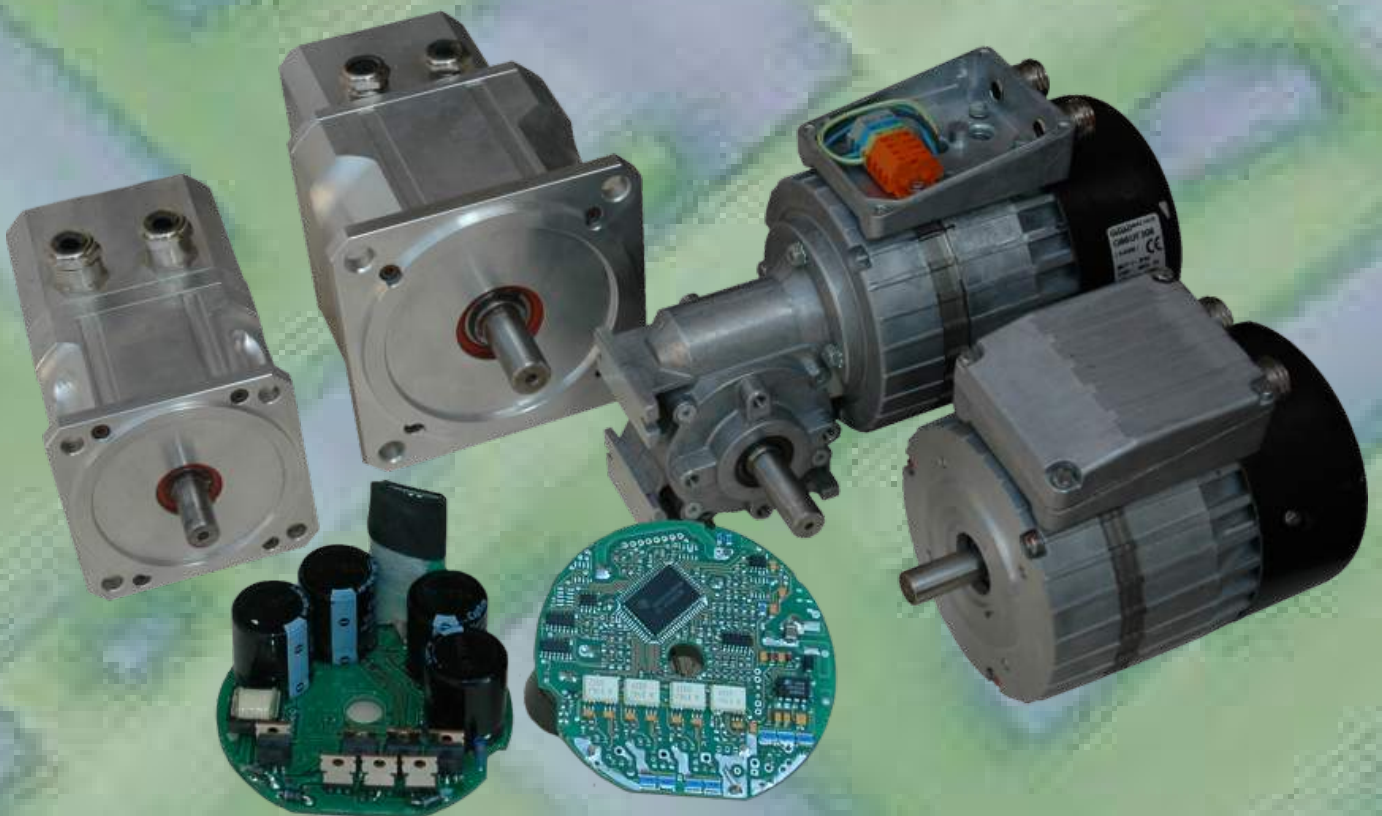


# EC Motor

## CAN – komunikační profil



## Komunikační profil CAN

### Možnosti řízení EC motoru:

- komunikace RS232/RS485
- **komunikace CAN**
- uživatelský program

### Komunikace CAN

- vychází z komunikačního protokolu **CANopen**
- zachovává definované typy komunikačních objektů **COB**
- využívá **11-ti bitový** identifikátor zprávy **COB\_ID**  
konstrukce: typ COB = 4b adresa modulu=7b ( viz.tabulka )
- vlastní definice objektů **SDO**  
index 2000h...5FFFh -> "Manufactuter Specific Profile Area"
- typy dat
  - I16** = 16bit signed
  - U16** = 16bit unsigned
  - B16** = 16bit binary
  - I32** = 32bit signed
- data jsou vysílána v pořadí byte "**low to high**"  
např. 1234h vysláno: 34h 12h  
např. 1234567h vysláno: 67h 45h 23h 01h

