

Fakulta elektrotechnická Regionální inovační centrum elektrotechniky

Pohon ramene robotu: uživatelský popis

Pracoviště: Výkonová elektronika a pohony
Číslo dokumentu: 22190-041-2017
Typ zprávy: Výzkumná zpráva
Řešitelé: doc. Ing. Václav Šmídl, Ph.D. et Ph.D.,
Ing. Tomáš Glasberger, Ph.D.,
Ing. Jan Molnár, Ph.D.,
Ing. Tomáš Košan, Ph.D.,
Ing. Josef Štengl,
Ing. Zdeněk Kehl.
Hlavní řešitel: prof. Ing. Zdeněk Peroutka, Ph.D.
Počet stran: 27
Datum vydání: 14. 8. 2017
Oborové zařazení: JA – Elektronika a optoelektronika, elektrotechnika

Zadavatel / zákazník:
Západočeská univerzita v Plzni
Výzkumné centrum - Nové technologie pro
informační společnost
Technická 8
301 00 Plzeň

Zpracovatel / dodavatel:
Západočeská univerzita v Plzni
Regionální inovační centrum elektrotechniky
Univerzitní 8
306 14 Plzeň

Kontaktní osoba:

Ing. Zdeněk Kehl
tel. 377 634 140
kehlz@rice.zcu.cz

Tato práce vznikla v rámci projektu č. TE02000103 „Centrum inteligentních pohonů a pokročilého řízení strojů“ s finanční podporou TA ČR.

Anotace

Tato výzkumná zpráva se zabývá uživatelským popisem hardwarové a softwarové konstrukce měniče kmitočtu pro pohon ramene robotu.

Klíčová slova

Frekvenční měnič, synchronní motor s permanentními magnety, vektorové řízení, CAN.

Název zprávy v anglickém jazyce / Report title

Robot arm drive: User description

Anotace v anglickém jazyce / Abstract

This research report describes the hardware and software design of the frequency converter for the robot arm drive.

Klíčová slova v anglickém jazyce / Keywords

Frequency converter, permanent magnet synchronous motor, vector control, CAN.

Seznam symbolů a zkratk

REMCS	RICE Embedded Modular Control System
DIF	Direct interface
PWM	Pulsně šířková modulace
GUI	Graphical User Interface
PMSM	Synchronní motor s permanentními magnety
Sync	Synchronizační signál
PI	Proporcionálně integrační regulátor
LPF	Dolnoproustný filtr
arctg	Funkce Arkus tangens
P_p	Počet pólpárů
NC	Nezpojeno

Obsah

1	ÚVOD	5
2	ZÁKLADNÍ POPIS HARDWARU	6
2.1	MĚNIČ KMITOČTU	6
2.1.1	D/A převodník.....	7
2.2	SERVOMOTOR	9
2.3	NADŘAZENÉ ŘÍZENÍ	12
3	POPIS SOFTWARE.....	12
3.1	ŘÍDICÍ SOFTWARE POHONU	12
3.2	NADŘAZENÉ ŘÍZENÍ	15
3.2.1	RxControl 0x10	17
3.2.2	TxConverterState 0x11	19
3.2.3	RxRegulatorConstant 0x12	22
3.2.4	TxRegulatorConstant 0x13	23
4	MĚŘENÍ.....	25
5	ZÁVĚR	25

1 Úvod

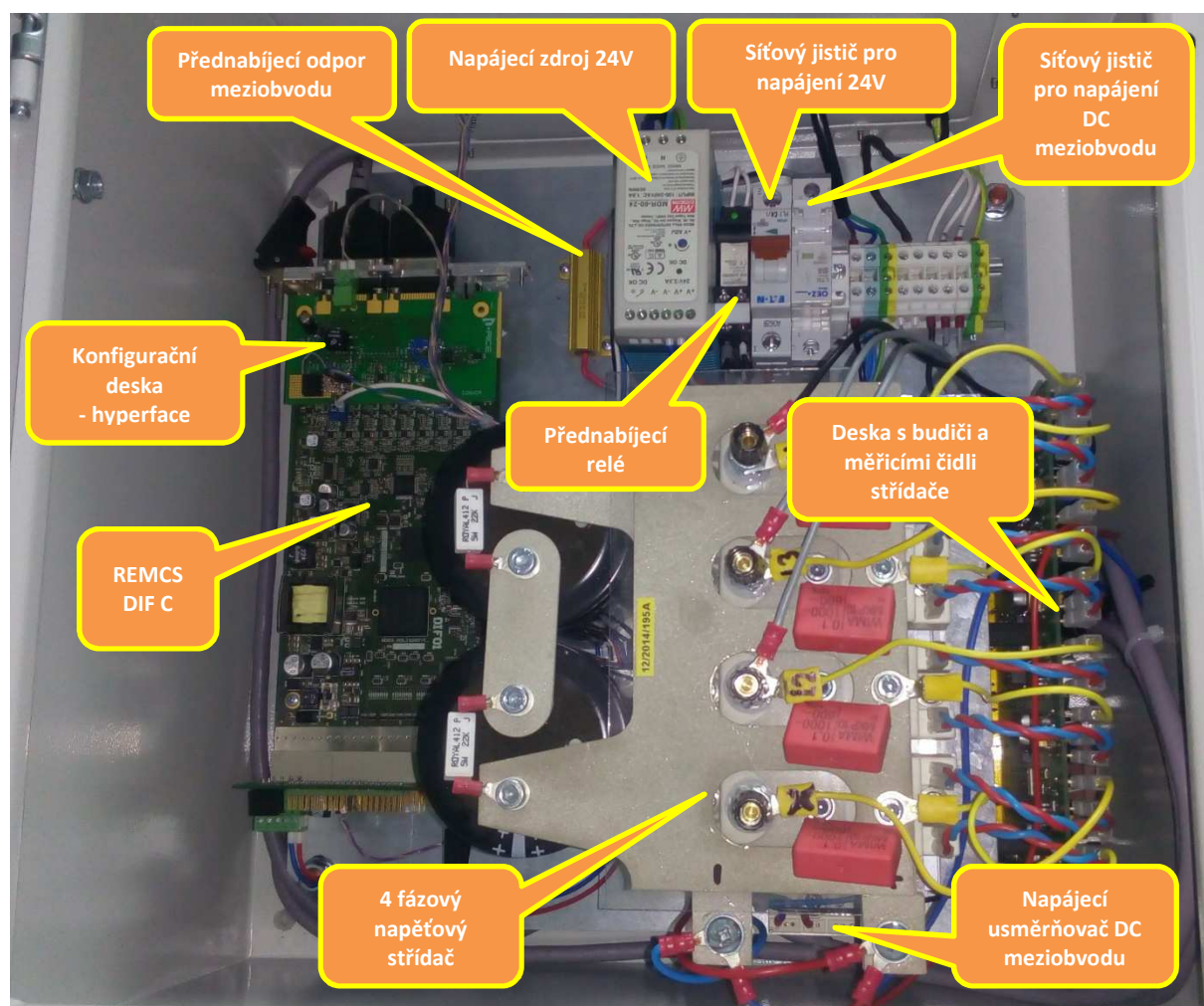
Tato výzkumná zpráva vznikla jako uživatelský manuál pro uživatele daného pohonu. Zpráva tedy uživatelsky definuje veškeré hardwarové a softwarové části. První část je věnována základnímu popisu hardwaru všech součástí: frekvenční měnič, motor a interface pro nadřazené řízení. V druhé části je rozebrána softwarová část z důvodu pochopení funkčnosti a možnosti parametrizace zařízení. Popis je tak směřován na vektorové řízení motoru a komunikaci pro nadřazené řízení.

2 Základní popis hardwaru

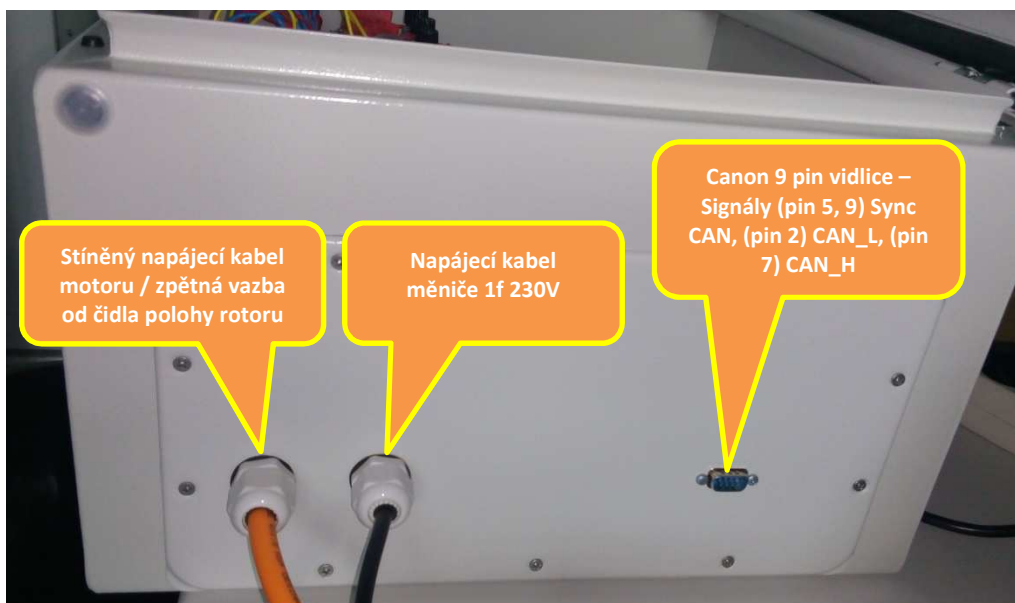
Sestavený pohon obsahuje servomotor od výrobce TG drives, měnič kmitočtu a povelující PC s CANoe hardware and software interface pro nadřazené řízení měniče po CAN sběrnici. Dále jsou blíže popsány jednotlivé části.

2.1 Měnič kmitočtu

Fyzické provedení měniče je znázorněno na Obr. 2.1 a Obr. 2.2. Měnič je založen na jedné řídící kartě ze systému REMCS. Přesně DIF (Direct interface) rev. C. Tato karta obsahuje procesor Hercules TMS570LS Safety (ARM® Cortex®- R4F) a hradlové pole ALTERA cyclone 3. Procesor se stará o regulaci pohonu a hradlové pole o PWM modulaci a ovládání třífázového napěťového měniče (1 fáze měniče není využívána). Popis výstupního signálového konektoru je uveden v Tab. I.



