm	: ?	xxx	Maximální PWM pro 'm'
		<0,30000>	

Hodnota omezuje maximální plnění výstupu PWM pro motor 'm'. Snížením max. PWM se snižuje v příslušném poměru k napájecímu napětí maximální napětí, které může být přivedeno na motor. V praxi se používá k ochraně motorů stavěných na menší jmenovité napětí, než je napájecí napětí. Hodnota 32000 odpovídá plnému napětí.

Jméno	Op	Parametry	Funkce
REGMDm	: ?	xxx	Maximální povolená regulační odchylka pro 'm'
		<0,30000>	

Definuje maximální povolenou regulační odchylku pro motor 'm' v počtu IRC impulzů. Při větší hodnotě odchylky a nastaveném příznaku MD2E v konfiguračním slově pro danou osu dojde k nastavení chyby a zastavení pohybu.

Jméno	Op	Parametry	Funkce
REGCFGm	: ?	xxx	Konfigurační slovo pro 'm'
		<0,30000>	

Konfigurace definuje pro motor 'm' průběh rychlosti při pohybu z jedné polohy do druhé, způsob nalezení nulové polohy a přepočty logických a fyzických souřadnic. Zadané dekadické číslo je interpretováno jako bitové pole xxxxxMxTxLCRDSSS. Kompletní popis je uveden v 2.3.

Jméno	Op	Parametry	Funkce
REGSFRQ	: ?	x	Vzorkovací frekvence regulátorů
*opt		<200,2500>	

Nastaví vzorkovací frekvenci regulační smyčky všech motorů. Zadávaná hodnota přímo odpovídá požadované vzorkovací frekvenci v Hz, která je dále systémem zakrouhlena na nejbližší možnou hardwarově realizovatelnou hodnotu. Skutečně nastavenou hodnotu je možné zpětně číst. Defaultní hodnota je 1000 Hz. Při volení vyšší frekvence je nutné zvážit celkové zatížení systému a při použití všech osmi os by již frekvence neměla být zvyšována nad defaultní hodnotu.

#### 2.2.4 Podpora ladění odezvy regulátorů

Jméno	Op	Parametry	Funkce
REGDBGm	:	x	Povolení sledování 'm'
		<0,1>	

Povolení ukládání historie pohybu motoru 'm', zároveň označuje motory, na které se budou vztahovat ostatní **REGDBGxxx** příkazy. Pro motory s nastaveným příznakem jsou vždy po spuštění pohybu ukládány hodnoty skutečné rychlosti a hodnoty vysílané do PWM budičů.

Jméno	Op	Parametry	Funkce
REGDBGHIS	:	xxxx	Vyčtení historie pohybu
		<0,3000>	

Příkaz pro vyčtení historie pohybu, parametr udává počet vyslaných čísel. Každé číslo je vysláno na samostatné řádce.

Jméno	Op	Parametry	Funkce
REGDBGPRE	:	xxxx	Příprava dat pro sledování odezvy
		<0,3000>	

Uloží do paměti data pro příkazy sledování odezvy motoru. Parametr udává počet ukládaných hodnot. Hodnoty se vysílají na samostatných řádkách.

Jméno	Op	Parametry	Funkce
REGDBGGR	:		odezva regulace motoru

Nastavuje požadavek na změnu polohy pro regulátory motorů s nastaveným **REGDBGm** na 1 podle hodnot uložených příkazem **REGDBGPRE**. Příkaz **REGDBGHIS** po provedení příkazu umožňuje vyčíst skutečný průběh pohybu motoru.

### 2.2.5 Nulování, zastavení a odpojení regulátorů

Jméno	Op	Parametry	Funkce
CLEAR <sub>m</sub>	:		Vypnutí řízení a nulování 'm'
CLEAR	:		Vypnutí řízení a nulování všech motorů

Vypnutí řízení motoru 'm' a vynulování odečtu polohy a koncové požadované polohy pohybu.

Jméno	Op	Parametry	Funkce
STOP <sub>m</sub>	:		Zastavení pohybu motoru 'm'
STOP	:		Zastavení pohybu všech motorů

Zastavení pohybu motoru 'm', regulace však pokračuje. Při povoleném lichoběžníkovém průběhu rychlosti dojde k plynulému zpomalení a pak teprve k úplnému zastavení.

Jméno	Op	Parametry	Funkce
PURGE	:		Zastaví regulace s chybou

Zastaví regulace motorů s chybou a chyby vynuluje, ostatní motory pokračují v započaté činnosti.