Komunikační objekt COB		Identifikátor zprávy COB ID			
mnemo	popis	hex	dec.	bin	směr
<b>NMT</b>	Network Management	000h	0	000 0000 0000	MA->EC
SYNC	Synchronizace	080h	128	000 1000 0000	
<b>EMCY</b>	Emergency	080h+ aa	128 + aa	000 1.aa aaaa	EC->MA
TIME	Time Stamp	100h	256 + aa	001 0000 0000	
<b>TPDO1</b>	Transmit Proces Data	180h+ aa	364 + aa	001 1.aa aaaa	EC->MA
<b>RPDO1</b>	Receive Proces Data	200h + aa	512 + aa	010 0.aa aaaa	MA->EC
TPDO2	Transmit Proces Data	280h + aa	640 + aa	010 1.aa aaaa	
RPDO2	Receive Proces Data	300h + aa	768 + aa	011 0.aa aaaa	
TPDO3	Transmit Proces Data	380h + aa	896 + aa	011 1.aa aaaa	
RPDO3	Receive Proces Data	400h + aa	1024 + aa	100 0.aa aaaa	
TPDO4	Transmit Proces Data	480h + aa	1152 + aa	100 1.aa aaaa	
RPDO4	Receive Proces Data	500h + aa	1280 + aa	101 0.aa aaaa	
<b>TSDO</b>	Transmit Servis Data	580h + aa	1408 + aa	101 1.aa aaaa	EC->MA
<b>RSDO</b>	Receive Servis Data	600h + aa	1536 + aa	110 0.aa aaaa	MA->EC
NODG	NodeGuard	700h + aa	1792 + aa	111 0.aa aaaa	

aa,aa aaaa adresa node motoru v **c\_set.adr0** ( object 200Dh)  
**MA** master PLC, PC s interface CAN, ...  
**EC** EC motor

### Nastavení EC motoru pro komunikaci CAN

- jen pro interface **EC113** ( viz manual... )  
a verzi software **ECA19** a vyšší
- připojit motor k PC pomocí RS232 a zapnout motor
- spustit program "EC motor Setings" s adresou motoru = 0
  - v menu SETUP, záložka .. , c\_set (viz. object 200Dh)  
nastavit **adresu motoru** 1-15  
nastavit **komunikační rychlost CAN** 1-7  
uložit do EEPROM ( Ctrl Enter)
- provést HW reset motoru ( vypnout na 30s nebo mod=RESET )



- programu "EC motor Setings" při nastavení adresy=0 komunikuje s motory bez ohledu na nastavení adresy v c\_set
- pro správnou funkci sběrnice CAN je třeba připojit mezi vodiče CANH a CANL zakončovací odpor 120R ( např. do svorkovnice)

**Implementované komunikační objekty:****NMT - Network Management**

NMT - reset motoru		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	proved' reset	NMT	000h	2	82h	0xh						
EC:	bez odpovědi											

**82h** kod reset U8 kod CANopen  
**0xh** adresa motoru v **c\_set.adr0** U8 ( object 200Dh)

**EMCY - Emergency**

- vypnutí SW/HW ochrany motoru
- odmítnutý příkaz komunikace
- nastavením bitu **c\_sss.bcse** ( object 200Ch)
- příkaz uživatelského programu UPRG

EMCY hlášení po chybě		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
EC:	chybové hlášení	EMCY	08xh	8	10h	00h	m_sss		c_sss		m_ssm	

**0010h** nespecifikovana chyba U16 kod CANopen  
**m\_sss** chybové stavy motoru B16 ( object 200Eh)  
**c\_sss** chybové stavy komunikace B16 ( object 200Ch)  
**m\_ssm** základní stav motoru B16 ( object 200Fh)

**TPDO1 Proces Data Object**

- odpověď na RPDO1 - žádost o stav z MA
- nastavením bitu **c\_sss.bcscs** ( object 200Ch)
- příkaz uživatelského programu UPRG

PDO1 stav motoru		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli stav	RPDO1	20xh	0								
EC:	posílám stav	TPDO1	18xh	8			h_poz			m_eci		m_ssm

**h\_poz** skutečná poloha ze snímače I32 ( object 2012h)  
**m\_eci** stavy vstupů a výstupů B16 ( object 200Ah)  
**m\_ssm** základní stavy motoru B16 ( object 200Fh)

**SDO Service Data Object**

SDO index		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	povel / dotaz	RSDO	60xh	4..8	cs		index	subi.				data
EC:	potvrzení / odpověď	TSDO	58xh	4..8	cs		index	subi.				data