Obr. 2.1 Vnitřní uspořádání měniče kmitočtu



Obr. 2.2 Vnější vstupy/výstupy měniče kmitočtu

Tab. I Výstupní signálový konektor (Canon 9 pin, vidlice)

PIN	Signál
1	NC
2	CAN_L
3	NC
4	NC
5	GND
6	NC
7	CAN_H
8	NC
9	SyncCAN

CAN_H a CAN_L - standardní signály CAN sběrnice.

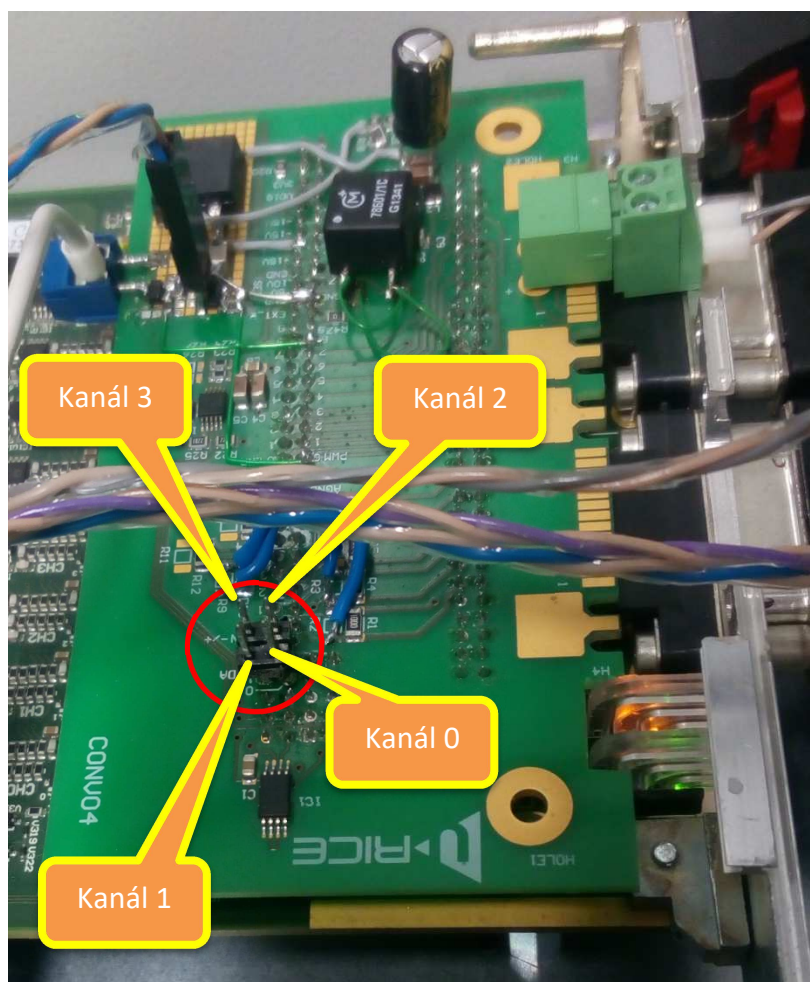
SyncCAN – požadovaný logický signál, který se neuguje vždy při odesílání zpráv sběrnice CAN. Pomocí nařazeného řízení lze měnit tuto periodu viz signál **CanPeriod** ve zprávě RxControl. Fyzicky se jedná o výstup otevřený kolektor.

GND – zem pro otevřený kolektor

2.1.1 D/A převodník

Z řídicí karty je vyveden výstupní konektor D/A převodníku viz Obr. 2.3 s označením kanálů. Jednotlivé výstupy jsou konfigurovány následovně:

Pro všechny kanály platí, že nulová hodnota veličiny odpovídá 2.5V a rozsah převodníku je 0 – 5V.



Obr. 2.3 D/A převodník – konektor

Konfigurace výstupů převodníku pro režim Speed control:

K0: n_w – požadované mechanické otáčky, rozsah: ± 1480 ot/min

K1: I_{sq_w} – požadovaný proud, rozsah: $\pm 2,828$ A

K2: n – měřené mechanické otáčky, rozsah: ± 1480 ot/min

K3: I_{sq} – měřený moment, rozsah: $\pm 2,828$ A

Konfigurace výstupů převodníku pro režim I_{sq} request:

K0: není využíván

K1: I_{sq_w} – požadovaný proud, rozsah: $\pm 2,828$ A

K2: n – měřené mechanické otáčky, rozsah: ± 1480 ot/min

K3: I_{sq} – měřený moment, rozsah: $\pm 2,828$ A

Konfigurace výstupů převodníku pro režim Torque request:

K0: není využíván

K1: T_w – požadovaný moment, rozsah: $\pm 14.4 \text{ Nm}$ ($\pm 3T_N$)

K2: n – měřené mechanické otáčky, rozsah: $\pm 1480 \text{ ot/min}$

K3: T – měřený moment, rozsah: $\pm 14.4 \text{ Nm}$

2.2 Servomotor

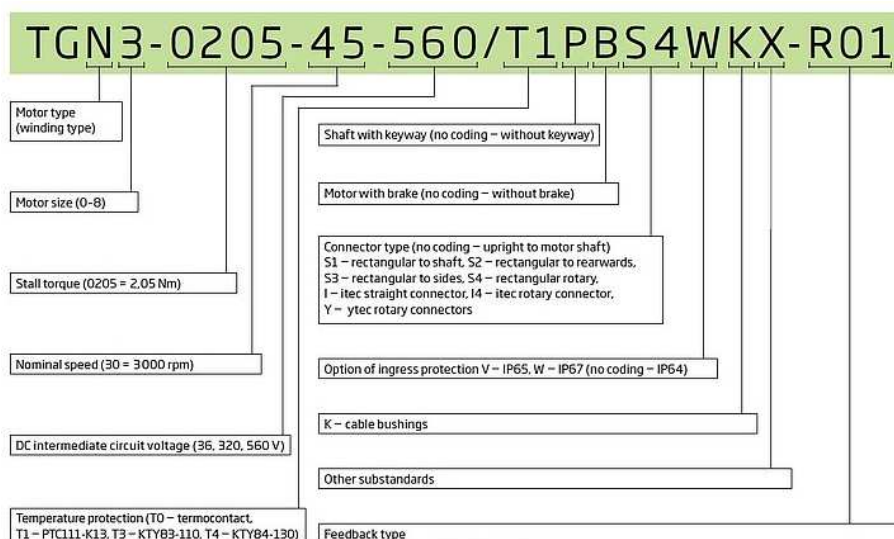
Jedná se o motor výrobce TG drives s označením N3-0480-5-320a/T3 S4-S05. Základní popis motoru uvádí následující Tab. II s následnými vysvětlivkami v podobě Obr. 2.4, Obr. 2.5 a Obr. 2.6, které jsou získány z manuálů výrobce [1].

Tab. II Parametry servomotoru

RATED DATA

Motor type

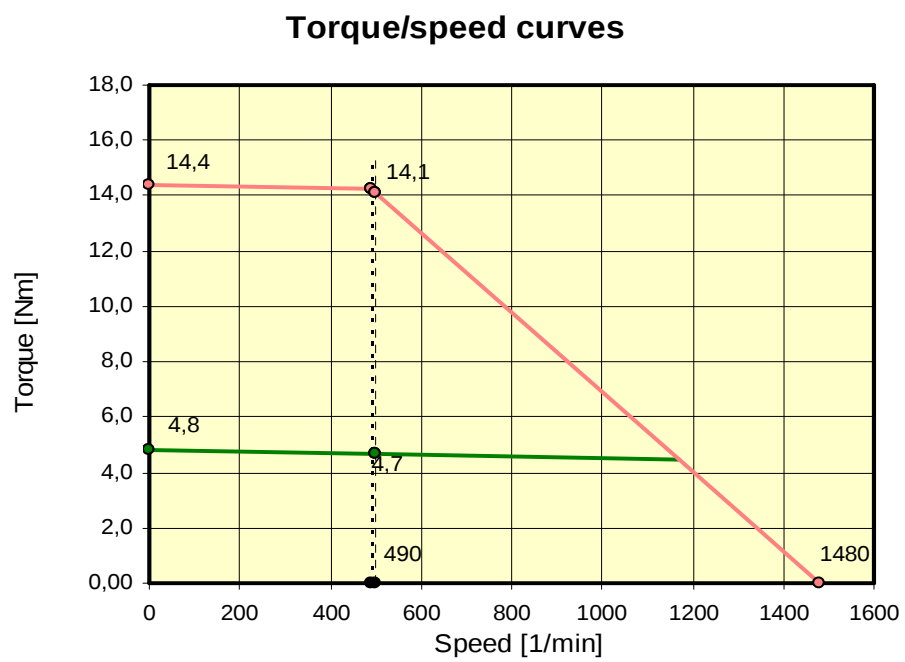
	N3	0480	5	320
Rated Speed	n_n	500	min^{-1}	
DC Bus Voltage	U_{dc}	320	V	
Nominal AC Voltage	U_n	220	V	
Rated Motor Voltage	U_m	113	V	
Rated Torque	M_n	4.7	Nm	
Rated AC Current	I_n	2.0	A	
Stall Torque	M_o	4.8	Nm	
Stall AC Current	I_o	1.96	A	
Peak Torque	M_{max}	14.4	Nm	
Peak Current	I_{max}	7.0	A	
Max. Speed	n_{max}	12000	min^{-1}	
EMF Constant	K_E	148.0	V/1000	
Torque Constant	K_T	2.5	Nm/A	
Terminal Resistance	R_{2ph}	15.5	Ω	
Terminal Inductance	L_{2ph}	46.6	mH	
Number of poles	$2p$	10		
No Load Speed	n_o	1480	min^{-1}	
Torque at I_{max}/U_n	M_z	14.2	Nm	
Speed at I_{max}/U_n	n_z	490	min^{-1}	
Max. Torque at n_n	M_x	14.1	Nm	
El. Time Constant	T_{el}	3.0	ms	
Mech. Time Constant	T_{mech}	0.68	ms	
Thermal Time Constant	T_{th}	30	min	
Rotor Inertia	J	1.5	kgcm^2	