Jméno	Op	Parametry	Funkce
RELEASE <sub>m</sub>	:		Odpojí regulátor a zastaví 'm'
RELEASE	:		Odpojí regulaci a zastaví všechny motory

Odpojí regulátory a uvolní motory pro manuální pohyb nebo pro uklidnění.

### 2.2.6 Potvrzování přijetí a dokončení příkazů

Jméno	Op	Parametry	Funkce
R	:		Ohlas ukončení jen jednou

Vyšle jedno 'R!' nebo 'FAIL!' po skončení právě probíhajících operací nebo okamžitě, není-li žádný motor zaneprázdněn.

Jméno	Op	Parametry	Funkce
Rm	:		Ohlas pro jednotlivý motor

Vyšle 'Rm!' nebo 'FAILm!' po skončení aktivit motoru 'm'.

### 2.2.7 Joystick a klávesnice

### 2.2.8 Status systému

Jméno	Op	Parametry	Funkce
STm	?		Status motoru 'm'

Vrací status motoru 'm'. Dekadické číslo je třeba interpretovat jako bitové pole.

Bit	Význam
0	Povolen odečet IRC
1	Povolen regulátor
2	Povolen generátor
3	Chyba
4	Probíhá minulý příkaz
5	Zapnuto ukládání průběhu regulace
6	Probíhá koordinovaný pohyb
7	Fronta požadavků dosáhla hranice zaplnění
16	Vypnuté výkonové napájení
17	Vypnuto tlačítkem power stop

Jméno	Op	Parametry	Funkce
ST	?		Status všech motorů

Vrací logický součet stavu všech motorů.

Jméno	Op	Parametry	Funkce
STBSYBITS	?		Informace o probíhajících operacích

Vrací do dekadické hodnoty zakódované jednobitové příznaky probíhající operace ze všech motorů. Operace na motoru 'A' je signalizována přidáním hodnoty 1, motoru 'B' hodnoty 2, motoru 'C' hodnoty 4 atd.

### 2.2.9 Přímé řízení rychlosti

Jméno	Op	Parametry	Funkce
SPDm	:	xxx	Točit zadanou rychlostí
*opt		<-32000,32000>	

Zadává požadavek na točení motoru 'm' zadanou rychlostí. Změna rychlosti probíhá se zrychlením zadaným příkazem **REGACCm**.

Jméno	Op	Parametry	Funkce
SPDTm	:	xxx,yyy	Točit rychlostí 'xxx' po max dobu 'yyy'
*opt		<-32000,32000>	
		<0,32000>	

Zadává požadavek na točení motoru 'm' zadanou rychlostí 'xxx'. Změna rychlosti probíhá se zrychlením zadaným příkazem **REGACCm**. Točení pokračuje po 'yyy' period vzorkování (standardně krok 1 ms). Nepřijde-li další příkaz pro daný motor do této doby dojde k plynulému zastavení pohybu.

### 2.2.10 Řízení vstupů a výstupů

Jméno	Op	Parametry	Funkce
PWMm	:	xxx	Přímé nastavení PWM
*opt		<-32000,32000>	

Přímé nastavení PWM výstupu motoru 'm' na hodnotu 'xxx'

Jméno	Op	Parametry	Funkce
DIGO	:	xxxxx	Nastavení digitálních výstupů
DIGOn	:	xxxxx	Rozšířené nastavení výstupů
*opt		<0,65535>	

Nastaví digitální výstupy 0 až 15 podle jednotlivých bitů binární reprezentace parametru. Varianta **DIGO** nastavuje výstupy ve skupině 0, varianta **DIGOn** umožňuje pracovat i s dalšími skupinami výstupů.

Jméno	Op	Parametry	Funkce
DIGI	?		Čtení stavu digitálních vstupů
DIGIn	?		Rozšířené čtení vstupů
*opt			

Přečte digitální vstupy 0 až 15 a vyše přečtený stav jako dekadickou hodnotu. Varianta **DIGI** pracuje pouze se skupinou 0, varianta **DIGIn** umožňuje pracovat i s dalšími skupinami vstupů.

Jméno	Op	Parametry	Funkce
DIGM	:	mask, xor	Modifikace stavu digitálních výstupů
DIGMn	:	mask, xor	Rozšířená modifikace stavu nastavení výstupů
*opt		<0,65535>, <0,65535>	

Modifikuje digitální výstupy 0 až 15 podle jednotlivých bitů binární reprezentace parametrů. Z původní hodnoty na výstupech jsou nejdříve vynulovány bity odpovídající jedničkám v masce, hodnota je dále xorována a vyslána na výstupy. Zápis v jazyce C: `new=(old&~mask)^xor`; Varianta **DIGM** nastavuje výstupy ve skupině 0, varianta **DIGMn** umožňuje pracovat i s dalšími skupinami výstupů.

