MANUAL

CAN-BUS
for
Digital Servo Controller
DS, DSP, BAMO-D
BAMOBIL-D
BAMOCAR-D


CNC- LABOD electronic

UNITEK Industrieelektronik GmbH

Industriestraße 1
D-71397 Leutenbach 3 - Nellmersbach

Tel.: 07195/9283-0
Fax 07195/928329
email info@unitek-online.de
Http// www.unitek-online.de

Issue V1 1011

Contents

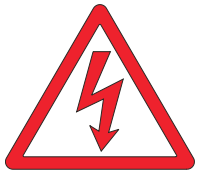
Contents

Contents	Page
Safety instractions	3
General	4
CAN-BUS-connection	5
CAN- format description	6,7
Examples	8..16
Parameter overview	17..

Attention:
The commissioning guide should be used only in conjunction with the DS hardware manual and the software description NDrive.

Safety symbols

Attention danger to life
High voltage



Attention
Important
Warning

The user must ensure:
- that when the unit fails
- when an operator error occurs,
- when the controlling and control units fail, etc..

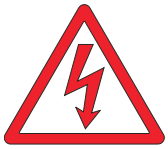
the drive homes to a safe operating state.

Apart from that machines and installations must be provided with monitoring and safety systems independent from the unit. Danger to man and property must be impossible to incur!

Assembly work
- only if all power is cut
- only by qualified professionals

Installation work
- only if all power is cut
- only by qualified electrical professionals
- Comply with safety regulations

Setting and programming work
- only by qualified professionally specialised in electronic drives and software
- observe programming notes
- Comply with safety regulations



CE:
When it is mounted in machines and installations the proper operation of the unit is not permitted until the machine or installation has been approved according to the regulations of the EC machine directive 2006/42/EG, and the EMC directive 2004/108/EG .
Die EC directive 2004/108/EG plus the EMC standards EN61000-2 and EN61000-4 are complied with in the EMC Notes chapter specified installation and test conditions.
A manufacturer's certificate is available.
The manufacturer of the machine or the installation is responsible for the observance of the threshold values prescribed in the EMC regulations..

General

The serial data bus system CAN (Controller Area Network) was originally developed for application in vehicles.
In the mean time ,CAN- BUS is widely used in equipments and machinery. CAN was standardized in ISO 11898.
CAN meets , in particular meets the high security requirements demanded by highly available machines and medical equipments.
High transmission rates and favorable connectivity costs vouch for CAN-BUS. With CAN –BUS, Stations are not addressed here , instead, the content of a message are marked by an unique network wide Identifier .
The identifier also decides the priority of a message.
A good deal of system and configuration flexibility can be achieved.
Hassle-free addition of further devices to the network can be done.

CAN-BUS interface has been incorporated in all digital UNITEK equipments as a **slave** and has been designed to be connected to a CAN-BUS master.
Primary power supply is fed by the Master or through an internal DC/DC converter.

The UNITEK-CAN-Bus can take over the following functions.

From Master (CNC/SPS) to the Slave (DRIVE-DS) (Receive)

Logic functions	Setpoint	**Parameters**	
Enable	Torque setpoint	Control parameters	
Reverence fahrt	Speed setpoint	Settings	
Start, Stop	Position setpoint		
	current limits		

From Slave (DRIVE-DS) to the Master (CNC/SPS) (Transmit)

Logic functions	Actual value	**Parameters**	**Meldungen**
RUN	Torque actual value	Control parameters	Status-message
ENABLE	Speed actual value	Settings	Error message
POS	Position actual value		
Limit switch			

The addresses (REGID) are in the parameter overview page 17-20

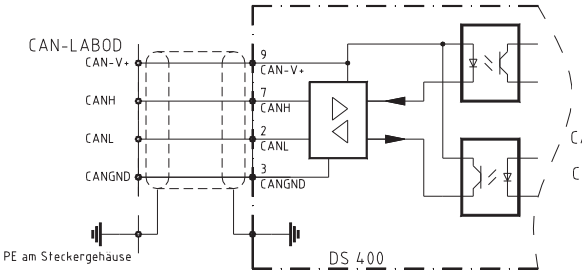
f.e.. Speed reference (SPEED_CMD) = REGID 0x31 <Value in Hex>

CAN-BUS Connections.