Obr. 2.4 Definice kódu motoru

S – Sick (Hiperface, HDLS)	
S01	Sick EKS36 (HDLS), single-turn, 17 bits
S02	Sick EKM36 (HDLS), multi-turn, 17 bits
S03	Sick EKS36 (HDLS), single-turn, 18 bits
S04	Sick EKM36 (HDLS), multi-turn, 18 bits
S05	Sick EKS36 (HDLS), single-turn, 20 bits
S06	Sick EKM36 (HDLS), multi-turn, 20 bits
S21	Sick SR550 (Hiperface), single-turn, 1024 imp./rev.
S22	Sick SRM50 (Hiperface), multi-turn, 1024 imp./rev.
S23	Sick SKS36 (Hiperface), single-turn, 128 imp./rev.
S24	Sick SKM36 (Hiperface), multi-turn, 128 imp./rev.
S25	Sick SEK34 (Hiperface), single-turn, 16 imp./rev.
S26	Sick SEL34 (Hiperface), multi-turn, 16 imp./rev.
S27	Sick SEK37 (Hiperface), single-turn, 16 imp./rev.
S28	Sick SEL37 (Hiperface), multi-turn, 16 imp./rev.
S29	Sick SEK52 (Hiperface), single-turn, 16 imp./rev.
S30	Sick SEL52 (Hiperface), multi-turn, 16 imp./rev.
S31	Sick SEK90 (Hiperface), single-turn, 64 imp./rev.
S32	Sick SEK160 (Hiperface), single-turn, 128 imp./rev.
S33	Sick SEK260 (Hiperface), single-turn, 256 imp./rev.

Obr. 2.5 Typy zpětnovazebních čidel firmy Sick



Obr. 2.6 Momentové charakteristiky servomotoru

2.3 Nadřazené řízení

Z pohledu hardware je nadřazené řízení prozatím prováděno pomocí CANoe Vector VN1640A CAN/LIN Interface viz [2]. Součástí interface je GUI pro nadřazené PC.

3 Popis software

3.1 Řídicí software pohonu

Pohon je řízen standardním vektorovým řízením v kartézských souřadnicích, které vzniklo v rámci předchozích prací na projektu CIDAM. Blokové schéma řízení je uvedeno na Obr. 3.1. Celkové schéma je rozděleno na jednotlivé bloky, které jsou fyzicky tvořeny třídami nebo funkcemi v jazyce C/C++. Základem je již zmiňované základní vektorové řízení v třídě C_PMSM_VC_Variant_1. Tato část se stará o regulaci jednotlivých proudů I_{sd} a I_{sq} . Požadavky pro tyto proudy tvoří jak regulátor otáček R_w (požadavek I_{sq}), tak smyčka odbuzování C_SM_Flux_Weakening (požadavek I_{sd}). Prostředníkem mezi těmito bloky a základním vektorovým řízením je blok C_PMSM_VC_Beta_limitation. Pro danou aplikaci dochází pouze k omezování vektoru proudu na maximální hodnotu motoru, ale není využívána část omezování zátěžného úhlu, kterou provádí regulátor R_{uq} . Jeho zesílení jsou tedy nastaveny na nulové hodnoty. Daný kód podporuje přepínání režimů regulace otáček (Speed control), regulace proudu i_{sq} (i_{sq} request) a regulace momentu (Torque request). V režimech regulace proudu i_{sq} a regulace momentu je vyřazen regulátor otáček a požadavek je přímo nasměrován do signálu, který vychází z regulátoru otáček.

Dále je uveden výčet parametrů a jejich hodnoty, které je nutné nastavit při inicializaci řízení.

Výčet parametrů:

/2017_05_18_dif01C_VC_PMSM_classes_offline/source/application/main.cpp

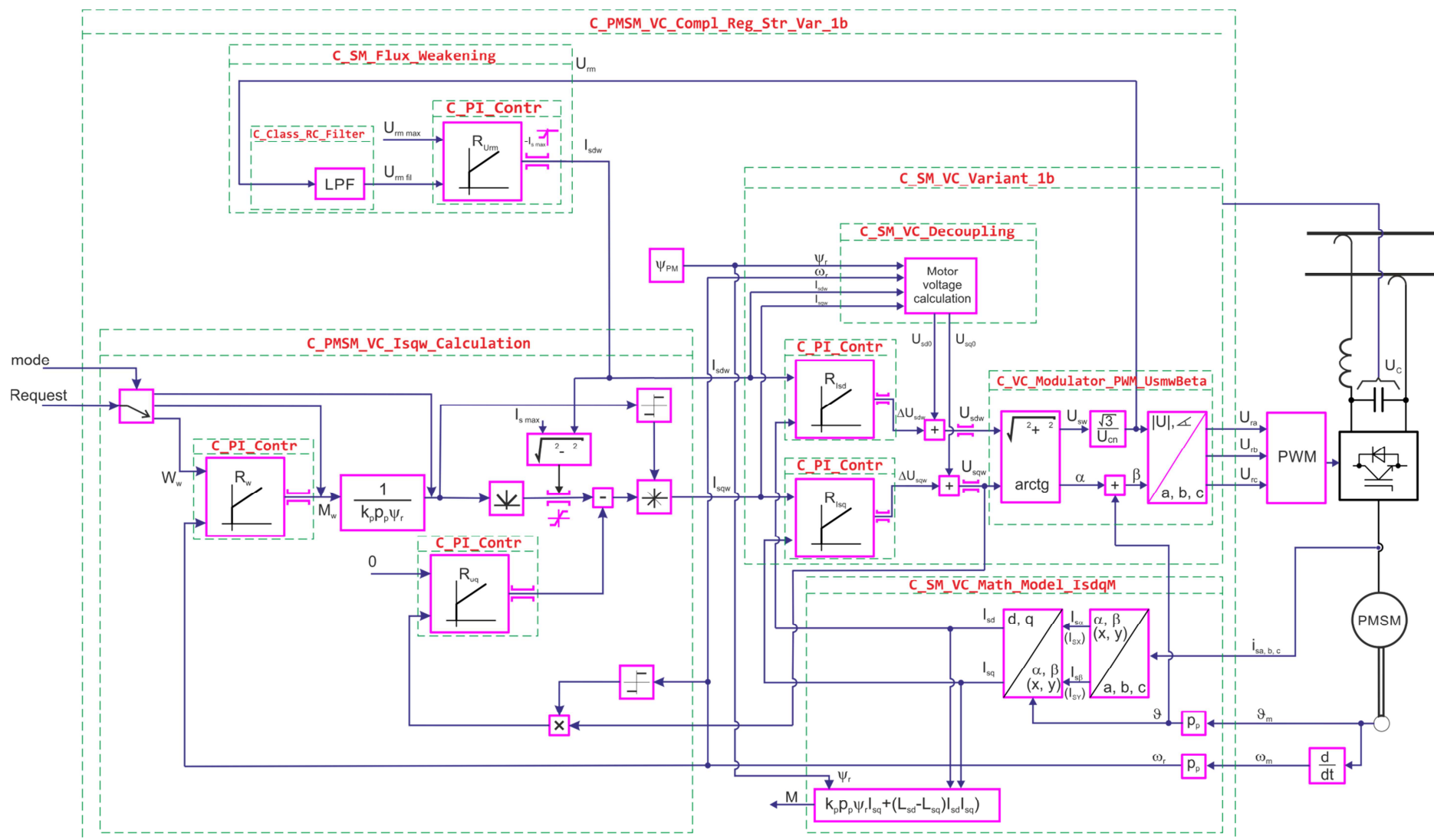
Řádek kódu 209 - 233

```
VC_Par.Math_Model.Lsd=LS;
VC_Par.Math_Model.Lsq=LS;
VC_Par.Flux_Weak.Psi_pm=FPM;
VC_Par.Math_Model.Rs=RS;
VC_Par.Math_Model.pp=PP;

VC_Par.Vect_Contr.Contr_Isd.Kp=INV_KP_IS;
VC_Par.Vect_Contr.Contr_Isd.Tau=INV_TR_IS;
VC_Par.Vect_Contr.Contr_Isd.HiLimit=INV_DUSDW_MAX;
VC_Par.Vect_Contr.Contr_Isd.LoLimit=INV_DUSDW_MIN;

VC_Par.Vect_Contr.Contr_Isq.Kp=INV_KP_IS;
VC_Par.Vect_Contr.Contr_Isq.Tau=INV_TR_IS;
VC_Par.Vect_Contr.Contr_Isq.HiLimit=INV_DUSQW_MAX;
VC_Par.Vect_Contr.Contr_Isq.LoLimit=INV_DUSQW_MIN;

VC_Par.Flux_Weak.Contr_Urm.Kp=KP_URM;
VC_Par.Flux_Weak.Contr_Urm.Tau=TR_URM;
VC_Par.Flux_Weak.Modul_Index.Tau_Urm_fil=TF;
VC_Par.Flux_Weak.Modul_Index.Urm_max=URM_MAX;
VC_Par.Flux_Weak.Syn_Motor.Ism_n=ISM;
VC_Par.Isqw_Calcul.Contr_Uq.Kp=0;
VC_Par.Isqw_Calcul.Contr_Uq.Tau=0;
VC_Par.Isqw_Calcul.Contr_Wr.Kp=KP_WR;
VC_Par.Isqw_Calcul.Contr_Wr.Tau=TR_WR;
```



Obr. 3.1 Blokové schéma řízení pohonu

Dále je nutný parametr jmenovitého napětí ve stejnosměrném meziobvodu U_{CN} . Z důvodu tolerancí v síti je toto napětí měřeno a to po procesu přednabití meziobvodu.