Jméno	Op	Parametry	Funkce
TRIGt	:	s,m,do	Nastavení triggeru
TRIG	:	s,m,n,mask,xor	
*opt			

Triggery událostí jsou popsány v odstavci 2.4.

Jméno	Op	Parametry	Funkce
CMPc	:	f,m,p,do	Komparátory polohy
CMPc	:	f,m,p,n,mask,xor	
*opt			

Komparátory jsou popsány v odstavci 2.5.

Jméno	Op	Parametry	Funkce
CMPREPOc	:	poffs	Offset polohy při periodickém spouštění komparátoru
*opt, ver 2.1			

Komparátory jsou popsány v odstavci 2.5.

Jméno	Op	Parametry	Funkce
CMPQUEm	:	f,p,r,do	Přidání komparátoru do třídného seznamu dané osy 'm'. Výstupy aplikuje na I/O skupinu 0
CMPQUEm	:	f,p,r,n,mask,xor	Totéž s volbou I/O skupiny 'n' a modifikace výstupů s maskou
CMPQUEFLUSHm			Vyprázdní položky komparátorů ve frontě dané osy 'm'
CMPQUELISTm	:		Vypíše obsah fronty pro ladící účely, funkce není míněna pro použití při běžném provozu, protože může zpožďovat ostatní operace
CMPQUEFROMTRIG		mf,t,dm[,dn]{,[f,]o,n,mas	Provádí naplnění fronty komparátorů motoru 'm' podle polohy uložené při aktivaci triggeru 't'
CMPQUEFROMTRIGFLUSH			Vymazání všech akcí zadaných přes CMPQUEFROMTRIGm
CMPQUEFROMTRIGCHKSEQ			Kontrolní sekvence
*opt			

Komparátory jsou popsány v odstavci 2.5.

Jméno	Op	Parametry	Funkce
ADCa	?		Čtení analogového vstupu
*opt			

Vrací hodnotu analogového vstupu 'a'.

### 2.2.11 Systémové příkazy

Jméno	Op	Parametry	Funkce
VER	?		Vrací verzi software

Slouží k zjištění verze firmware jednotky.

Jméno	Op	Parametry	Funkce
STAMP	:	ssss	Synchronizační značka pro komunikaci

Po přijetí řádku s příkazem 'STAMP:' následovaného libovolným řetězcem dojde k okamžitému odeslání řetězce 'STAMP=' a přijatého řetězce. Tento příkaz umožňuje nadřazenému počítači přesnou kontrolu, kdy prošel zadaný příkaz všemi odchozími a odezva všemi příchozími frontami systému.



Jméno	Op	Parametry	Funkce
ERRSTOP	: ?	x	Zastavování všech pohybů při chybě
		<0,1>	

Je-li tato funkce nastavena (hodnota 1), dojde při vzniku chyby na jedné ose k plynulému zastavení všech ostatních pohybů.

Jméno	Op	Parametry	Funkce
IDLEREL	: ?	xxx	Vypínání regulátorů při nečinnosti
		<0,4000000>	

Doba ve vteřinách po které při nečinnosti všech motorů dojde k uvolnění/vypnutí regulace. Hodnota 0 zakazuje tuto funkci.

Jméno	Op	Parametry	Funkce
REGGOFLG	: ?	x	Příznak pro modifikaci příkazu Gm
		<0,1>	

Je-li tento příznak nastaven (hodnota 1), dojde při příkazu k nejetí na požadovanou pozici nejdříve k vyhodnocení, lze-li původní pohyb prodloužit nebo zkrátit. Při kladném výsledku testu dojde pouze ke změně cílové polohy pohybu. Není-li příznak povolen nebo při záporném výsledku testu dojde mezi pohyby k zastavení motoru.

Jméno	Op	Parametry	Funkce
ECHO	: ?	x	Znakové echo pro znaky z RS232
		<0,1>	

Při zapnutí funkce echo je každý přijatý znak okamžitě vyslán zpět z jednotky.