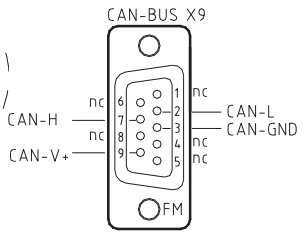
CAN-BUS is the digital connection for CNC- Control (CAN-Master)
CAN-BUS components of LABOD electronics.
Programming and operation through a control panel with CAN-BUS.
Interface nach ISO 11898

Surge impedance	120 Ohm,
Conductor resistance (loop)	160 Ohm/km
Operating capacity(800Hz)	< 60 nF/km

Input circuit



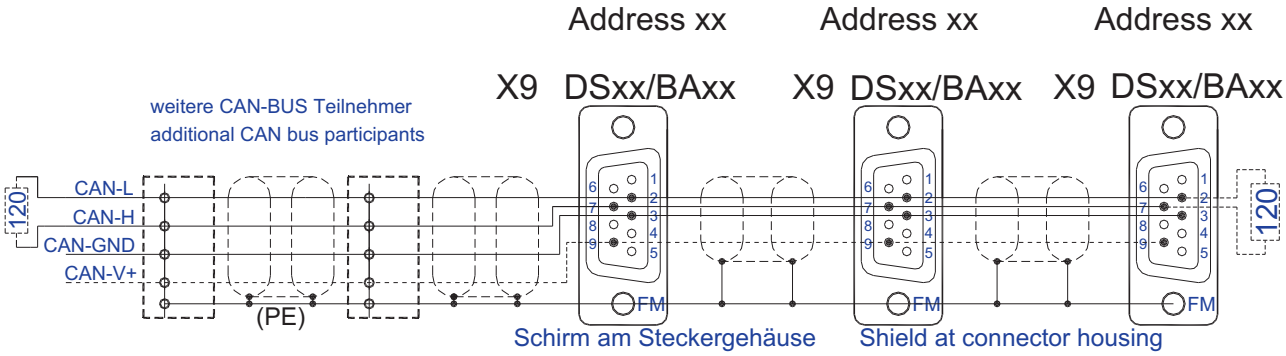
Connector pin assignment X9 (D9p)



cable color (recommend)
LiYCY 4x0.25+Schirm

CAN-V+	9	braun
CAN-GND	3	weiss
CAN-H	7	grün
CAN-L	2	gelb

CAN-BUS connection with several servo amplifiers DS xx, BAxx (slave)



Cable and connection impedance between CAN-H and CAN-L approx. 120 Ohms.
Attention: Please do not connect Can-V+(X9:9) in case of internal CAN-BUS power supply.!

Setting CAN-BUS

Station addresses for transmitting and receiving , as well as transmission rates are set up in the parameter field CAN-Set up in the program PC-DRIVE.

Address	Symbol	Default	Range
Empfangs-Adresse (Slave)	RPDO1	0x201	0x201 bis 0x27F
Sende-Adresse (Slave)	TPDO1	0x181	0x181 bis 0x1FF

Transfer rate NBT	Setting value BTR	Cable length max.	
1000 kBaud	0x4002	20 m	
500 kBaud	0x4025	70 m	
625 kBaud	0x4014	70m	
250 kBaud	0x405c	100m	
100 kBaud	0x4425	500m	

Description of Software format.

The software format is designed for optimal communication with CNC- machines and CAN Modules of LABOD electronics. This format does not correspond to CAN Open Unitek Standard 500 kB/s (LABOD 615kB/S) .
UNITEK Drives can be added as a slave to a CAN Open Network (TPDOI,RPD01).