**ID** identifikátor zprávy  
**x** adresa motoru 1-15 ( 01h-0Fh) v **c\_set.adr0** ( object 202Fh)  
**DLC** délka zprávy 0..8  
 EC příjem : není kontrolována ( zkrátit dle typu dat nebo doplnit 00h na DLC=8 )  
 EC vysílání: zkrácena dle typu dat  
**cs** specifikace typu zprávy "command specifier"  
 použít pouze zkrácený typ přenosu "expedited transfer" => max 4byte dat  
 zápis **w** MA: **22h** "zápis" EC: **60h** "potvrzení"  
 čtení **r** MA: **40h** "pošli" EC: **40h** "posílám"  
**index** index objektu  
 2000h ... 203Fh **objekty motoru a uživatele**  
 2040h ... 2045h **speciální funkce**  
**subi.** subindex objektu  
 modifikuje přístup k položkám objektu ( délka, funkce..)  
**data** přenášená data v pořadí byte "low to high"

**SDO 2000h-203Fh Proměnné motoru a uživatele**

CAN		UPRG	bitově přístupné objekty a stavová slova				
index	subi.	adresa	type	r/w	class	mnemo	popis
2000h	x	00h/00h	x		x	URAM12	user memory
2001h	x	01h/04h	x	x	x	URAM13	user memory
2002h	x	02h/08h	x	x	x	URAM14	user memory
2003h	x	03h/0Ch	x	x	x	URAM15	user memory
2004h	x	04h/10h	x	x	x	URAM16	user memory
2005h	x	05h/14h	x	x	x	URAM17	user memory
2006h	x	06h/18h	x	x	x	URAM18	user memory
2007h	x	07h/1Ch	x	x	x	URAM19	user memory
2008h	x	08h/20h	x	x	x	URAM20	user memory
2009h	x	09h/24h	-	-	-	----	
200Ah	0h	0Ah/28h	B16	rw	-	m_eci	stavy vstupů a výstupů
200Bh	0h	0Bh/2Ch	B16	rw	-	S_err	maska SW ochrany
200Ch	0h	0Ch/30h	B16	r	-	c_sss	stavové slovo komunikace
200Dh	0h	0Dh/34h	B16	rw	-	m_set	řídící slovo motoru
200Eh	0h	0Eh/38h	B16	r	-	m_sss	chybové slovo motoru
200Fh	0h	0Fh/3Ch	B16	r	-	m_ssm	stavové slovo motoru

CAN		UPRG	proměnné motoru - jen pro čtení				
index	subi.	adresa	type	r/w	class	mnemo	popis
2010h	1h	10h/40h	I32	r	P	p_pox	okamžitá pozice regulace
2011h	0h	11h/44h	I16	r	V	p_vex	okamžitá rychlost regulace
2012h	1h	12h/48h	I32	r	P	h_poz	skutečná pozice
2013h	1h	13h/4Ch	I32	r	P	h_tet	absolutní poloha pro vektorové řízení
2014h	0h	14h/50h	I16	r	V	h_vel	skutečná rychlost
2015h	0h	15h/54h	I16	r	M	iq_ref	změřený proud -> moment
2016h	0h	16h/58h	U16	r	V	udc	napětí stejnosměrného meziobvodu
2017h	0h	17h/5Ch	U16	r	T	tsn	teplota snímače
2018h	0h	18h/60h	U16	r	T	tcu	tepelné využití motoru
2019h	0h	19h/64h	I16	r	P	p_err	okamžitá chyba polohy
201Ah	0h	1Ah/68h	-	-	-	----	
201Bh	0h	1Bh/6Ch	-	-	-	----	
201Ch	0h	1Ch/70h	-	-	-	----	
201Dh	0h	1Dh/74h	-	-	-	----	
201Eh	0h	1Eh/78h	-	-	-	----	
201Fh	0h	1Fh/7Ch	-	-	-	----	

CAN		UPRG	proměnné motoru - zápis				
index	subi.	adresa	type	r/w	class	mnemo	popis
2020h	1h	20h/80h	U32	rw	-	u_tim	systemový čas
2021h	1h	21h/84h	I32	rw	P	P_lil	limit polohy spodní
2022h	1h	22h/88h	I32	rw	P	P_lih	limit polohy horní
2023h	1h	23h/8Ch	I32	rw	P	p_pon	požadovaná pozice
2024h	0h	24h/90h	I16	rw	V	p_ven	požadovaná rychlost
2025h	0h	25h/94h	U16	rw	A	p_acc	nastvené zrychlení
2026h	0h	26h/98h	U16	rw	D	p_dec	nastvené zpomalení
2027h	0h	27h/9Ch	I16	rw	M	p_mon	omezení momentu
2028h	0h	28h/A0h	-	-	-	----	
2029h	0h	29h/A4h	-	-	-	----	
202Ah	0h	2Ah/A8h	-	-	-	----	
202Bh	0h	2Bh/ACh	-	-	-	----	
202Ch	0h	2Ch/B0h	-	-	-	----	
202Dh	0h	2Dh/B4h	-	-	-	----	
202Eh	0h	2Eh/B8h	-	-	-	----	
202Fh	0h	2Fh/BCh	U16	rw	-	c_set	definice komunikace