Jméno	Op	Parametry	Funkce
RS232BAUD	:	xxx	Změna přenosové rychlosti
		<1200,38400>	

Okamžitě po přijetí tohoto příkazu dojde ke změně přenosové rychlosti interface RS232 na jednotce **MARS 8**. Nadřazený systém musí po odeslání tohoto příkazu změnit odpovídajícím způsobem nastavení svého komunikačního interface, jinak dojde ke ztrátě komunikace. Po zapnutí jednotky **MARS 8** je rychlost nastavena na 9600 baud.

## 2.3 Konfigurace

Nastavení konfigurace pro jednotlivé osy příkazem REGCFGm určuje průběh rychlosti při pohybu z jedné polohy do druhé a způsob nalezení nulové polohy. Zadané dekadické číslo je interpretováno jako bitové pole.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Příznaky	x	x	x	x	Y	M	x	T	P	L	C	R	D	S	S	S

Význam jednotlivých příznaků je uveden v následující tabulce.

Příznaky	Význam
SSS	rychlost hledání dorazu při HH bude $REGMSm / 2^{SSS}$
D	počáteční směr hledání dorazu při HH
R	používá se značka otáčky z HP HEDS pro nalezení HH
C	hledat střed značky
L	používá se limitní switch
P	polarita limitního spínače
T	používat lichoběžníkový profil rychlosti a plynulé brždění
M	chyba při odchylce polohy větší než REGMDm
Y	osa pracuje v cyklickém režimu
LCR=111	jen hledání značky
CR=11	hledání pouze středu značky

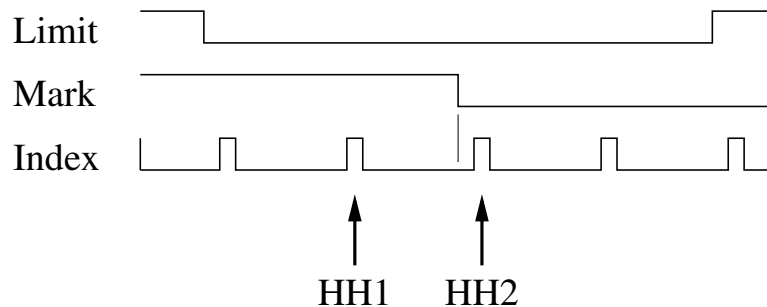
Pro usnadnění výpočtu konfigurace jsou dále uvedeny všechny kombinace bitů LCR. K číslu je nutné přičíst pouze 8 (D) pro změnu počátečního směru, snížení rychlosti (SSS) a 256 (T) pro plynulé rozjezdy.

Hodnota	Význam
0	hledat jen koncový vypínač motoru
16	najít první značku otáčky od koncového vypínače
32	hledat střed první značky od koncového vypínače
48	hledat střed značky
64	hledat jen limitní spínač
80	najít první značku otáčky od limitního spínače
96	hledat střed první značky od limitního spínače
112	pouze hledat značku

### 2.3.1 Doporučené řešení Hard Home pro MARS8

Pro ochranu mechaniky je vhodné do série s motorem zařadit limitní spínače. Nejlepší je takové řešení, kdy při odpojení motoru dojde zároveň k jeho zastavení zkratováním přes odpor nebo diodu. Po přidání diod ke spínačům umožňuje vyjetí z limitní polohy. V případě bez diod je nutné osu navrátit do pracovního rozsahu manuálně. Pokud je mechanika vybavena mechanickými dorazy, které ztlumí při výjimečné situaci tvrdý náraz pohonu, pak není nutné limitní spínače použít. Jednotka vyhlásí v takovém případě chybu v důsledku překročení limitního proudu.

Pro přesné a jednoznačné definování počáteční kalibrační polohy (hard home) je nejvýhodnější použít kombinace přesné optické značky otáčky (**Index**) optického



Obrázek 2: Doporučený průběh signálů pro Hard Home

snímače a mechanického spínače (označovaného jako referenční značka nebo **Mark**), který přepíná pouze jednou v celém pracovním rozsahu příslušné osy. Průběh logických signálů referenční značky **Mark**, značky otáčky snímače **Index** a výkonové ochrany **Limit** jsou znázorněny na obrázku 2.

Po nalezení počáteční polohy příkazem **HHm** odpovídá nulová poloha HH poloha zvolenému bodu **HH1** nebo **HH2**. Volba mezi bodem **HH1** a **HH2** se provádí volbou polaritu spínače značky **Mark** (REGCFG bit 7). Počáteční směr hledání počáteční polohy (REGCFG bit 3) musí být zvolen tak, aby se motor při spuštění příkazu **HHm** začal pohybovat k poloze, ve které dojde k přepnutí spínače **Mark**. Pokud je bit 7 = 0, tak je značka považována za aktivní v log 1 (HI). Pokud je bit 7 = 1, pak je značka aktivní v log 0 (LO). Bit 3 = 0 určuje kladný směr počátečního hledání značky **Mark** v oblasti s neaktivní značkou. Hodnota 1 bitu 3 nastavuje počáteční směr na záporný. Je-li značka při spuštění příkazu **HHm** aktivní dojde nejdříve k vyjetí z oblasti vymezené značkou v opačném směru než je směr definovaný bitem 3.