Number format

Parameter values and parameter numbers are in Little-Endian-Format.
Bit7 to 0, Bit 15 to 8, Bit 23 to 16, Bit 31 to 24

CAN Format for UNITEK-DS-Servo, BAMOCAR-D3 , BAMOBIL-D3 und BAMO-D3.
The CAN-Protocol is a 3 or 5 byte data packet on reception and a 4 or 6 packed data packet on transmission .

The identifier is 11 bit wide. It includes the **COB-identifier**, the **RTR function** (Remote Transmission Request)and the DLC-information (Data Length Code). The first byte from the data field is for the REGID-Index(Parameter number) , the Data of RIGID Index(Parameter value).

CAN Format

Range	Head			Data field				
	COB-ID	RTR	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Funktion	11 Bit	0	Länge	REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24

Master-Slave connection.

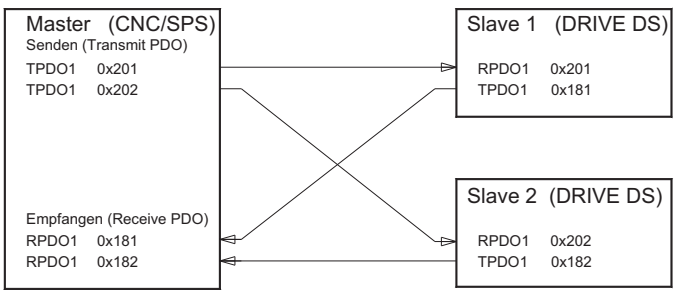
In order to simplify the configuration, a predefined Master/Slave connection set has been specified in CAN open. For networks with one master and up to 127 slaves , this classification of COB-identification offers a simple solution for a CANopen Network. Information distribution originates only from the master. Direct communication between slaves is not possible.

COB - Indentifier										
Functions-Code				Knode-ID						
10			7	6						0
Example 0x181										
0	0	1	1	0	0	0	0	0	0	1
1				8				1		

Favorite Objects (slave)
TPDO1 (0x201 to 0x27F) and RPDO1. (0x181 to 0x1FF)

The objects
TPDO2 .. 4 .. 4 RPDO2 and can also be used.

Connection from master to slave



1. Head field

Range	Head			Data field				
	COB-ID	RTR	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Functions	11 Bit	0	Länge	REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24

1.a COB-ID-Bits (CAN OBJECT ID)

The default value is for CANopen for TPDO1 = 0x181und for RPDO1 = 0x201.

COB-Identifier											Object	
Functions code				Knote-ID								
0	0	1	1	0	0	0	0	0	0	1	TPDO1	0x181-0x1FF
1			8				1					
	1	0	0	0	0	0	0	0	0	1	RPDO1	0x201-0x27F
2			0				1					

This address can be changed , in which a direct station address in the servo amplifier DS4xx for receiving (FORE-CANIDREAD 0 x 68) and for transmitting (FORE-CANIDWRITE 0x69) has been initialized in the field CAN-Setup using a PC through the programming software DRIVE. This address can also be changed through CAN.
The servo amplifier (DRIVE-DSxx) will be dialed with the latest address and the receive ID (REGID 0x68 <Address>) and the transmit –ID (REGID 0x69 <Address>) would be received from DSxx (slave).(Refer Example 1)

1.b RTR-Bit (REMOTE TRANSMISSION REQUEST)

The value for RTR is always set to 0.

1.c DLC-Bits (DATA LENGTH CODE)

With the DLC bits the size of the data field is determined.

Receiving: 0x3 value corresponds REGID plus 2byte (16Bit)
0x5 value corresponds REGID plus 4byte (32Bit)

Transmitting: 0x4 value corresponds REGID plus 2byte plus Dummy (16Bit)
0x6 value corresponds REGID plus 4byte plus Dummy (32Bit)

2. Data field
Data field for receiving in the DS Servo is 3 or 5 Byte wide , whereas it is 4 or 6 Byte wide for transmitting from DS-Servo over the CAN-Bus.

2.a REGID
Up to 254 registers can be defined. The important parameter index are available in the REGID-List.