CAN		UPRG	inicializované proměnné UPRG				
index	subi.	adresa	type	r/w	class	mnemo	popis
2030h	x	30h/C0h	x	x	x	URAM00	user memory
2031h	x	31h/C4h	x	x	x	URAM01	user memory
2032h	x	32h/C8h	x	x	x	URAM02	user memory
2033h	x	33h/CCh	x	x	x	URAM03	user memory
2034h	x	34h/D0h	x	x	x	URAM04	user memory
2035h	x	35h/D4h	x	x	x	URAM05	user memory
2036h	x	36h/D8h	x	x	x	URAM06	user memory
2037h	x	37h/DCh	x	x	x	URAM07	user memory
2038h	x	38h/E0h	x	x	x	URAM08	user memory
2039h	x	39h/E4h	x	x	x	URAM09	user memory
203Ah	x	3Ah/E8h	x	x	x	URAM10	user memory
203Bh	x	3Bh/ECh	x	x	x	URAM11	user memory
203Ch	x	3Ch/F0h	-	-	-	----	
203Dh	x	3Dh/F4h	-	-	-	----	
203Eh	x	3Eh/F8h	-	-	-	----	
203Fh	x	3Fh/FCh	-	-	-	----	

---- neobsazeny index - nelze zapisovat a číst -> chyba EMCY  
**URAMxx** pole proměnných pro uživatelský program  
**x** využití definované v uživatelském programu  
**type** formát proměnné (u16, i16, b16, i32)  
**r/w** typ přístupu w – zápis, r – čtení, rw – zápis i čtení  
**class** třída proměnné pro přepočítání na fyz. veličiny, viz tabulka  
 (rychlost, poloha, zrcyhlení, proud, moment, napětí)

Class	Name	Conversion const.	Unit	Minimum	Maximum
<b>V</b>	Velocity	0.1875	RPM	-32768	32767
<b>A</b>	Acceleration	0.0146484375	RPM/ms	0	4096
<b>D</b>	Deceleration	0.0146484375	RPM/ms	0	4096
<b>P</b>	Position	0.1171875	DEG	-99999999	99999999
<b>M</b>	Torque	0.000135	Nm	0	16384
<b>C</b>	Current	0.000135	A	-50000	50000
<b>V</b>	Voltage	0.01	V	-50000	50000
<b>T</b>	Temperature	0.01	°C	-20000	20000
<b>K</b>	Time	0.8	ms	0	32768
--	machine unit	1	-	0	65535

SDO zápis I16,U16,B16		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	zapiš	RSDO	60xh	6/8	<b>22h</b>	ii	20h	<b>00h</b>	d0	d1	0	0
EC:	potvrzeno	TSDO	58xh	4	<b>60h</b>	ii	20h	00h	--	--	--	--

SDO zápis I32		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	zapiš	RSDO	60xh	6/8	<b>22h</b>	ii	20h	<b>01h</b>	d0	d1	d2	d3
EC:	potvrzeno	TSDO	58xh	4	<b>60h</b>	ii	20h	01h	--	--	--	--

SDO pošli I16,U16,B16		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli	RSDO	60xh	4/8	<b>40h</b>	ii	20h	<b>00h</b>	0	0	0	0
EC:	posílám	TSDO	58xh	6	<b>40h</b>	ii	20h	00h	d0	d1	--	--

SDO pošli I32		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli	RSDO	60xh	4/8	<b>40h</b>	ii	20h	<b>01h</b>	0	0	0	0
EC:	posílám	TSDO	58xh	8	<b>40h</b>	ii	20h	01h	d0	d1	d2	d3

**ii** nižší byte indexu (index = 2000h+ ii)  
**--** data nejsou vysílána  
**0** data nejsou vysílána nebo možno doplnit hodnotou 00h na DLC=8



Příklad 1: Zadej požadovanou rychlost p\_ven = 1234h  
 adresa motoru c\_set.adr0=03h ( **použito ve všech dalších příkladech !!!** )  
 požadovaná rychlost p\_ven : index= 2024h subí.= 00h

SDO zapiš I16,U16,B16		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	p_ven=1234h	RSDO	603h	8	22h	24h	20h	00h	34h	12h	00h	00h
EC:	potvrzeno	TSDO	583h	4	60h	24h	20h	00h				



Příklad 2: Zadej požadovanou polohu p\_pon = 1234567h  
požadovaná poloha p\_pon : index= 2023h subi.= 01h

SDO zapiš I32		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	p_pon = 1234567h	RSDO	603h	8	22h	23h	20h	01h	67h	45h	23h	01h
EC:	potvrzeno	TSDO	583h	4	60h	23h	20h	01h				



Příklad 3: Načti požadovanou rychlost p\_ven  
požadovaná rychlost p\_ven : index= 2024h subi.= 00h ( dříve zapsáno p\_ven = 1234h )