Algoritmus hledání nulové polohy je pro doporučenou konstrukci spínačů následující:

1. Po příkazu **HHm** je nejdříve otestováno je-li osa v oblasti aktivní značky **Mark**. Pokud ano, pak algoritmus přejde na krok 2., pokud ne, tak na krok 3.
2. Motor se postupně začne pohybovat rychlostí  $REGMSm/2^{bity0..2}$  z oblasti aktivní značky **Mark**. Směr pohybu je opačný, než směr určený bitem 3.
3. Motor se začne pohybovat ve směru definovaném bitem 3, rychlosti  $REGMSm/2^{bity0..2}$ .
4. Je očekávána změna hodnoty vstupu značky **Mark** do aktivní úrovně, po jejím nalezení dojde k postupnému zpomalení.
5. Motor se rozběhne v opačném směru rychlostí ještě  $4\times$  menší, než je počáteční rychlost hledání značky.
6. Po nalezení hrany značky je aktivována detekce značky otáčky **Indexu**.
7. Po nalezení **Indexu** dojde k vynulování polohy a postupnému zastavení Motoru.

Postup zaručuje, že je vždy nalezena shodná aktivace indexu i při nezanedbatelné hysterezi spínače značky **Mark**. Hrana značky i index jsou v kritickém průchodu hledány při nízké rychlosti pro zajištění vyšší přesnosti. Index je zachycen hardware s přesností na 1 impuls IRC.

Pro zaručení správné funkce algoritmu je nutné zajistit, aby se rozptýl rozepnutí spínače značky **Mark** nekryl s impulzem **Indexu**. Výhodou použití jediného přepnutí spínače značky **Mark** uprostřed pracovní oblasti je eliminace zajíždění do

limitních poloh a případného rušení vznikajícího na limitních spínačích a dále možnost zvolit mechanicky vhodnou počáteční polohu celého systému, ke které směřuje počáteční pohyb všech os.

Hodnota **REGCFGm** pro takto řešené hledání počáteční polohy bude kombinací

- MRS\_CFG\_MD2E\_m bit 10 maska 1024 - překročení regulační odchylky vede k chybě
- MRS\_CFG\_SMTH\_m bit 8 maska 256 - plynulé rozjezdy a zastavení
- MRS\_CFG\_HPS\_m bit 7 maska 0/128 - polarita značky **Mark**
- MRS\_CFG\_HLS\_m bit 6 maska 64 - používá se značka **Mark**
- MRS\_CFG\_HRI\_m bit 4 maska 16 - používá se signál **Index** z IRC snímače
- MRS\_CFG\_HDIR\_m bit 3 maska 0/8 - počáteční směr pohybu
- Snížení počáteční rychlosti pohybu 0 až 7

Pro výše uvedený příklad bude konfigurace pro nalezení pravého bodu (**HH2**):

značka aktivní v 1, počáteční směr mimo značku je záporný,

počáteční rychlost  $REGMSm/4$

$$1024 + 256 + 64 + 16 + 8 + 2 = 1370$$

Pro vyhledání leví možnosti (**HH1**) pak bude potřeba nastavit:

značka aktivní v 0, počáteční směr mimo značku je kladný,

počáteční rychlost  $REGMSm/4$

$$1024 + 256 + 128 + 64 + 16 + 2 = 1490$$

Vstupy jednotky jsou optimalizované pro diferenční logiku signálů (veden pozitivní i negativní signál). Toto řešení je nejlépe odolné proti rušení a standardně se používá u motoru Maxon s IRC čidly HP HEDL. Pokud není diferenční signál k dispozici, je možné zapojit pouze jeden z komplementárních vstupů. Vstupy jsou na desce impedančně zakončeny proti aktivně udržovaným úrovním napětí okolo 2.5 V. Zároveň je zakončení navrženo tak, aby pozitivní a negativní vstupy byly mírně rozvážené aby byla vstupní hodnota stabilní a definovaná i v případě rozpojeného vstupu. Rozvážení umožňuje vstupy využít i pro přímé připojení samotného spínače, který pozitivní nebo negativní vstup připojuje k logické zemi nebo napájení.