/2017_05_18_dif01C_VC_PMSM_classes_offline/source/application/main.cpp

Řádek kódu 252

VC_Par.Vect_Contr.DC_link.Ucn = UCN;

Hodnoty konstant (popřípadě inicializačních hodnot) parametrů:

/2017_05_18_dif01C_VC_PMSM_classes_offline/source/application/ctr_VectorFloat.h

```
/** \name Motor parameters.
 * \{ */
#define ISM (2 * 1.414f) /**< [A] Maximal stator current.*/
#define RS 7.75f /**< [Ohm] Stator resistance.*/
#define FPM 0.27f /**< [Wb] Stator linkage flux of perm. magnets.*/by count
0.282, from PI outputs 0.27 is more accurate
#define LS 23.3e-3f /**< [H] stator inductance.*/
#define PP 5.0f /**< [A] Number of pole pairs.*/
/** \} */

/** \name Motor inverter control parameters.
 * \{ */

#define TS 100e-6f /**< [s] Current loop sample time.*/

//fieldweakening control
#define TF 0.01f /**< [s] Time constant of the LPF in field weakening line.*/

#define URM_MAX 0.95f /**< [-] Maximal deep of modulation.*/
#define KP_URM 5.0f /**< [-] Proportional constant of field weakening PI
controller.*/
#define TR_URM 0.2f /**< [s] Time constant of integration part of the field
weakening PI controller.*/
#define INV_ISDW_MAX 0.0f /**< [A] Output max. limitation of the field
weakening PI controller.*/
#define INV_ISDW_MIN (-ISM) /**< [A] Output min. limitation of the field
weakening PI controller.*/

//speed control
#define SPEED_SAMPLING_PERIOD 100e-6f /**< [s] Speed loop sample time.*/
//#define WR_EXP_FILTER_LENGTH 10u//wr Exponential filter length
#define SPEED_MAX_LIMIT (2*PI*123.)//max speed 1480rpm -> 123.3Hz el., end of
field-weakening

#define KP_WR 0.0005f
#define TR_WR 0.1f

//current control
#define INV_KP_IS 200.0f//50.0f//100.0f /**< [-] Proportional constant of
stator current PI controllers.*/
#define INV_TR_IS 0.0025f//0.01f /**< [s] Time constant of integration
part of stator current PI controllers.*/
```

```

#define INV_DUSDW_MAX 1000.0f /**< [V] Output max. limitation of the d-axis
stator current PI controller.*/
#define INV_DUSDW_MIN -1000.0f /**< [V] Output min. limitation of the d-axis
stator current PI controller.*/
#define INV_DUSQW_MAX 1000.0f /**< [V] Output max. limitation of the q-axis
stator current PI controller.*/
#define INV_DUSQW_MIN -1000.0f /**< [V] Output min. limitation of q-axis the
stator current PI controller.*/

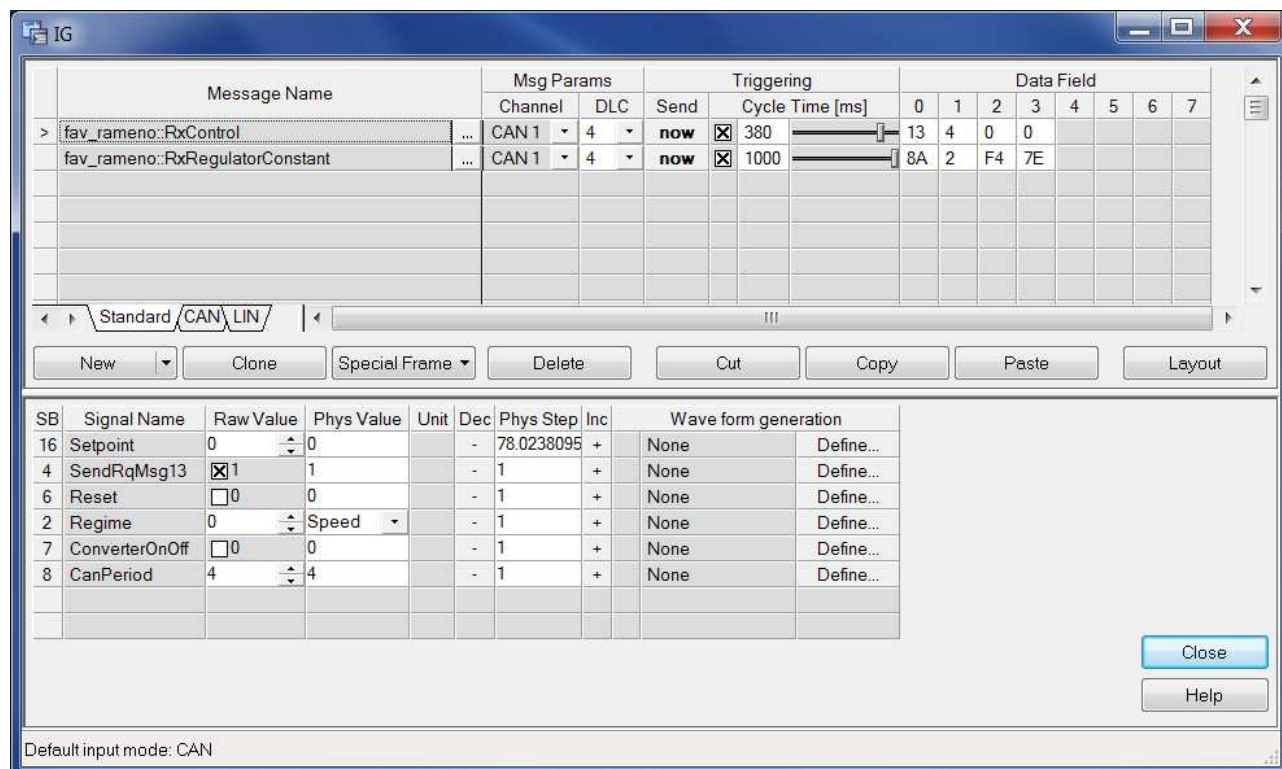
//voltage management
#define CHARGING_HighLevel 360.0f
#define CHARGING_LowLevel 293.0f
#define INV_PROTECTION_VOLTAGE_MAX_LIMIT 440.0f /**< [A] Maximal rectifier DC
voltage.*/
#define INV_PROTECTION_VOLTAGE_MIN_LIMIT 168.0f /**< [A] Minimal rectifier DC
voltage.*/

//current management
#define INV_PROTECTION_CURRENT_LIMIT 8.9f /**< [A] Maximal inv. current.*/

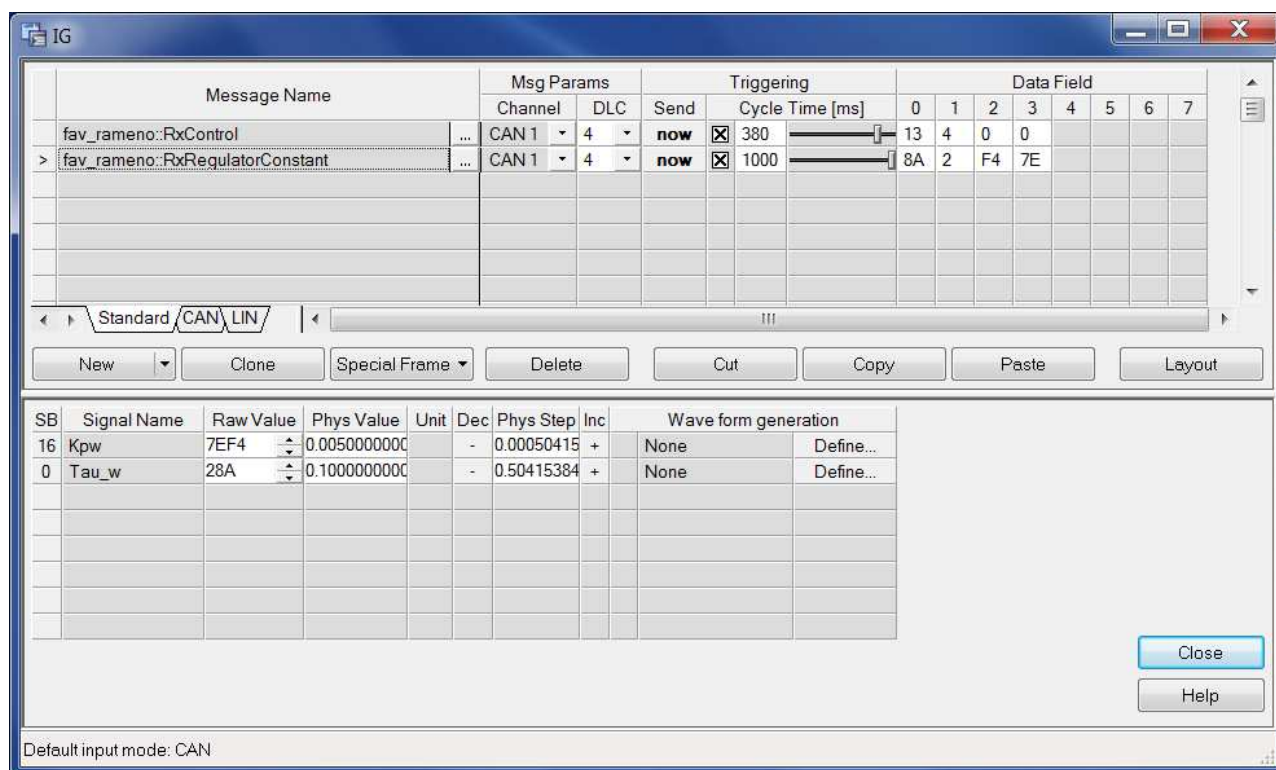
```