SDO pošli I16,U16,B16		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli p_ven	RSDO	603h	8	40h	24h	20h	00h	00h	00h	00h	00h
EC:	posílám p_ven	TSDO	583h	6	40h	24h	20h	00h	34h	12h		



Příklad 4: Načti požadovanou polohu p\_pon  
požadovaná poloha p\_pon : index= 2023h subi.= 01h ( dříve zapsáno p\_pon = 1234567h )

SDO pošli I32		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli p_pon	RSDO	603h	4	40h	23h	20h	01h				
EC:	posílám p_pon	TSDO	583h	8	40h	23h	20h	01h	67h	45h	23h	01h

**SDO 2000h-200Fh Bitově orientované objekty**

200Dh		m_set		řídící bity motoru													
číslo bitu		--	cgr	cbi	cdi	funkce ECI200				--	--	--	--	cen	--	clp	cdn
dec.	hex.	maska		mnemo	popis												
15	0Fh	8000h															
14	0Eh	4000h		car	dec=acc pro generator profilu GR MAX												
13	0Dh	2000h		cbi	blokovani BIO v potenciometru ...												
12	0Ch	1000h		cdi	kladný směr otáčení												
11	0Bh	0800h			ECI200 definice funkce 0-15												
10	0Ah	0400h															
09	09h	0200h															
08	08h	0100h															
07	07h	0080h															
06	06h	0040h															
05	05h	0020h															
04	04h	0010h															
03	03h	0008h		cen	aktivace enable -				1=BIO vvpíná vvkonovou část								
02	02h	0004h															
01	01h	0002h		clp	aktivace limitu polohv												
00	00h	0001h		cdn	po dosažení pozice do modu 03												

200Fh		m_ssm			stavové slovo motoru								
číslo bitu		ber	bwa	bx	bio	bpz	bru	bpo	bup	bci	bco	bex	mod motoru
dec.	hex.	maska		mnemo	popis								
15	0Fh	8000h		ber	celkový stav chyba ( HW.SW.Reset)								
14	0Eh	4000h		bwa	stav warning viz. <b>m sss</b>								
13	0Dh	2000h		bx	uživatelský flag v uživatelském programu								
12	0Ch	1000h		bio	indikace vstupu BIO								
11	0Bh	0800h		bpz	výkonová část zapnuta								
10	0Ah	0400h		bru	motor se točí								
09	09h	0200h		bpo	motor v dané pozici								
08	08h	0100h		bup	běží uživatelský program								
07	07h	0080h		bci	bit CAN Rx priem (pri CAN=IO)								
06	06h	0040h		bco	bit CAN Tx vyslani (pri CAN=IO)								
05	05h	0020h		bex	externí start osciloskopu								
04	04h	0010h		mod	pracovní mod motoru - viz tab <b>T_MOD</b>								
03	03h	0008h											
02	02h	0004h											
01	01h	0002h											
00	00h	0001h											

200Eh		m_sss			chybové stavy motoru												
číslo bitu		bid	brs	bhw	bsw	buh	bul	bih	bep	bts	btm	btb	bek	ben	bee	bll	blh
dec.	hex.	maska		mnemo	popis												
15	0Fh	8000h		bid	chvba integrity dat ve FLASH a EEPROM												
14	0Eh	4000h		brs	proběhl RESET -> nutná nova inicializace												
13	0Dh	2000h		bhw	vvpła HW ochrana												
12	0Ch	1000h		bsw	vvpła SW ochrana												
11	0Bh	0800h		buh	max napájecí napětí												
10	0Ah	0400h		bul	min napájecí napětí												
09	09h	0200h		bih	max. moment / proud Iq												
08	08h	0100h		bep	max. chvba regulace polohv												
07	07h	0080h		bts	max. teplota snímače( změřeno)												
06	06h	0040h		btm	max. vvužití motoru ( teplotní model)												
05	05h	0020h		btb	max. vvužití brzdvy ( teplotní model)												
04	04h	0010h		bek	chvba komunikace z c sss												
03	03h	0008h		ben	disable ( BIO=0 povoluie <b>m set.cen</b> )												
02	02h	0004h		bee	chvba ECI modulu												
01	01h	0002h		bll	min .limit polohv												
00	00h	0001h		blh	max. limit polohv												



- bity jsou nastaveny pro zjištění chyby
- vynulovat je nutno příkazem (např. SDOindex=2045h subi.=0)