2.b Data
The data length is specified in the field DLC bits. (16 or 32 bit)
Byte 2 through byte 5 are determined for the 32-bit data register
Byte 2 through byte 3 are determined for the 16-bit data register

Examples of field data
Position setpoint

Position	Num 300010000	0x11E1CA10
Data length		32Bit
REGID-Index for POS-SOLL is		0x6e

Data length
DLC Bits = **5** (for data length 32Bit)

REGID							
0	1	1	0	1	1	1	0
6				e			

Parameter-address
REGID = 0x**6e** (for POS-SOLL)

Data input

Daten D7 bis 0								Daten D15 bis 8								Daten D23 bis 16								Daten D31 bis 24							
0	0	0	1	0	0	0	0	0	1	0	01	1	0	1	0	1	1	1	0	0	1	0	0	0	0	0	1	0	0	0	1
1				0				C				A				E				1				1				1			

Data = 0x**11E1CA10** (corresponding to the position Num. 300010000)

The input is in little-endian format (Intel-Format)

0x = Input in HEX

Examples for CAN-BUS communication

CAN-BUS data reception at DS-Servo.

Stations address at DS-Servo	COB-ID	(default = 0x201)
Data format	DLC	(3,4,5)
Parameter	Data1	(REGID see Parameter list)
Parameter-Content	Data2 bis Data 5	

Examples:

1. Station address change through CAN receive Page 10
2. Controller block (no Release) receive Page 11
3. Rpm- Set value receive Page 11
4. Position set value receive Page 12
5. Torque – Set value receive Page 11
6. Parameter value receive Page 12
7. EEPROM write operation receive Page 13

CAN-Data transmission from DS-Servo over CAN-BUS

Request data transmit from DS-Servo

Request transmit **once**	REGID 0x3d	in case of data 1 input
Parameter-content transmit from	REGID ????	in case of data 2 input
	00	in case of data 3 input
Request transmit **after event**	REGID 0x51	in case of data 1 input.
Mode Bit for CAN-request	REGID 0x10	in case of data 2 input.
	00	in case of data 3 input.
Request **permanent transmit**	REGID 0x51	in case of data 1 input.
Request parameter content	REGID 0x51	in case of data 2 input.
for time information for the repeated timing.	0x??	in case of data 3 input
Abord permanent transmit	Data1,Data2 same, enter for Data3 = 0xff	

Examples :

8. Transmit status once transmitting Page 14
9. Transmit once after event transmitting Page 15-17
10. Permanent transmission of rpm actual value . transmitting Page 18

Information transmit to the CAN-BUS

Input Station addresses at DS-Servo	COB-ID	(default = 0x181) enter

Return back of DLC , parameter- REGID in case of Data 1 , parameter content in case of data 2 to 5

Examples:

11. Routine for simple rpm control Page 19
12. Routine for simple position control Page 19

Example

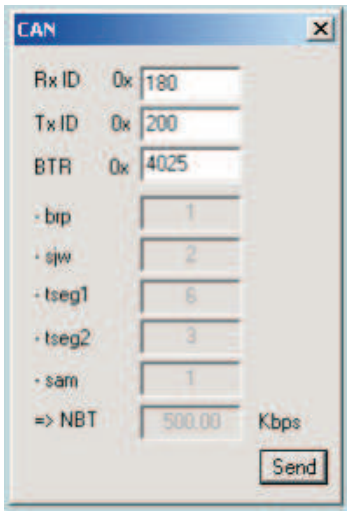
Example 1 : Station address change through CAN.

The address for reception (slave) in a new DS-Servo is 0x201 (default)
This address has to be changed into 0x 210.
The REGID Index for reception ID is 0x 68 ((FORE_CANIDREAD)

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Functions				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example1	**0x201**	**0**	**3**	**0x68**	**0x10**	**0x02**	----	----

Address 0201
RTR 0
DLC 3
REGID 0x68
Data 0x0210

Station address change in the PC program NDrive



Input:

Rx ID Empfangs-Adresse am DS (default 201)
Tx ID Sende-Adresse vom DS (default 181)
BTR Übertragungsrate (Hexwert) s.Seite 5
default 4025 =