3.2 Nadřazené řízení

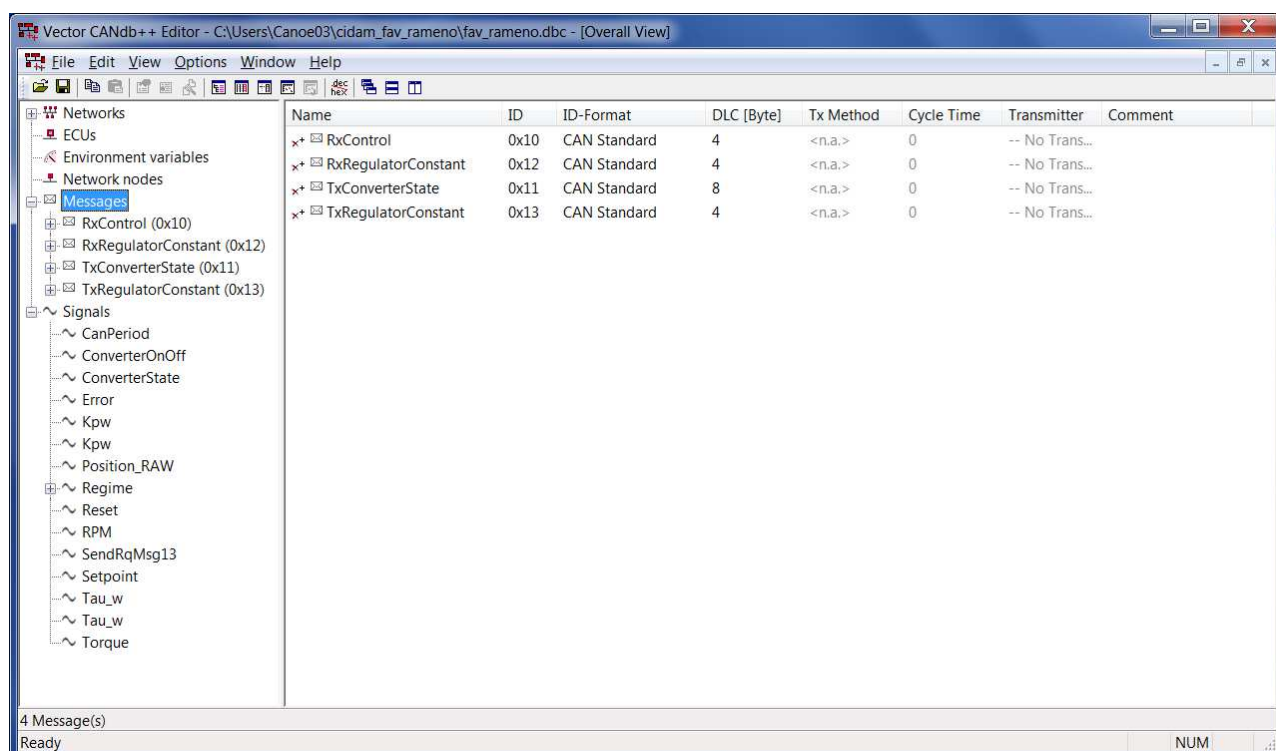
Nadřazené řízení je základně provedeno v grafickém prostředí výrobce CAN interface. Povelování pohonu je možné základně přes tzv. Interactive Generator Block (IG) viz Obr. 3.2 a Obr. 3.3. Pomocí tohoto objektu je možné konfigurovat odesílané zprávy a odesílat je, popřípadě nastavit jejich automatické, periodické odesílání. Dané zprávy musí být předem nakonfigurovány. Jejich strukturu a jednotlivé bity seskupené do signálů uvádí Obr. 3.4 a Obr. 3.5. V následujícím textu budou detailně rozebrány jednotlivé zprávy a jejich signály. Označení zpráv Rx a Tx je bráno z pohledu frekvenčního měniče.



Obr. 3.2 IG pro řídicí zprávu RxControl



Obr. 3.3 IG pro zprávu se změnou konstant regulátorů RxRegulatorConstant



Obr. 3.4 CAN zprávy

Name	Length [Bit]	Byte Order	Value Type	Initial Value	Factor	Offs...	Mini...	Max...	Unit	Value Table	Comment
~ CanPeriod	4	Intel	Unsigned	0	1	0	0	0		<none>	
~ ConverterOnOff	1	Intel	Unsigned	0	1	0	0	0		<none>	
~ ConverterState	1	Intel	Unsigned	0	1	0	0	0		<none>	
~ Error	4	Intel	Unsigned	0	1	0	0	0		<none>	
~ Kpw	16	Intel	Unsigned	0	1.53846e-007	0	0	0		<none>	
~ Kpw	16	Intel	Unsigned	0	1.53846e-007	0	0	0		<none>	
~ Position_RAW	32	Intel	Unsigned	0	1	0	0	0		<none>	
~ Regime	2	Intel	Unsigned	0	1	0	0	0		VtSig_Regi...	
~ Reset	1	Intel	Unsigned	0	1	0	0	0		<none>	
~ RPM	16	Intel	Signed	0	1	0	0	0		<none>	
~ SendRqMsg13	1	Intel	Unsigned	0	1	0	0	0		<none>	
~ Setpoint	16	Intel	Signed	0	0.0238095	0	0	0		<none>	
~ Tau_w	16	Intel	Unsigned	0	0.000153846	0	0	0		<none>	
~ Kpw	16	Intel	Unsigned	0	0.000153846	0	0	0		<none>	
~ Torque	16	Intel	Signed	0	0.0001	0	0	0		<none>	

Obr. 3.5 Signály CAN zpráv

3.2.1 RxControl 0x10

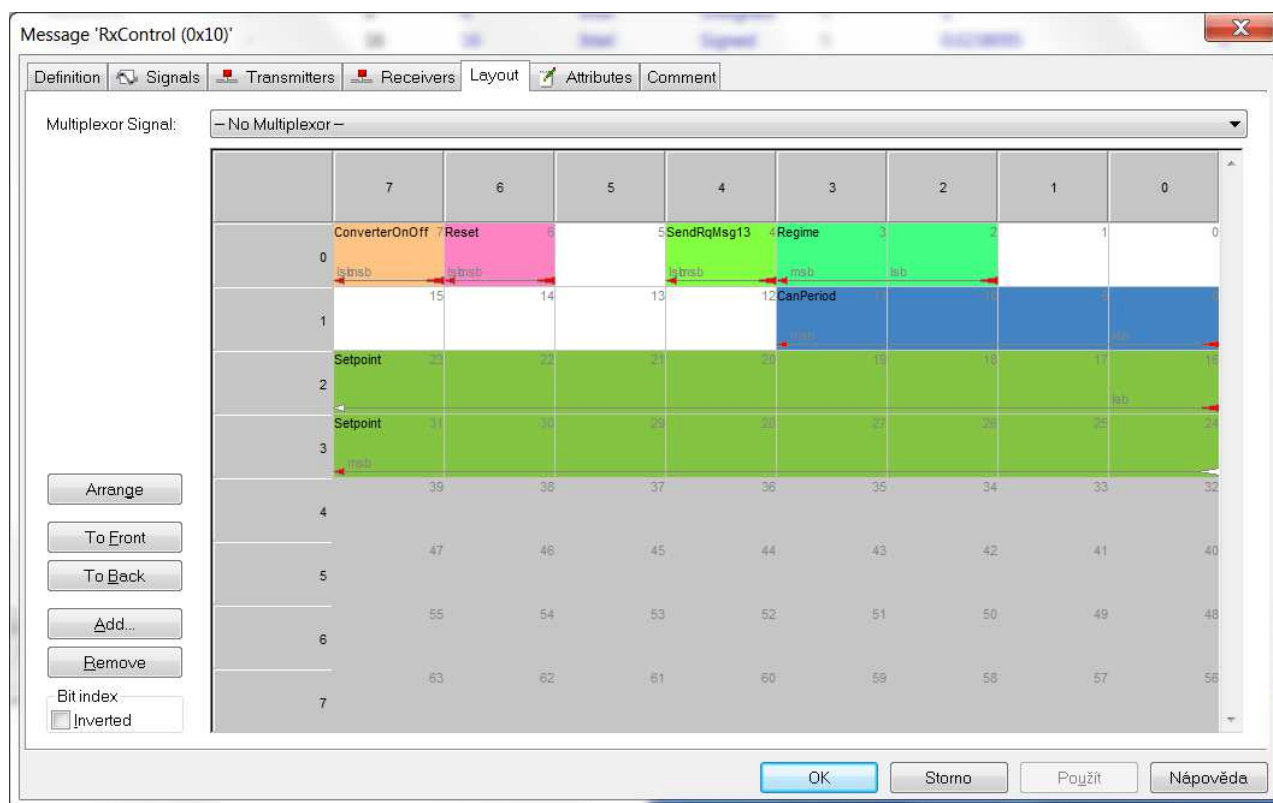
Jedná se o základní řídicí zprávu měniče. Její popis a rozvržení ukazují Obr. 3.6 a Obr. 3.7. Zpráva obsahuje následující signály:

ConverterOnOff – slouží k zapnutí a vypnutí spínacích signálů měniče, tedy stejnosměrný obvod měniče je stále přednabit a napájen, ale vypíná se pouze napájení motoru.

1 – Zap, 0 - Vyp

Name	Message	Multiplexin...	Startbit	Length [Bit]	Byte Order	Value Type	Initial Value	Factor	Offset	Minimum	Maximum	Unit	Value Table	Comm
~ Regime	RxControl	-	2	2	Intel	Unsigned	0	1	0	0	0		VtSig_Regi...	
~ SendRqMsg13	RxControl	-	4	1	Intel	Unsigned	0	1	0	0	0		<none>	
~ Reset	RxControl	-	6	1	Intel	Unsigned	0	1	0	0	0		<none>	
~ ConverterOnOff	RxControl	-	7	1	Intel	Unsigned	0	1	0	0	0		<none>	
~ CanPeriod	RxControl	-	8	4	Intel	Unsigned	0	1	0	0	0		<none>	
~ Setpoint	RxControl	-	16	16	Intel	Signed	0	0.0238095	0	0	0		<none>	

Obr. 3.6 Struktura zprávy RxControl



Obr. 3.7 Rozložení zprávy RxControl

Reset – provádí kompletní restart řídicí karty a software. Během opětovného náběhu je provedeno přednabití stejnosměrného meziobvodu a odmazání errorů měniče.

1 – reset stav, 0 - neaktivní

Regime – nastavuje režim pohonu resp. vektorové regulace

2 – regulace proudu i_{sq} (i_{sq} request), 1 – regulace momentu (Torque request), 0 – regulace otáček (Speed control)

SendRqMsg13 – umožňuje zapnout nebo vypnout posílání zprávy s hodnotami konstant regulátorů

1 – zpráva 0x13 je posílána, 0 - zpráva 0x13 není posílána

CanPeriod – čtyřbitové číslo jímž je možné nastavit periodu CAN komunikace od trojnásobku periody vzorkování řízení pohonu ($3 \times 100\mu s$) až do šestnáctinásobku ($16 \times 100\mu s$). Je zde provedeno omezení na min délku periody o délce trojnásobku periody vzorkování řízení pohonu z důvodu rychlosti posílání zpráv.