202Fh		c_set		řídící slovo komunikace							
číslo bitu		mapm		adr1		cfk	cfc		adr0		
dec.	hex.	maska	mnemo	popis							
15	0Fh	8000h	mapm	maska příkazu monitoru 0-15							
14	0Eh	4000h	adr1	adresa skupiny CAN 1-15							
13	0Dh	2000h									
12	0Ch	1000h									
11	0Bh	0800h									
10	0Ah	0400h									
09	09h	0200h	cfc	rychlost RS232: 0 = 9,6kBd 1 = 38kBd							
08	08h	0100h		ccc	rychlost CAN : 000 = CAN_OFF -> IO 001 = 10kBd 010 = 20kBd 011 = 50kBd 100 = 125kBd 101 = 250kBd 110 = 500kBd 111 = 1MBd						
07	07h	0080h	adr0		adresa motoru CAN / RS232 1-15						
06	06h	0040h									
05	05h	0020h									
04	04h	0010h									
03	03h	0008h	adr0	adresa motoru CAN / RS232 1-15							
02	02h	0004h									
01	01h	0002h									
00	00h	0001h									

200Ch		c_sss		stavy komunikace													
číslo bitu		bcer	bcef							bcse	bcss			bcok	bcep		
dec.	hex.	maska	mnemo	popis													
15	0Fh	8000h	bcer	CAN chyba formátu komunikace CANESR.FER													
14	0Eh	4000h	bcef	CAN nesprávný příkaz													
13	0Dh	2000h															
12	0Ch	1000h															
11	0Bh	0800h															
10	0Ah	0400h															
09	09h	0200h	bcse	CAN pošli zprávu "chvba"													
08	08h	0100h	bcss	CAN pošli zprávu "stav "													
07	07h	0080h															
06	06h	0040h															
05	05h	0020h	bcok	RS232 přijmut platný paket													
04	04h	0010h	bcep	RS232 nesprávný příkaz													
03	03h	0008h															
02	02h	0004h															
01	01h	0002h															
00	00h	0001h															

200Ah		m_eci		stavové slovo vstupů a výstupů s ECI200- návrh nedoděláno !!!											
číslo bitu															
dec.	hex.	maska	mnemo	popis											
15	0Fh	8000h	brfc	přečti funkci											
14	0Eh	4000h													
13	0Dh	2000h													
12	0Ch	1000h													
11	0Bh	0800h													
10	0Ah	0400h													
09	09h	0200h	in9	vstup 9											
08	08h	0100h	in8	vstup 8											
07	07h	0080h	in7	vstup 7											
06	06h	0040h	io6	vstup / výstup 6											
05	05h	0020h	io5	vstup / výstup 5											
04	04h	0010h	io4	vstup / výstup 4											
03	03h	0008h	io3	vstup / výstup 3											
02	02h	0004h	ou2	výstup 2											
01	01h	0002h	ou1	výstup 1											
00	00h	0001h	ou0	výstup 0											



### Bitové operace

Pro bitově orientované objekty typu B16 (2000h-200Fh) je možno kromě SDO pro čtení a zápis U16 využít logické funkce :

subi.= 02h	<b>SET</b>	nastav bity dle masky
subi.= 03h	<b>CLR</b>	nuluj bity dle masky
subi.= 04h	<b>NEG</b>	neguj bity dle masky



SDO B16 nastav bity		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	nastav bity	RSDO	60xh	6/8	<b>22h</b>	ii	20h	<b>02h</b>	<b>b0</b>	<b>b1</b>	0	0
EC:	potvrzeno	TSDO	58xh	4	<b>60h</b>	ii	20h	02h	--	--	--	--

SDO B16 nuluj bity		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	nuluj bity	RSDO	60xh	6/8	<b>22h</b>	ii	20h	<b>03h</b>	b0	b1	0	00h
EC:	potvrzeno	TSDO	58xh	4	<b>60h</b>	ii	20h	03h	--	--	--	

SDO B16 neguj bity		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	neguj bity	RSDO	60xh	6/8	<b>22h</b>	ii	20h	<b>04h</b>	b0	b1	0	00h
EC:	potvrzeno	TSDO	58xh	4	<b>60h</b>	ii	20h	04h	--	--	--	--

**b1b0** maska bitů viz . SDO 2000h-2000Fh bitově orientované objekty



Příklad 5: Nastavení, nulování a negace bitu **m\_ssm.bxx** ( uživatelský flag )

stavové slovo motoru **m\_ssm** : index =200Fh ,  
z popisu 200Fh m\_ssm bit **bxx** : maska = 2000h  
subindex dle funkce : subl. = 02h-> SET 03h->CLR 04h->NEG

SDO B16 nastav bity		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	nastav bit m_ssm.bxx	RSDO	603h	8	22h	0Fh	20h	02h	00h	20h	00h	00h
EC:	potvrzeno	TSDO	583h	4	60h	0Fh	20h	02h				

SDO B16 nuluj bity		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	nuluj bit m_ssm.bxx	RSDO	603h	8	22h	0Fh	20h	03h	00h	20h	00h	00h
EC:	potvrzeno	TSDO	583h	4	60h	0Fh	20h	03h				

SDO B16 neguj bity		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	neguj bit m_ssm.bxx	RSDO	603h	8	22h	0Fh	20h	04h	00h	20h	00h	00h
EC:	potvrzeno	TSDO	583h	4	60h	0Fh	20h	04h				

