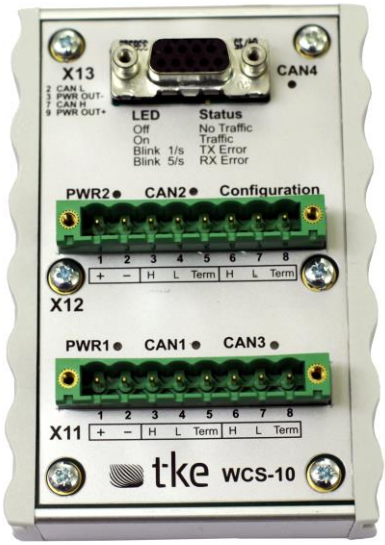
WCS-10
USER MANUAL



WCS-10 manual version 1.41
Firmware version 0x1002b

Marine Type Approvals

info@tke.fi
www.tke.fi

Table of contents

List of tables

List of figures

The following revision history table summarizes changes contained in this document.

Revision History

Date	Revision	Author	Description
31.10.07	1.0	PSAN	New Book
14.11.07	1.1	PSAN	Added object dictionary for BTR0 and BTR1 for all CAN channels
15.11.07	1.2	PSAN	Added examples of bit rates and Sampling point
24.01.08	1.3	BCAB	Renamed CAN-Switch to WCS-10
24.01.08	1.4	BCAB	Changed hardware chapter
25.03.08	1.5	PSAN	Removed objects 0x1001 and 0x1014 from documentation. Changed the logo to TKE logo.
25.03.08	1.6	BCAB	Changed address on front page, added picture on front page
01.04.08	1.7	BCAB	Added more info to HW
25.06.08	1.8	BCAB	Changed first page picture
20.08.08	1.9	BCAB	Changed min max voltage levels
22.08.08	1.10	BCAB	Changed Firmware nr and version on front page
25.09.08	1.11	BCAB	Changed switch in text to WCS-10
07.10.08	1.12	BCAB	Changed text in 4.3 Power supply
13.10.08	1.13	BCAB	LED chapter added
21.10.08	1.14	BCAB	Grounding chapter added, prio added to led chapter
13.11.08	1.15	MCJW	Legal Disclaimer
14.11.08	1.16	BCAB	Changed order of Legal, HW and SW
26.11.08	1.17	PSAN	Changed Firmware version
22.12.08	1.18	BCAB	Changed Firmware version
29.12.08	1.19	PSAN	Changed EMCY-codes to match firmware
29.12.08	1.20	BCAB	Changed help text in EMCY-codes table
30.12.08	1.21	BCAB	Changed EMCY bus state messages and firmware version
05.01.09	1.22	BCAB	Changed EMCY, firmware updated, WCD-10 changed to WCS-10, Changed front page picture.
07.01.09	1.23	BCAB	WCS-10 front page picture changed
19.01.09	1.24	BCAB	WCS-10 front page picture changed
29.01.09	1.25	BCAB	WCS-10 block schema added and mechanical drawing
04.02.09	1.26	BCAB	HW filter example added updated to match firmware 0x1002b. Changed block schema v.1.4, updated connector picture, Technical information added.
04.02.09	1.27	BCAB	GL type approval changed to pending GL type approval
04.02.09	1.28	BCAB	Factory default routing table changed, spelling corrections, baud rate changed to bit rate.
06.02.09	1.29	BCAB	PE ground text changed, HW filter sample text changed
11.02.09	1.30	BCAB	New block schema v.1.5
12.02.09	1.31	BCAB	Halt removed, GL removed from PE grounding
12.02.09	1.32	BCAB	Blink times table added
13.02.09	1.33	BCAB	New block schema v.1.6
05.03.09	1.34	BCAB	New internal terminator picture, and spelling fixes
10.03.09	1.35	BCAB	EMCY error text changed
26.03.09	1.36	BCAB	BRT1 changed to BTR1
31.03.09	1.37	BCAB	X13 pin out changed. Pin 8 added as NC
12.06.09	1.38	BCAB	WCS-10 front page picture changed

12.06.09	1.39	BCAB	GL type approval status changed
19.01.17	1.41	RCS	Front page picture changed to one with TKE logo, spelling fixes

References

/2/ MC9S12XDP512 Data Sheet, Rev. 2.11, Freescale Semiconductor
/3/ CiA301, CiA Draft Standard Proposal 301, V4.1
/4/ Embedded Networking with CAN and CANopen, Pfeiffer, Ayre and Keydel
/5/ CiA302-1, CiA Addition application layer functions, V3.4.1
/6/ CiA Draft Recommendation 303-3 Indicator specification V1.2

Definitions, acronyms and abbreviations

ASAP	As soon as possible
ASCII	American standard for information interchange
CAN	Controller Area Network
CAN-ID	Identifier of the CAN-telegram
CANopen	Higher level CAN-protocol by CiA
CiA	CAN in Automation organization
DLC	Data Length Code
EMCY	CANopen emergency protocol or –telegram
HW	Hardware
LSB	Least significant byte
Mbps	Megabits per second
MSB	Most significant byte
NMT	CANopen network management
OD	Object Dictionary
RX	Receive/reception
SDO	CANopen Service Data Object
SW	Software
TX	Transmit/Transmission

1 Legal disclaimer

Important note! Please read before using WCS-10

1.1 Exclusion of liability

All machines, vehicles or other technical installations, which are controlled by electronic systems can be through disturbing the network communication or other intervention, lead to disorder or failure, which can injure persons or cause material-damage.