0,1,2 – $T_{CAN} = 300\mu s$; 3 – $T_{CAN} = 400\mu s$; ... F – $T_{CAN} = 1600\mu s$

pro periodu vzorkování řízení pohonu $100\mu s$

Setpoint – znaménkové šestnáctibitové číslo které představuje požadovanou hodnotu řízení pohonu. Dle režimu pohonu reprezentuje:

1) požadovanou úhlovou rychlost rotoru v elektrických stupních

Pro převod čísla do plovoucí řádové čárky je použit faktor:

```
#define SCALE_w (1.0f/42.0f)//42scale ... 1480rpm->775 wr
```

Float hodnota požadavku může tedy nabývat hodnot cca ± 780 rad/s (elektrická úhlová rotorová rychlost)

2) požadovaný moment motoru

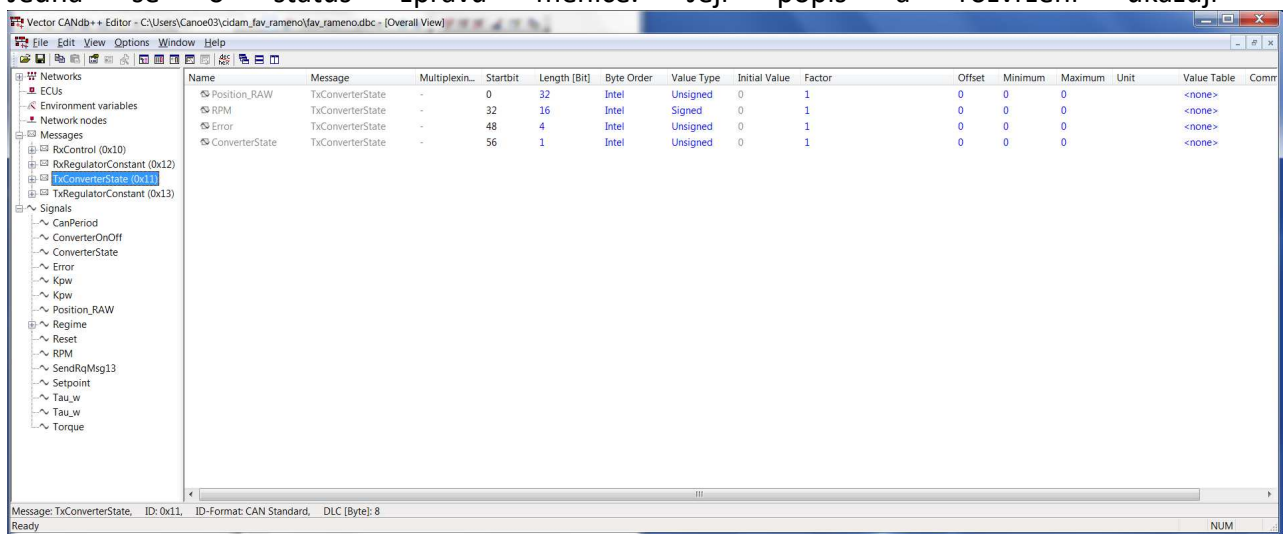
Pro převod čísla do plovoucí řádové čárky je použit faktor:

```
#define SCALE_T (1.0f/10922.0f)//ca 14.4 Nm max
```

Float hodnota požadavku může tedy nabývat hodnot cca ± 3 [-], kde se jedná o normovanou hodnotu s normou o velikosti jmenovitého momentu $T = 4.8\text{Nm}$. Maximální požadavek je tedy $\pm 14.4\text{Nm}$.

3.2.2 TxConverterState 0x11

Jedná se o status zprávu měniče. Její popis a rozvržení ukazují

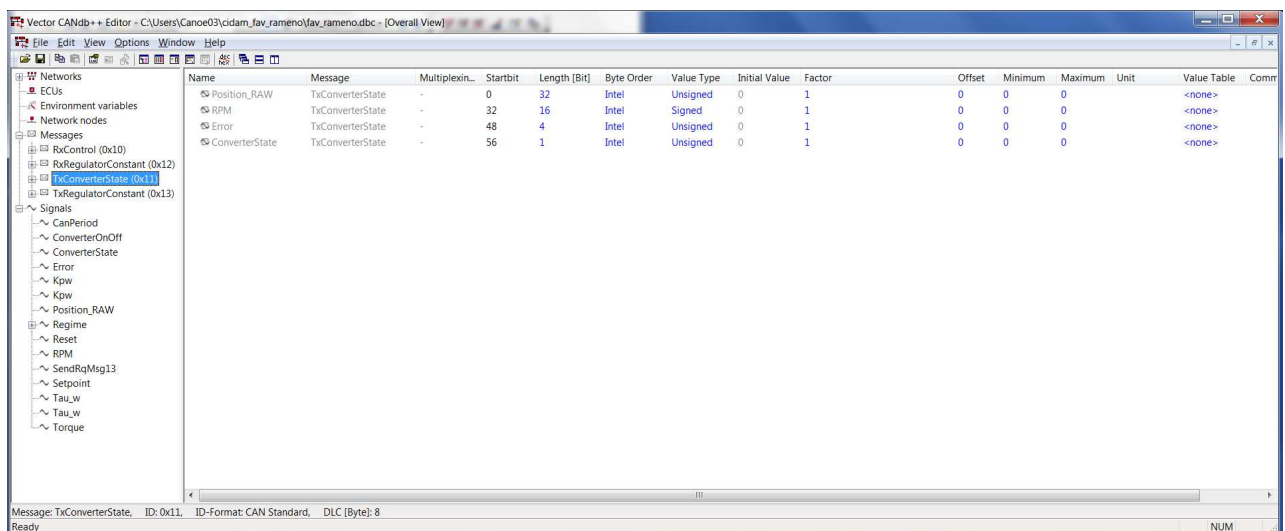


The screenshot shows the Vector CANdb++ Editor interface. The left sidebar displays a tree view of the CAN database, with 'TxConverterState (0x11)' selected under the 'Messages' folder. The main window displays a table with the following columns: Name, Message, Multiplexin..., Startbit, Length [Bit], Byte Order, Value Type, Initial Value, Factor, Offset, Minimum, Maximum, Unit, Value Table, and Comm... The table contains the following data:

Name	Message	Multiplexin...	Startbit	Length [Bit]	Byte Order	Value Type	Initial Value	Factor	Offset	Minimum	Maximum	Unit	Value Table	Comm...
Position_RAW	TxConverterState	-	0	32	Intel	Unsigned	0	1	0	0	0		<none>	
RPM	TxConverterState	-	32	16	Intel	Signed	0	1	0	0	0		<none>	
Error	TxConverterState	-	48	4	Intel	Unsigned	0	1	0	0	0		<none>	
ConverterState	TxConverterState	-	56	1	Intel	Unsigned	0	1	0	0	0		<none>	

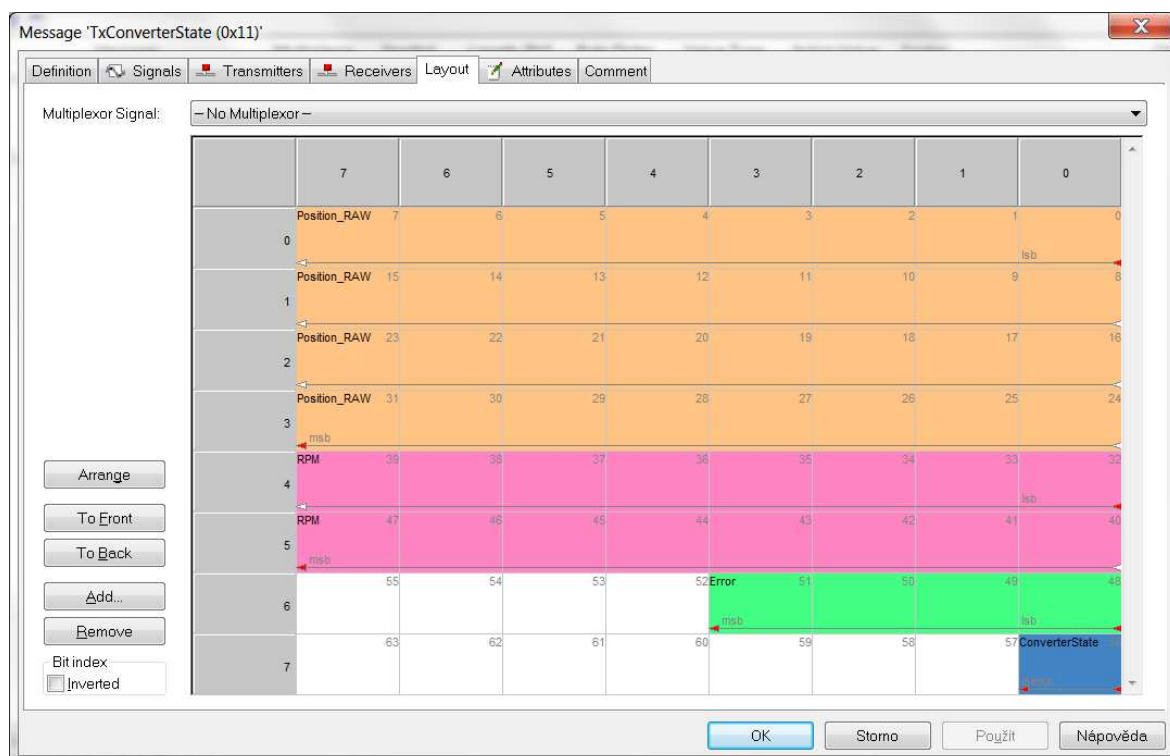
At the bottom of the window, the status bar indicates: Message: TxConverterState, ID: 0x11, ID-Format: CAN Standard, DLC [Byte]: 8, Ready, NUM.

Obr. 3.8 a Obr. 3.9.



This screenshot is identical to the one above, showing the same Vector CANdb++ Editor interface with the 'TxConverterState (0x11)' message structure table.

Obr. 3.8 Struktura zprávy TxConverterState



Obr. 3.9 Rozvržení zprávy TxConverterState

Zpráva obsahuje následující signály:

Position_RAW – celé 4B informace o poloze rotoru z čidla.

Torque – znaménkové šestnáctibitové číslo které představuje moment motoru

Pro převod čísla do plovoucí řádové čárky je použit faktor:

`#define SCALE_T (1.0f/10922.0f)//ca 14.4 Nm max`

Float hodnota požadavku může tedy nabývat hodnot cca ± 3 [-], kde se jedná o normovanou hodnotu s normou o velikosti jmenovitého momentu $T = 4.8\text{Nm}$. Maximální moment je tedy $\pm 14.4\text{Nm}$.