## SDO 2040h ..2045h - Funkce řízení motoru

CAN		parametr			funkce řízení motoru definované pro CAN	
index	subi	data	type	r/w	mnemo	popis
2040h	0h	<b>a1a0</b>	U16	r		čtení dat U16 z fyzické adresy <b>a1a0</b>
2040h	1h	<b>a1a0</b>	U16	r		čtení dat U32 z fyzické adresy <b>a1a0</b>
2041h	0h	----	--	r		načti číslo verze SW ( type U16)
2042h	0h	<b>m0</b>	U16	w		nastav mod motoru dle tabulky T_MOD
2043h	0h	0000h	U16	w		zastav uživatelský program 1
2043h	0h	<b>u1u0</b>	U16	w		spusť uživatelský program 1 od adresy <b>u1u0</b>
2044h	0h	0000h	U16	w		zastav uživatelský program 2
2044h	0h	<b>u1u0</b>	U16	w		spusť uživatelský program 2 od adresy <b>u1u0</b>
2045h	<b>0h</b>	----	--	w		nuluj chybové stavy ( warning <b>m_sss</b> , <b>m_ssm</b> , <b>c_sss</b> )
2045h	<b>1h</b>	----	--	w		nuluj pozici když stojí
2045h	<b>2h</b>	----	--	w		nuluj pozici za běhu ( jen v mod=05 rychlostní !!!)
2045h	<b>3h</b>	----	--	w		nuluj pozici absolutně+ ( dle absolutní polohy snímače )
2045h	<b>4h</b>	----	--	w		nuluj pozici absolutně- ( dle absolutní polohy snímače )
2046h	x		x	x	----	nevyužito
2047h	x		x	x	----	nevyužito

## 2040h Čtení z fyzické adresy

- jen pro ladění

SDO pošli U16 z adresy		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli	RSDO	60xh	6/8	<b>40h</b>	<b>40h</b>	20h	00h	<b>a0</b>	<b>a1</b>	0	00h
EC:	posílám	TSDO	58xh	6	<b>40h</b>	40h	20h	00h	<b>d0</b>	<b>d1</b>	--	--

SDO pošli U32 z adresy		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli	RSDO	60xh	6/8	<b>40h</b>	<b>40h</b>	20h	<b>01h</b>	a0	a1	0	00h
EC:	posílám	TSDO	58xh	8	<b>40h</b>	40h	20h	01h	<b>d0</b>	<b>d1</b>	<b>d2</b>	<b>d3</b>

**a1a0** - fyzická adresa U16  
**d1d0** - data U16  
**d3d2d1d0** - data U32



Příklad 6: Čtení z fyzické adresy - jen pro ladění - uživatel nezná fyzické adresy  
 index: 2040h sub.: 00h předpoklad: p\_ven : a1a0 = 367h p\_ven = 1234h  
 index: 2040h sub.: 01h předpoklad: p\_pon : a1a0 = 365h- p\_pon = 1234567h

SDO pošli U16 z adresy		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli (367h)	RSDO	603h	8	40h	40h	20h	00h	67h	03h	00h	00h
EC:	posílám (367h)	TSDO	583h	6	40h	40h	20h	00h	34h	12h		

SDO pošli U32 z adresy		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli (365h)	RSDO	603h	8	40h	40h	20h	01h	65h	03h	00h	00h
EC:	posílám (365h)	TSDO	583h	8	40h	40h	20h	01h	67h	45h	23h	01h

## 2041h Načtení čísla verze firmware

SDO pošli verzi SW		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli	RSDO	60xh	4/8	<b>40h</b>	<b>41h</b>	20h	00h	0	0	0	0
EC:	posílám	TSDO	58xh	6	<b>40h</b>	41h	20h	00h	<b>v0</b>	<b>v1</b>	--	--

**v1v0** - verze firmware



Příklad 7: Načti číslo verzi firmware  
 index: 2041h sub.: 00h  
 načteno : **00BEh** -> 00h = ECA BEh = 190<sub>10</sub> => SW = **ECA19.0**

SDO pošli verzi SW		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	pošli verzi	RSDO	603h	4	40h	41h	20h	00h				
EC:	posílám verzi	TSDO	583h	6	40h	41h	20h	00h	<b>BEh</b>	<b>00h</b>		

## 2042h Nastav pracovní mod motoru

SDO pracovní mod		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	nastav mod	RSDO	60xh	6/8	<b>22h</b>	42h	20h	00h	m0	00h	0	0
EC:	potvrzeno	TSDO	58xh	4	<b>60h</b>	42h	20h	00h	--	--	--	--