Before you connect WCS-10 to such an electronic system, please ensure that connecting the WCS-10 to your system/network will not injure persons or cause material-damage.

You must not use WCS-10 in applications/environments where the use of WCS-10 can directly lead to disorder or failure, of such machines, vehicles or other technical installations or where such failures or damages can lead to injuring of persons.

Do not use WCS-10 if you are not absolutely certain that you know how to use the WCS-10. If you are uncertain about compatibility between WCS-10 and your system, do not use WCS-10 in your system.

TK Engineering Oy does not take over any liability for damages, injuries etc. caused by the use of WCS-10.

2 Hardware specifications

2.1 Technical details

- Operating voltage 10…40V DC.
- Power consumption max 3.5W typical 3W
- CAN routing ports, CAN1 – CAN4, galvanically isolated 1kV, ISO11898, Max 1Mbps
- CAN configuration port, CANA, ISO11898, Max 1Mbps
- Operating temperature range -40°C…+85°C
- Storage temperature range -50°C…+105°C
- Protection class IP30
- Weight ~210g
- Flammability UL94 V0
- GL type approval: Environmental Category D, EMC1, software class 3
  GL type approval certificate number: 7518909 HH

2.2 Mechanical Dimensions

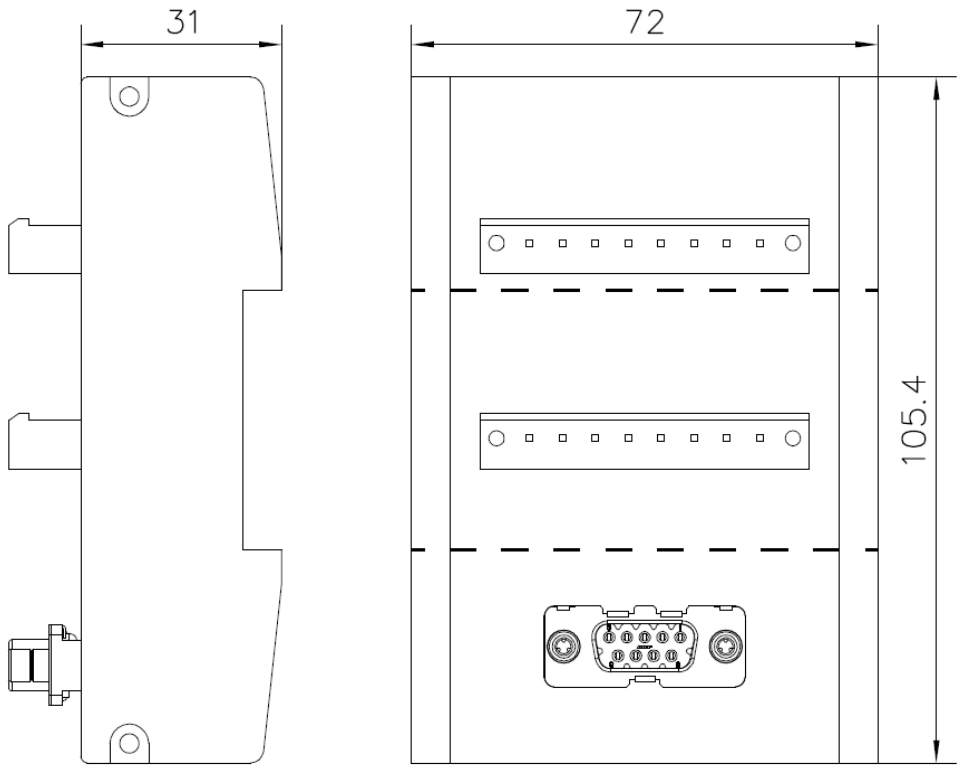


Figure 1: Mechanical dimensions

2.3 Connectors

The WCS-10 has 2 Phoenix MSTBV 2,5/8-GF-5,08-AU connectors and one DSUB 9 pin Female connector Tyco AMP 3-388313-2

Phoenix MSTBV 2,5/8-GF-5,08-AU	
X11	
X11.1	PWR 1+
X11.2	PWR 1-
X11.3	CAN1 H
X11.4	CAN1 L
X11.5	CAN1 Term
X11.6	CAN3 H
X11.7	CAN3 L
X11.8	CAN3 Term