Error – indikace erroru měniče. Následuje zablokování spínacích signálů měniče a uvedení do error stavu s vyblikáváním kodu erroru pomocí `FPG_LED_ERROR`. Pro vymazání je nutné poslat jednotkový signál **reset** ve zprávě RxControl.

0 – NO_ERROR - bez erroru

1 – CHARGING_UNDERVOLTAGE – přednabíjení - nízké napětí DC meziobvodu $U_c < 293\text{V}$, nabíjecí rezistor zařazen

2 – CHARGING_OVERVOLTAGE – přednabíjení - vysoké napětí DC meziobvodu $U_c > 360\text{V}$, nabíjecí rezistor zařazen

3 – UCN_UNDERVOLTAGE – nízké napětí DC meziobvodu $U_{cN} < 293\text{V}$, měření U_{cN} – nabíjecí rezistor překlenut

4 – UCN_OVERVOLTAGE – vysoké napětí DC meziobvodu $U_{cN} > 360\text{V}$, měření U_{cN} – nabíjecí rezistor překlenut

5 – OVERCURRENT - překročení maximálního fázového proudu statoru $I_{s\text{MAX}} = 8.9\text{A}$ (okamžitá hodnota)

6 – VOLTAGE_ERROR - napětí DC meziobvodu mimo bezpečné meze (168V, 440V)

7 – OVERSPEED - překročení maximální rychlosti $n_{MAX} = 1480\text{rpm} \Rightarrow 775\text{rad/s}$

8 – POSITION_ERROR – hyperface viz [3] – chyba čidla polohy rotoru

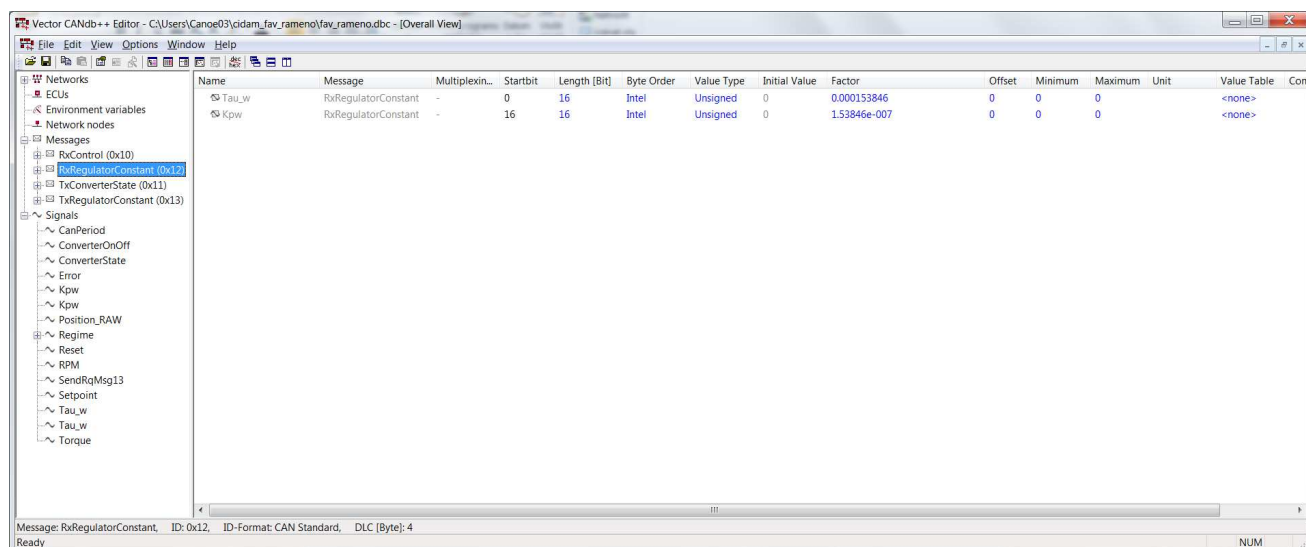
9 – ENCODING_ERROR - hyperface viz [3] – chyba komunikace

ConverterState – indikace chodu měniče

1 – Zap, 0 - Vyp

3.2.3 RxRegulatorConstant 0x12

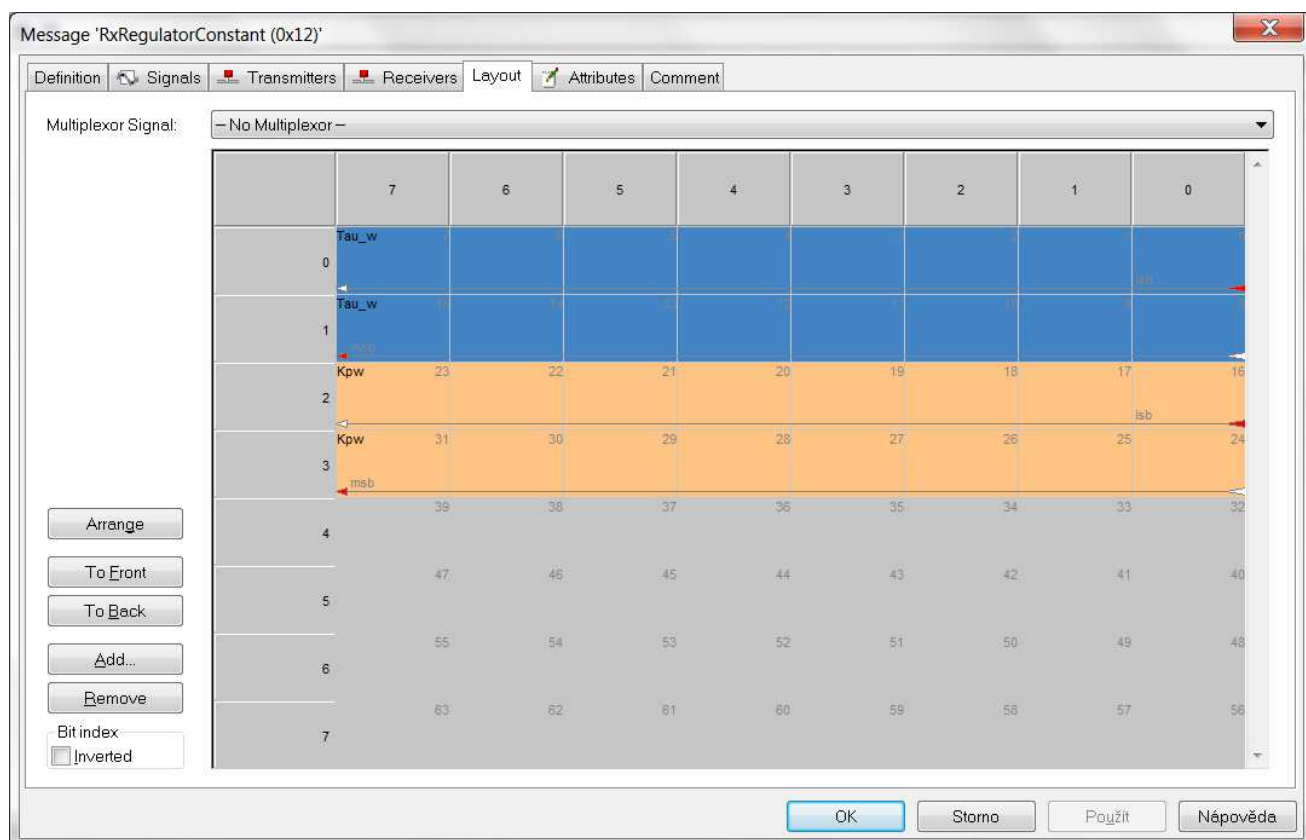
Jedná se o zprávu nastavující konstanty PI regulátoru otáček. Popis a rozvržení zprávy ukazují Obr. 3.10 a Obr. 3.11.



Name	Message	Multiplexin...	Startbit	Length [Bit]	Byte Order	Value Type	Initial Value	Factor	Offset	Minimum	Maximum	Unit	Value Table	Comments
Tau_w	RxRegulatorConstant	-	0	16	Intel	Unsigned	0	0.000153846	0	0	0		<none>	
Kpw	RxRegulatorConstant	-	16	16	Intel	Unsigned	0	1.53846e-007	0	0	0		<none>	

Message: RxRegulatorConstant, ID: 0x12, ID-Format: CAN Standard, DLC [Byte]: 4
Ready

Obr. 3.10 Struktura zprávy RxRegulatorConstant



Obr. 3.11 Rozvržení zprávy RxRegulatorConstant

Zpráva obsahuje následující signály:

Kx – neznaménkové šestnáctibitové číslo které představuje příslušnou konstantu regulátoru.

Kpw – proporční konstanta regulátoru otáček (regulátor úhlové rychlosti)

Kiw – integrační konstanta regulátoru otáček (regulátor úhlové rychlosti)

Pro převod konstant do plovoucí řádové čárky jsou použity faktory:

```
#define SCALE_1_Kpw (1.0f/SCALE_Kpw)
```

```
#define SCALE_1_Kiw (1.0f/SCALE_Kiw)
```

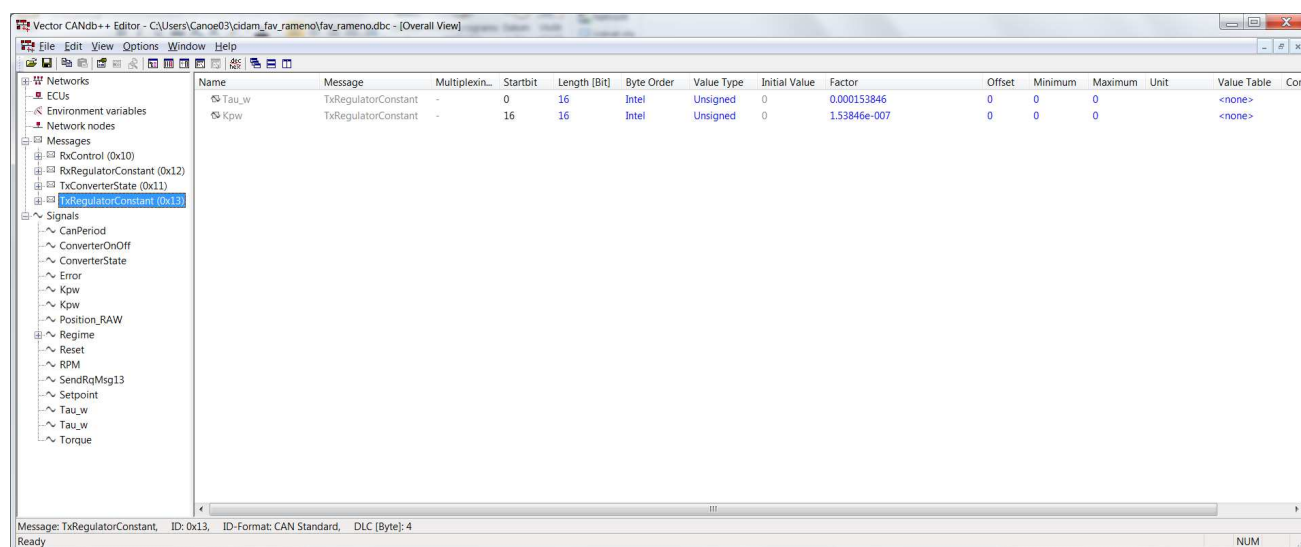
, kde

```
#define SCALE_Kpw 6500000.0f
```

```
#define SCALE_Kiw 65000.0f
```