Information:
-Brp
-Sjw
-Tseg1
-Tseg2
-Sam
=>NBT Transfer rate (kBaud)

Example 2 Inhibit controller (not enable) received the DS-Servo

Function	Hex value
Station address for receiving (DS-Slave)	201
REGID for inhibit is	0x51 (MODE)
Data length 2 Byte	DLC = 3
Data for inhibit MODE BIT2	0x0004

Range	Head			Data field				
	COB-ID	R	DLC	R	DLC	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example2	**0x201**	**0**	**3**	**0x51**	**0x04**	**0x00**	----	----

Example 3 Speed reference received the DS-Servo

Function	Hex value
Station address for receiving	0201
REGID for Speed reference (SPEED_SOLL) ist	0x31
Data length 2 Byte	DLC=3
Data for 10% Speed Num. 3277 (100%=32767) =	0x0CCD

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example3	**0x201**	**0**	**3**	**0x31**	**0xCD**	**0x0C**	----	----

Example 4 Positions reference received the DS-Servo

Function	Hex value
Station address for receiving	0x201
REGID for positions reference (POSITION_SOLL) is	0x6e
Data length 4 Byte	DLC=5
Data for Position 3000000	0x2DC6C0

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example4	**0x201**	**0**	**5**	**0x6e**	**0xC0**	**0xC6**	**0x2D**	**0x00**

Example

Example 5 Torque reference received the DS-Servo

Function		**Hex value**
Station address for receiving		0x201
REGID for torque reference (TORQUE_CMD) is		0x90=
Data length 2 Byte		DLC=3
Data for 50% torque	Num 16380	0x3FFC

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example5	**0x201**	**0**	**3**	**0x90**	**0xFC**	**0x3F**	**---**	---

Example 6 Parameter setting received the DS-Servo

Funktion	**Hexwert**
Station address for receiving	0x201
REGID for parameter acceleration (Acc-Ramp)	0x35
Data length 2 Byte	DLC=3
Daten for 1000ms acceleration	0x03E8

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example6	**0x201**	**0**	**3**	**0x35**	**0xCD**	**0x0C**	----	----

Example 7 write EEPROM received the DS-Servo

Funktion	**Hexwert**
Station address for receiving	0x201
REGID for EEPROM write	0x84
Data length 2 Byte	DLC=3
EEPROM-Level0	0x0000
(EEPROM-Level1 = 0X0001, Level2 = 0X0002)	

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example7	**0x201**	**0**	**3**	**0x84**	**0x00**	**0x00**	----	----

Example

Transmission from DS-Servo through CAN-BUS

(All examples with station addresses default (0x 201 receive , ox 181 transmit)

Example 8 Status message transmit once from DS-Servo

Function	Hex-value
Station addresses for transmit request is	0x201
REGID for data to be read from DS –Servo and sent to Can (Read) is	0x3d
REGID for status (CORE_ STATUS) is	0x 40

Actual status
Bit 0 Drive release (Enable)
Bit 7 Position control (POS)
Bit 8 Rpm control (SPEE)

Request to send
REGID READ in Data1, REGID KERN_STATUS in Data2, Data3 = 00 (keine Zeiteingabe)

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example8	**0x201**	**0**	**3**	**0x3d**	**0x40**	**0x00**	----	----

Returned Information

Function	**Hex value**
Station addresse for transmit is =	0x181
REGID for Status (KERN_ STATUS) ist	0x40

Actual status

Bit0	Drive enabled	(Enab)
Bit7	Position control	(POS)
Bit8	Speed control	(SPEE)

Data for KERN_STATUS (0x40) are 0x0181

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example8	**0x181**	**0**	**3**	**0x40**	**0x81**	**0x01**	----	

Example 9 After the event send from DS (0x51 - BIT4)

Status message

Function	Hex- value
Station address for transmit request is 0x 210	0x201
REGID for data after event (Event-trigger) read from DS Servo and send to CAN (READ) is MODE BIT (4)	0x51
REGID für MODE BIT 4 ist	0x10

Default value for REGID_MASK-STATUS is 3000
In case of change of status bit 12(CAL) or status bit 13 (Tol) , the complete status message (CORE-STATUS Bit 0 = up to Bit 15) would be sent over the CAN-BUS.