## 2.4 Triggery událostí

Nastavení triggeru události se provádí příkazem

```

TRIGt:s,m,do
TRIGt:s,m,n,mask,xor
t    .. číslo triggeru 0 až 3
s    .. zdroj triggeru 0 až 15 ( -1 odpojení triggeru )
      s.0 .. s.3 .. číslo zdroje v rámci skupiny
      s.4    .. náběžná hrana
      s.5    .. sestupná hrana
      s.6    .. vyslat stav na vstupech
      s.7    .. nastavit digitální výstupy
      s.8    .. filtr zákmitů
      s.12 .. s.15 výběr skupiny (I/O jednotky)
m    .. maska motorů, které se mají zastavit a vyslat polohu
      m.0    .. zastavit A
      m.1    .. zastavit B
      ...
      m.7    .. zastavit H
      m.8    .. vysílat A
      m.9    .. vysílat B
      m.15   .. vysílat H
do   .. hodnota na digitální výstupy při triggeru
n    .. číslo skupiny výstupů
mask .. maska nastavovaných bitů
xor  .. požadovaná hodnota maskovaných bitů
      pro bity neuvedené v masce dojde ke změně výstupu

```

Triggery vysílají

```

TGt!di,ma,mb,mc
t    .. číslo triggeru 0 nebo 1
di   .. co bylo na vstupech v době triggeru, není-li
      vysílání povoleno, vyšle "N"
ma,mb,mc .. podle volby m v nastavení vysílá polohu IRC

```

V základní verzi software odpovídají zdroje triggerů 0 a 1 vstupům **I1 (CN13)** a **I2 (CN14)** na desce **MO\_CTR1**. Tyto vstupy jsou schopné zaznamenat i změny trvající po kratší dobu než 1 ms. Hodnoty 2 až 15 odpovídají příslušným digitálním vstupům ze skupiny 0, které jsou čteny též příkazy **DIGI?** a **DIGI0?**.

## 2.5 Komparátory polohy

Nastavení komparátoru polohy se provádí příkazem

CMPc:f,m,p,do  
 CMPc:f,m,p,n,mask,xor  
 c .. číslo komparátoru 0 až 3  
 f .. příznaky akce ( 0 vypnutí komparátoru )  
   f.0 .. čekat na **AP<sub>m</sub>** > **p**  
   f.1 .. čekat na **AP<sub>m</sub>** < **p**  
   f.2 .. zatím nevyužito  
   f.3 .. pozastavit další komparátory, do události na tomto komparátoru  
   f.4 .. nastavit digitální výstupy  
   f.5 .. komparátor je spouštěn opakovaně s přírůstkem **CMPREPOc**  
   f.6 .. potlačení posílání zprávy při události  
 m .. písmeno komparovaného motoru  
 p .. komparovaná poloha ( přepočty a rozsah shodný **AP<sub>m</sub>** a **G<sub>m</sub>** )  
 do .. hodnota na digitální výstupy při události na komparátoru  
 n .. číslo skupiny výstupů  
 mask .. maska nastavovaných bitů  
 xor .. požadovaná hodnota maskovaných bitů

Při dosažení události je komparátor deaktivován a je odeslána následující řádka

CMPc!di  
 c .. číslo komparátoru 0 až 3  
 di .. co bylo na vstupech, není-li vysílání povoleno, vyšle "N"

Od verze firmware 0.6 je možné nastavit příznak pro opakované spouštění komparátoru. Pokud je příznak nastaven, dojde po dosažení nastavené polohy k odeslání řádky sériovou komunikací, nastavení výstupů, je-li požadováno, a nakonec je k přednastavené hodnotě polohy přičten offset **CMPREPOc** příslušného komparátoru a komparátor je nastaven do režimu čekání na překročení nově vypočtené komparované polohy v libovolném směru. Příznak pozastavení dalších komparátorů nastaven není.