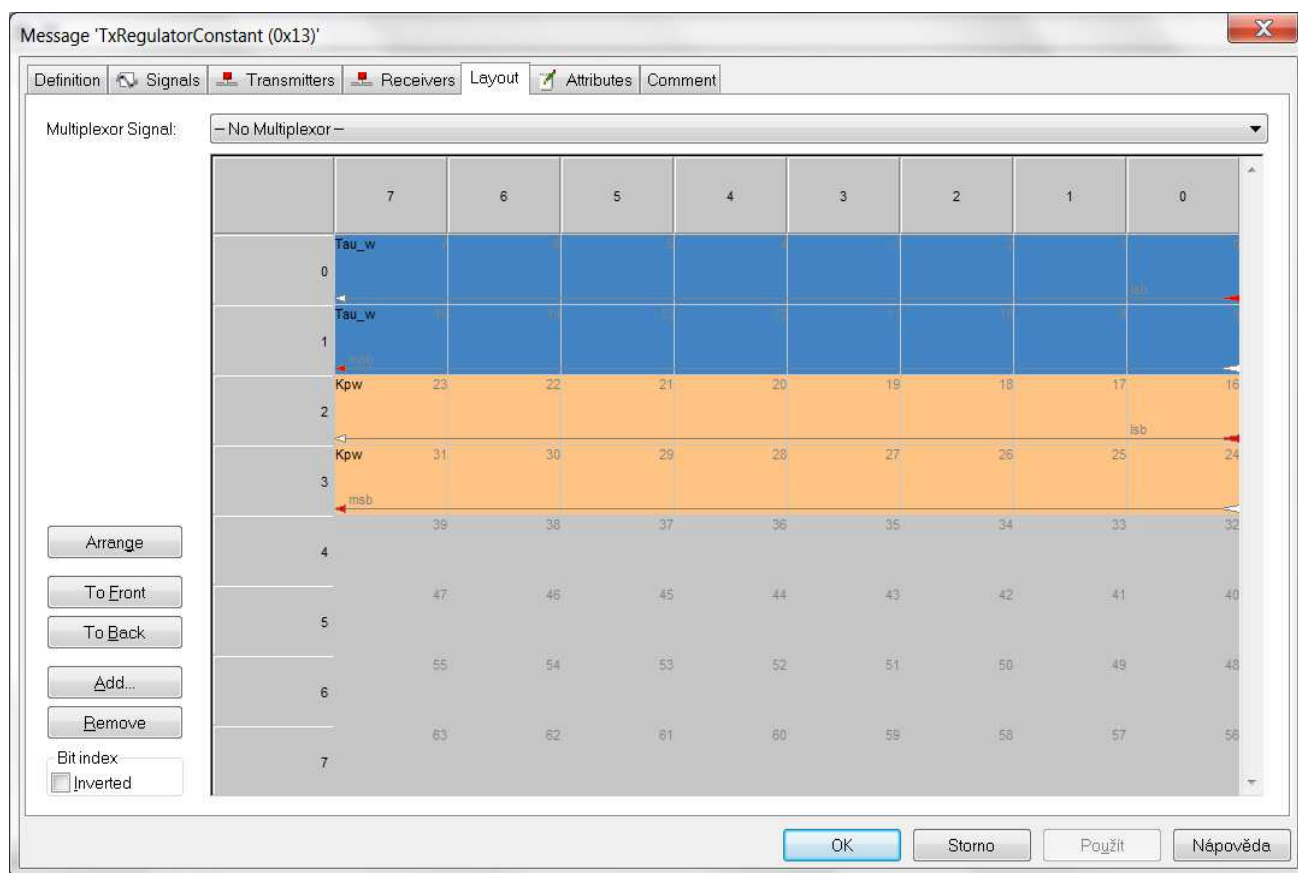
3.2.4 TxRegulatorConstant 0x13

Jedná se o zprávu s aktuálními konstantami PI regulátoru otáček, která je odesílána měničem na požadavek pomocí jednotkového signálu **SendRqMsg13** ve zprávě RxControl. Popis zprávy a její rozvržení ukazují Obr. 3.12 a Obr. 3.13.



Name	Message	Multiplexin...	Startbit	Length [Bit]	Byte Order	Value Type	Initial Value	Factor	Offset	Minimum	Maximum	Unit	Value Table	Cor
Tau_w	TxRegulatorConstant	-	0	16	Intel	Unsigned	0	0.000153846	0	0	0		<none>	
Kpw	TxRegulatorConstant	-	16	16	Intel	Unsigned	0	1.53846e-007	0	0	0		<none>	

Obr. 3.12 Struktura zprávy TxRegulatorConstant

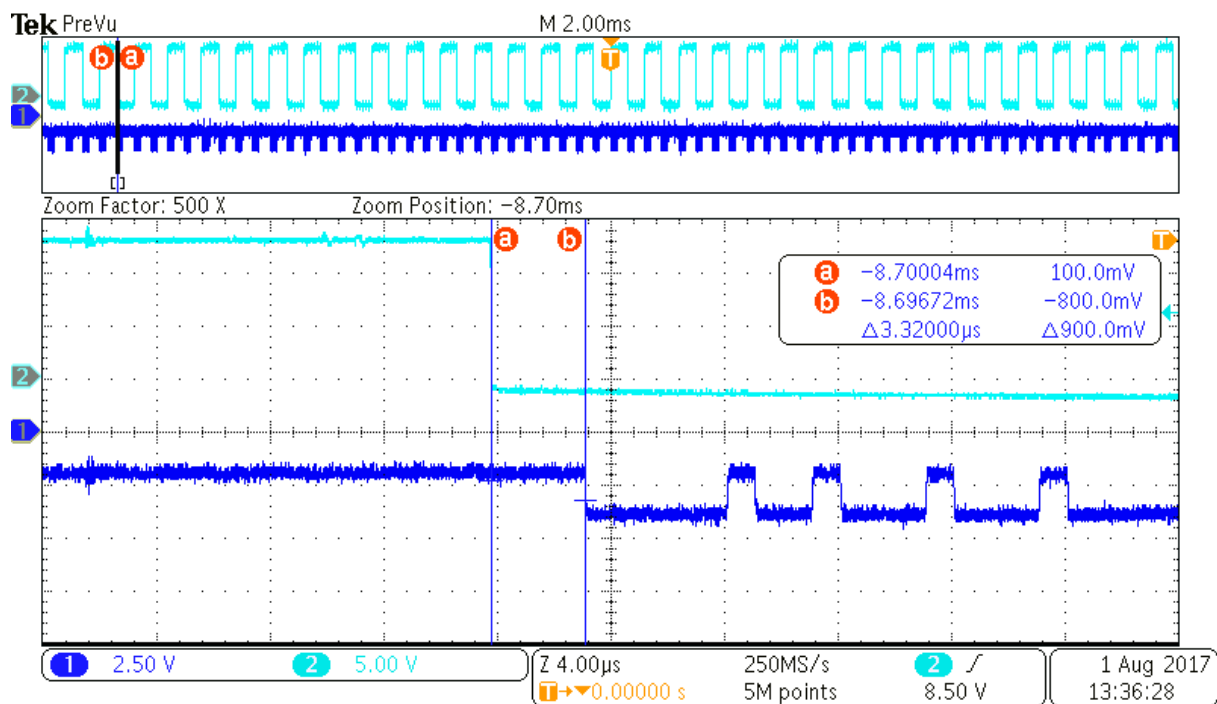


Obr. 3.13 Rozvržení zprávy TxRegulatorConstant

Reprezentace jednotlivých konstant a jejich přepoččet na desetiné číslo je totožný s předchozí zprávou RxRegulatorConstant.

4 Měření

Z důvodu požadavku na konstantní posuv synchronizačního signálu SyncCAN a posílané zprávy CAN byl tento fakt kontrolován viz následující oscilogram na Obr. 4.1. Proměnnost posuvu byla vypočítávána na několika událostech v maximální velikosti $\pm 0.2\mu\text{s}$ od předkládaného měření.



Obr. 4.1 Oscilogram - posuv SyncCAN vs CAN zpráva – měření délky posuvu

5 Závěr

V této zprávě je proveden popis pohonu pro rameno robotu se synchronním motorem s permanentními magnety. Veškeré části zařízení jsou popsány z hlediska uživatelského použití.

Měřením byl ověřen téměř konstantní posuv signálu SyncCAN a posílané první zprávy CAN s deviací v řádu desetin μs .

Literatura

- [1] [http://www.tgdrives.cz/fileadmin/user_upload/download/TG Drives 2014 Katalog servomotory TGN EN.pdf](http://www.tgdrives.cz/fileadmin/user_upload/download/TG_Drives_2014_Katalog_servomotory_TGN_EN.pdf)
- [2] [https://vector.com/portal/medien/cmc/manuals/VN1600 Interface Family Manual EN.pdf](https://vector.com/portal/medien/cmc/manuals/VN1600_Interface_Family_Manual_EN.pdf)
- [3] Hiperface DSL ® Implementation manual; SICK-Stegmann GmbH, donaueschingen, release 1.06, may 31, 2014

Historie revizí

Rev.	Kapitola	Popis změny	Datum	Jméno
0	Všechny	Publikování dokumentu	14.8.2017	D. Uzel