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Funktion				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example9	**0x201**	**0**	**3**	**0x51**	**0x10**	**0x00**	----	----

Returned Information

Aktual status (Status from DS-Servo, 0x40)

Bit 0	Drive enabeld	(Ena)
Bit 7	Position control	(P-S)
Bit 8	Speed control	(S-I)
Bit 12	calibrated	(Cal)

REGID_MASK_STATUS is set to 3000!

Bit 13 position reached, within the positional tolerance (Tol) >>> Event-Trigger
If a change of bit 13 of the core status is sent.

Data for KERN_STATUS (0x40)are 0x3181

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Funktion				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example9	**0x181**	**0**	**4**	**0x40**	**0x81**	**0x31**	----	

Example 9.1 After the selected event send from DS Status-message

With REGID_MASK_STATUS (0x52), the event trigger to change the associated status bits.
for example : REGID_MASK_STATUS = 0x20 equivalent continuous current (Icns)
REGID_MASK_STATUS = 0x12 equivalent limit switch + und - (Lim+, Lim-)

Trigger event to determine REGID_MASK_STATUS

Function	**Hex value**
Station addresseis for the transmission request	0x201
REGID for REGID_MASK_STATUS	0x52
REGID for status-trigger-selection (f.e. Limit switch)	0x12

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example9-1	**0x201**	**0**	**3**	**0x52**	**0x12**	**0x00**	----	----

Transmit status after selected Status-event

Function	**Hex value**
Station addresses for transmit request is	0x201
Read REGID for data after event (Event-trigger)from DS-Servo und an CAN senden (READ) ist MODE BIT (4)	0x51
REGID for MODE BIT 4 is	0x10

Set value for ERGID-MASK_STATUS is 0012
In case if a limit switch is occupied (+ or _) the complete set value for ERGID-MASK_STATUS will be 0012

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example9-1	**0x201**	**0**	**3**	**0x51**	**0x10**	**0x00**	----	**----**

Returned Information

Aktual status (Status from DS-Servo, 0x40)

Bit 0	Drive enabled	(Ena)
Bit 2 oder Bit 3	Limit switch ON	(Lim+ oder Lim-)
Bit 7	Position control	(P-S)
Bit 8	Speed control	(S-I)
Bit 12	calibrated	(Cal)

Data for KERN_STATUS (0x40) are 0x3185 oder 0x 3189

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example9-2	**0x181**	**0**	**4**	**0x40**	**0x85**	**0x31**	----	

Example 9.2 After the event send from DS (0x51 - BIT4)

Error message

Function	**Hex value**
Station address for transmit request is	0x201
REGID for error data after event(event-trigger) to be read from DS-Servo and transmitted to CAN_BUS (READ)	
ist MODE BIT (?)	0x??
REGID for MODE BIT ? (d(error)> CAN is	0x??

In case of an error message , the complete error message (0x8f) (ERR BITMAP1 Bit 0 = up to Bit 15) would be transmitted over the CAN-BUS.

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Funktion				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example9-2	**0x201**	**0**	**3**	**0x??**	**0x??**	**0x00**	----	----

Returned Information

Current error (Error message in the DS-Servo, 0x8f)

Bit 6 Motor temperature (motor-temp)

When changing from one or more bit errors ERR bitmap1 (0x8f) is sent.

Data for ERR BITMAP1 (0x8f) are 0x0040

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example9-2	**0x181**	**0**	**4**	**0x8f**	**0x40**	**0x00**	----	

Example

Example 10 Actual value of rpm to be permanently sent from DS (0x3d)

For a permanent re-transmission , the register REGID_READ will be refreshed with a repetition timing.
In the Byte Data 3, a time information in ms would be input for the re-transmission in Hex. format.