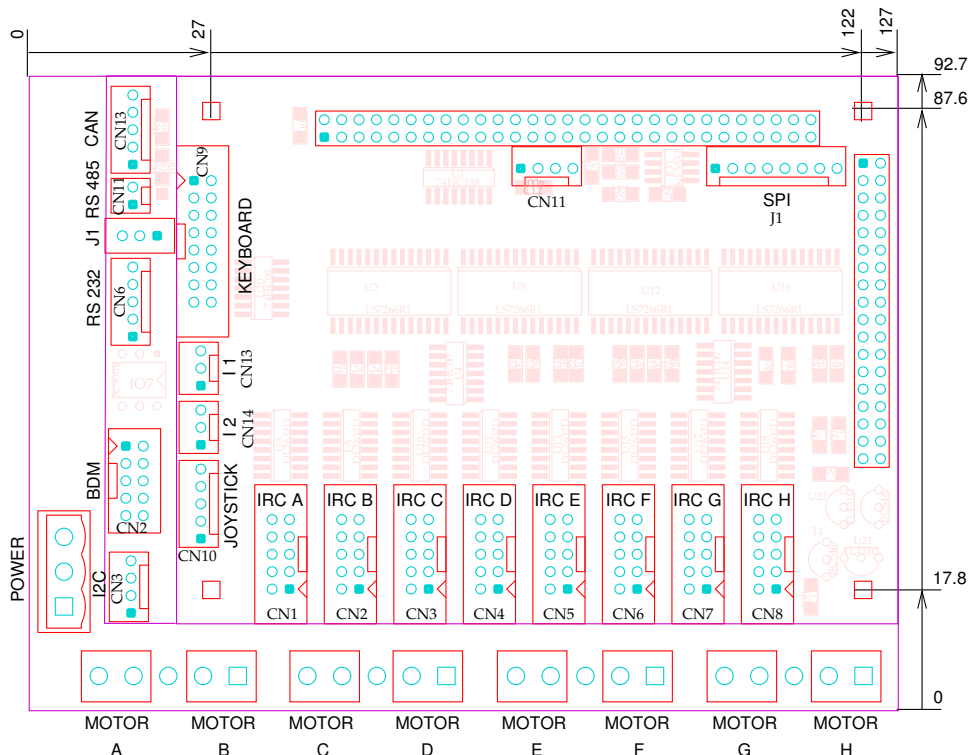
## 2.6 Rozšířené příkazy pro MARS8B

Jednotka MARS8B je z důvodů bezpečnosti vybavena vícenásobným jištěním a vypínáním výkonových výstupů. Pro zapnutí systému je potřeba nejdříve povolit zapnutí napájení.

Jméno	Op	Parametry	Funkce
REGPWRON	:	x	Povolení zapnutí napájení
		<0,1>	

Je-li povoleno zapnutí z řídicího systému, je možné stiskem žlutého tlačítka na čelním panelu jednotky zapnout výkonový zdroj. Zdroj je možné vypnout jak tlačítkem na čelním panelu, tak nastavením **REGPWRON** na hodnotu 0.

Jméno	Op	Parametry	Funkce
REGPWRFLG	:	xxx	Příznaky pro řízení výkonových výstupů
		<0,255>	



Obrázek 3: Rozmístění konektorů na jednotce **MARS8**

Bit	7	6	5	4	3	2	1	0
Příznaky						BREAK	LIM	ON

Příznak ON povoluje řízení výkonových výstupů z generátorů PWM. Příznak LIM ruší vypnutí výstupů způsobené pohybem robota mimo vymezenou pracovní oblast. Příznak by měl být nastavován jen po nezbytně nutnou dobu při automatickém návratu do pracovní oblasti. Příznak BREAK slouží k odbrzdění os vybavených elektromagnetickou brzdou.

### 3 Ovládání přes IIC rozhraní

### 4 Manuální ovládání

### 5 Konektory a připojení

#### Připojení napájení (POWER)

Řídicí jednotka **MARS 8**, sestavená v základní konfiguraci desek MO\_PWR1, MO\_CPU1 a MO\_CTR1, předpokládá přivedení vnějšího napájecího napětí 24 V na třípinový konektor typu ARKZ950/3. Pro případy, kdy je nutné umožnit samostatné vypínání zdroje pro výkonové výstupu, je přívod kladného napájecího napětí rozdělen na dvě svorky. Svorka **+24 V Control** slouží k napájení řídicí elektroniky, svorka **+24 V Motor** je přivedena na výkonové můstky motorů. Jednotka zastavená do standardní plechové skříňe je vybavena zdrojem pro napájení ze síťového napětí 230 VAC.

Pin	Signál
1	+24 V Control
2	+24 V Motor
3	Power GND

### Konektor pro připojení joysticku (JOYSTICK)

Jednotka **MARS 8** předpokládá analogový joystick se třemi potenciometry. Přivedení vodičů na špičky konektoru na řídicí jednotce je následující.