**m0** - pracovní mod motoru - viz tabulka **T\_MOD**

T_MOD		tabulka pracovních módů motoru										
mod		základní pracovní mody				mod		zákaznické pracovní mody				
dec.	hex.	mnemo	popis			dec.	hex.	mnemo	popis			
0	00h	MON	servisní mod			16	00h	ARRP	ARROW rychlostní profil			
1	01h	OFF	vypnutí výkon.části			17	01h	ETY1	Mavet stará etyketovačka			
2	02h	STP	nouzové zastavení			18	02h	ETY2	Mavet nová etyketovačka			
3	03h	PON	zapnutí výkonové části			19	03h	VYSR	Vyškov regulace SX_VYS			
4	04h	MOM	momentová regulace			20	04h	VYSL	Vyškov ladění SX_VYL			
5	05h	VEL	rychlostní regulace			21	05h					
6	06h	POZ	polohová regulace			22	06h					
7	07h					23	07h					
8	08h					24	08h					
9	09h					25	09h					
10	0Ah					26	0Ah					
11	0Bh					27	0Bh					
12	0Ch					28	0Ch	M24	starý monitor			
13	0Dh					29	0Dh	ERR	stav chyba			
14	0Eh					30	0Eh	JMP	ladění			
15	0Fh	VELP	rychlost z potenciometru			31	0Fh	RES	HW reset motoru			



- navolený mod možno načíst z **m\_ssm.mod** ( object 200Fh)
- vypnutím můstku mod=1 se ruší stav chyba mod=29 pokud je vše v pořádku možno pokračovat libovolným modem
- zapnutí můstku nelze provést dokud se motor točí
- mody 28-31 jsou interní pracovní mody - nevyužívat
- zákaznické mody 16-20 nezapínat bez odpovídajících datových struktur v URAM !!!
- mod 2 přejde po zastavení automaticky do modu 3 ( stojí s momentem )



Příklad 8: Přepni na rychlostní pracovní mod motoru  
index: 2042h sub.: 00h  
rychlostní mod = 05h

SDO pracovní mod		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	nastav mod = rychlost	RSDO	603h	6	22h	42h	20h	00h	<b>05h</b>	00h		
EC:	potvrzeno	TSDO	583h	4	60h	42h	20h	00h				

## 2043h Ovládání uživatelského programu 1

SDO user program 1		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	spust' program	RSDO	60xh	6/8	22h	<b>43h</b>	20h	00h	<b>u0</b>	<b>u1</b>	0	0
EC:	potvrzeno	TSDO	58xh	4	60h	<b>43h</b>	20h	00h	--	--	--	--

**u1u0** - adresa sekce uživatelského programu definovaná při překladu

SDO user program 1		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	zastav program	RSDO	60xh	6/8	22h	43h	20h	00h	<b>00h</b>	<b>00h</b>	0	0
EC:	potvrzeno	TSDO	58xh	4	60h	43h	20h	00h	--	--	--	--

## 2044h Ovládání uživatelského programu 2

SDO user program 2		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	spust' program	RSDO	60xh	6/8	22h	<b>44h</b>	20h	00h	<b>u0</b>	<b>u1</b>	0	0
EC:	potvrzeno	TSDO	58xh	4	60h	<b>44h</b>	20h	00h	--	--	--	--

SDO user program 2		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	zastav program	RSDO	60xh	6/8	22h	44h	20h	00h	<b>00h</b>	<b>00h</b>	0	0
EC:	potvrzeno	TSDO	58xh	4	60h	44h	20h	00h	--	--	--	--



Příklad 9: Zastav UPRG2 a spusť UPRG1 od adresy 0FF42h ( adresa generována při překladi UPRG )

UPRG2 index: 2044h subi.: 00h data: 0000h -> stop

UPRG1 index: 2043h subi.: 00h data: FF42h -> start

SDO user program 2		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	zastav UPRG2	RSDO	603h	8	22h	44h	20h	00h	00h	00h	00h	00h
EC:	potvrzeno	TSDO	583h	4	60h	44h	20h	00h				

SDO user program 1		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	spusť UPRG1	RSDO	603h	8	22h	43h	20h	00h	42h	FFh	00h	00h
EC:	potvrzeno	TSDO	583h	4	60h	43h	20h	00h				

## 2045h Funkce nulování

SDO nulování		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	nuluj	RSDO	60xh	4/8	22h	<b>45h</b>	20h	<b>ff</b>	0	0	0	0
EC:	potvrzeno	TSDO	58xh	4	60h	<b>45h</b>	20h	ff				

**ff** - funkce (subindex) nulování dle T\_CAN



Příklad 10: Nuluj pozici - motor stojí  
index 2045h subi.: 01h

SDO nulování		COB	ID	DLC	data1	data2	data3	data4	data5	data6	data7	data8
MA:	nuluj pozici	RSDO	603h	4	22h	45h	20h	01h				
EC:	potvrzeno	TSDO	583h	4	60h	45h	20h	01h				