Function	**Hex value**
Hex-value fort he repetition timing 100 ms is the input in Byte Data 3 =	0x64
REGID (Byte 1) for data to be read from DS4xx and sent to CAN (READ) is	0x3d
REGID (Byte2)for speed actual (SPEED_IST) ist	0x30

Transmit request.
REGID READ in Byte 1, REGID KERN_STATUS in Byte 2, Byte 3 = 0x64 (Zeiteingabe)

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Funktion				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example4-2	**0x210**	**0**	**3**	**0x3d**	**0x30**	**0x64**	----	----

Abort transmission:
At Data 3 (Time) >>> enter Hex value 0xFF .

In the time interval of 100ms returned information

Function	**Hex value**
Station addresse for transmit request is	0x201
REGID for speed actual (SPEED_IST) is	0x30
The value for actual speed 100% (Num 32767) is	0x7FFF

Range	Head			Data field				
	COB-ID	R	DLC	Data 1	Data 2	Data 3	Data 4	Data 5
Function				REGID	D7 bis 0	D15 bis 8	D23 bis 16	D31 bis 24
Example4-2	**0x190**	**0**	**4**	**0x30**	**0xFF**	**0x7F**	----	----

Example 11
Speed control
Driving with different speeds and stop

COB-ID	RTR	DLC	REGID	Data1	Data2	Data3	Data4	Data5	Kommentar
201	0	3	3d	e2	00				Empfangen Sendeanforderung BTB
181	0	4	e2	01	00	00			Senden BTB 0xe2
201	0	3	51	04	00				Empfangen Sperre
201	0	3	3d	e8	00				Empfangen Sendeanforderung Freigabe (Hardware)
181	0	4	e8	01	00	00			Senden Freigabe 0xe8
201	0	3	51	00	00				Empfangen Nicht Sperre (Freigabe)
201	0	3	35	f4	01				Empfangen ACC-Rampe (500ms = 0x01f4)
201	0	3	ed	e8	03				Empfangen DEC-Rampe (1000ms = 0x03e8)
201	0	3	31	d4	03				Empfangen Drehzahl-Sollwert 0x31 (30%=0x03d4)
201	0	3	3d	30	64				Empfangen Sendeanforderung Drehzahl-Istwert (alle 100ms)
181	0	4	30	xx	xx	xx			Senden Drehzahl Istwert 0x30 (Wert xxxx alle 100ms)
201	0	3	31	a4	7f				Empfangen Drehzahl-Sollwert 0x31 (100%=7fa4)
201	0	3	31	00	00				Empfangen Drehzahl Null
201	0	3	51	04	00				Empfangen Sperre

Example 12
Reference and driving position
Reference and drive to a position and back to zero position

COB-ID	RTR	DLC	REGID	Data1	Data2	Data3	Data4	Data5	Kommentar
201	0	3	3d	e2	00				Empfangen Sendeanforderung BTB
181	0	4	e2	01	00	00			Senden BTB 0xe2
201	0	3	51	04	00				Empfangen Sperre
201	0	3	31	00	00				Empfangen Drehzahl Null
201	0	3	3d	e8	00				Empfangen Sendeanforderung Freigabe (Hardware)
181	0	4	e8	01	00	00			Senden Freigabe 0xe8
201	0	3	51	00	00				Empfangen Nicht Sperre (Freigabe)
201	0	3	78	01	00				Empfangen Start Referenzfahrt
201	0	3	52	30	00				Empfangen Wert für REGID_MASK_STATUS
201	0	3	51	10	00				Empfangen Sendeanforderung Statusmeldung nach Ereignis
181	0	4	40	xx	xx	xx			Senden Statusmeldung (Wertxxxx)
201	0	5	6e	c0	c6	2d	00		Empfangen Positionsziel 3000000Num
201	0	3	3d	f4	00				Empfangen Sendeanforderung in Toleranz
181	0	4	f4	01	00	00			Senden in Toleranz
201	0	5	6e	00	00	00	00		Empfangen Positionsziel Null
181	0	3	f4	01	00				Senden in Toleanz
201	0	3	51	04	00				Empfangen Sperre