Pin	Signál	Alt. TTL vstup	DB9
1	VCC		8,9
2	VSTUP A	I13	3
3	VSTUP B	I14	4
4	VSTUP C	I15	5
5	GND		1,2

V případě zapouzdření jednotky MARS ve skřínce, je vstup joysticku vyveden na konektor Canon DB9 s dutinkami.

### Konektor komunikace RS232 (RS232)

Na desku jednotky se připojuje kabel krimpovacím konektorem. K jednotce je dodávána převodka na standardní konektor DB9 nebo DB25. V tabulce je uvedeno i propojení na DB25 pro připojení k PC.

MARS 8			PC		
Pin na desce	Signál	DB9	Signál	DB25	DB9
1	TxD	2	RxD	3	2
2	RxD	3	TxD	2	3
3	RTS	8	CTS	5	8
4	CTS	7	RTS	4	7
5	GND	5	GND	7	5

### Připojení komunikace RS485

Jednotka umožňuje alternativně využít interface asynchronní komunikace pro připojení ke sběrnici podle standardu RS-485. Propojkou **J1** je určeno, která norma a budič asynchronní komunikace je používán. Propojení špiček **1-2** nastavuje komunikaci podle standardu RS-485 a poloha **2-3** podle standardu RS-232. Dva vodiče pro komunikaci podle standardu RS-485 se připojují na konektor **RS485**.

### Konektory pro připojení motorů a IRC čidel

Výkonové výstupy pro jednotlivé motory jsou vyvedeny na dvoupinové konektory MOTOR A až H (typ ARKZ950/2). Inkrementální čidla polohy se připojují na desetipinové konektory IRC A až H (typ MLW10G / kabel PFL10). Pokud je jednotka zastavěna do plechové skříně, jsou výkonové výstupy vyvedeny na konektory MIC334 určené pro kabelové koncovky MIC324. U skříňového provedení jsou pro



vstupy IRC použity patnáctipinové delta konektory, které jsou pro připojení malých motorků vybaveny i výkonovými výstupy.

Signál	Samorezný10-pinů	Delta15-pinů	IRC HPHEDL	IRC HPHEDS
Mark Inv	1	1	-	
+5V	2	9	2	4
GND	3	2	3	1
Mark	4	10	-	
Fáze A Inv	5	3	5	
Fáze A	6	11	6	3
Fáze B Inv	7	4	7	
Fáze B	8	12	8	5
Index Inv	9	5	9	
Index	10	13	10	2
		6		
Motor +		14,7		
Motor -		15,8		

### Digitální vstupy a výstupy

Pin	Signál
Konektor I1 - CN13	
1	VCC
2	INPUT
3	GND
Konektor I2 - CN14	
1	VCC
2	INPUT
3	GND

### Synchronní komunikace SPI (SCI)

K jednotce **MARS8** lze přes synchronní sériovou komunikaci SPI připojit 3 (s dekodérem chipselectů až 7) vstupních a stejně množství výstupních registrů. Maximální bitová šířka posuvných registrů je 16 bitů. Registry mohou být samostatné obvody pro realizaci digitálních vstupů/výstupů nebo je možné připojit i složitější periferie komunikující podle standardu SPI (AD/DA převodníky, budiče segmentových displejů, atd.).

Pin	Signál
1	VCC
2	GND
3	SDI
4	SDO
5	SCLK
6	CS1
7	CS2
8	CS3

Na špičkách **VCC/GDN** je k dispozici napájení +5 V pro připojené periferie. Data synchronizovaná hodinami **SCLK** jsou přenášena po vodičích **SDI/SDO**. Signály **CSx** slouží k výběru periferie.

### Sběrnice pro řídicí systémy CAN (CAN)

Jedná se o moderní multimaster sběrnici s deterministickým přidělováním komunikačního média jednotlivým uzlům. Na jednotce **MARS8** je implementována varianta označovaná **HighSpeed** sloužící pro komunikaci do rychlosti 1 MBd.

Pin	Signál
1	CAN GND
2	CAN L
3	SHIELD
4	CAN H
5	CAN VCC

### Komunikace I<sup>2</sup>C (CN3)

Pin	Signál
1	GND
2	SCL
3	VCC
4	SDA

### Další konektory

Rozhraní **BDM** slouží k připojení jednotky k vývojovému systému umožňujícímu nahrávat a ladit firmware mikrokontroléru.

## 6 Výrobce

Elektronickou řídicí jednotku dodává firma:

PiKRON s.r.o.  
Kaňkovského 1235  
18200 PRAHA 8

Tel. Fax: (2)84684676  
Tel. : (2)96781671