Phoenix MSTBV 2,5/8-GF-5,08-AU	
X12	
X12.1	PWR 2+
X12.2	PWR 2-
X12.3	CAN2 H
X12.4	CAN2 L
X12.5	CAN2 Term
X12.6	CANA H
X12.7	CANA L
X12.8	CANA Term

Tyco AMP 3-388313-2	
X13	
X13.1	NC
X13.2	CAN4 L
X13.3	PWR OUT-
X13.4	NC
X13.5	NC
X13.6	CAN4 CAN_GND
X13.7	CAN4 H
X13.8	NC
X13.9	PWR OUT+
X13.Screen	PE

Crimp Terminal, Ring, 3mm	
X14	
X14.1	PE

Figure 2: WCS-10 connectors

2.4 Block schema

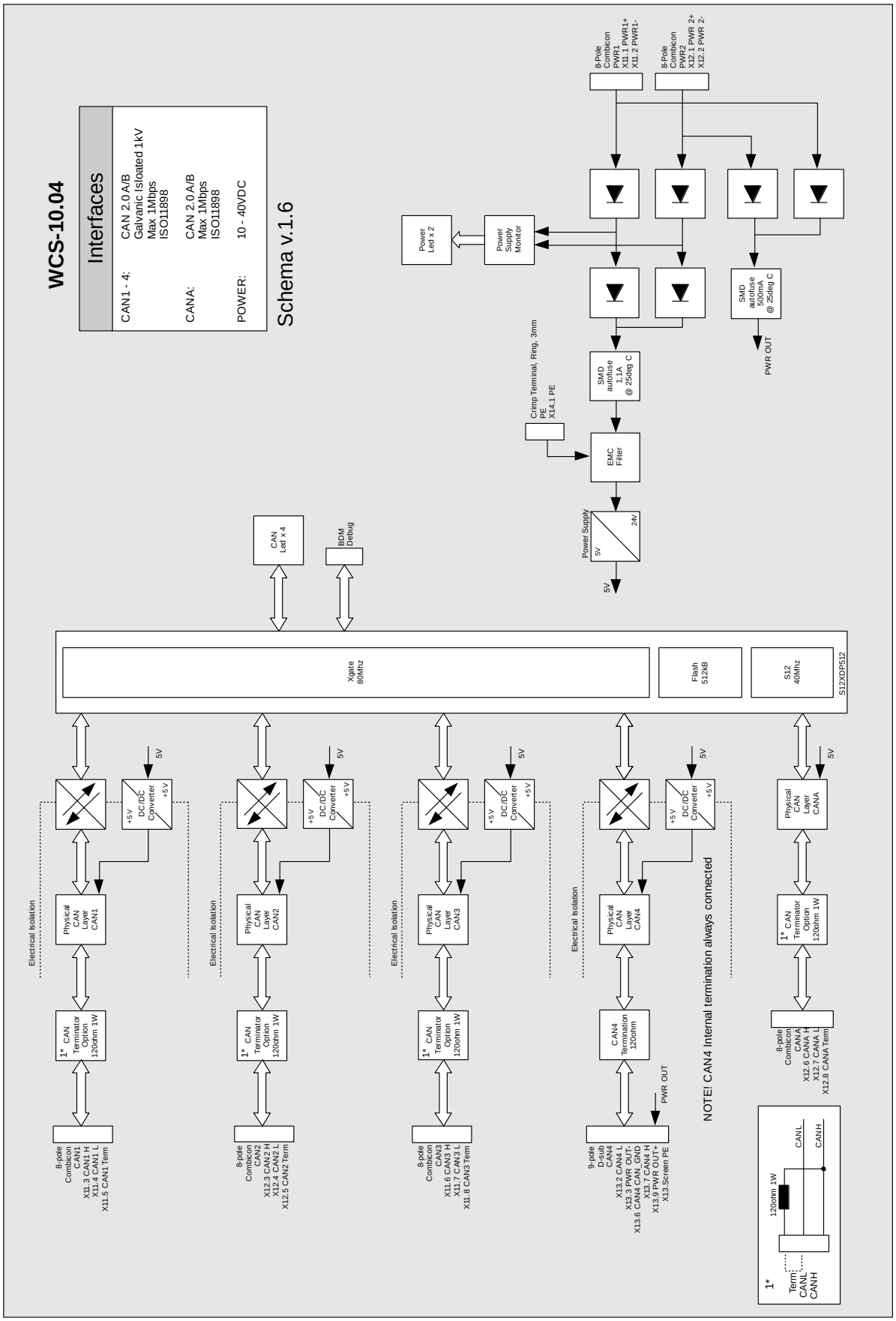


Figure 3: WCS-10.04 Block schema

2.5 Termination

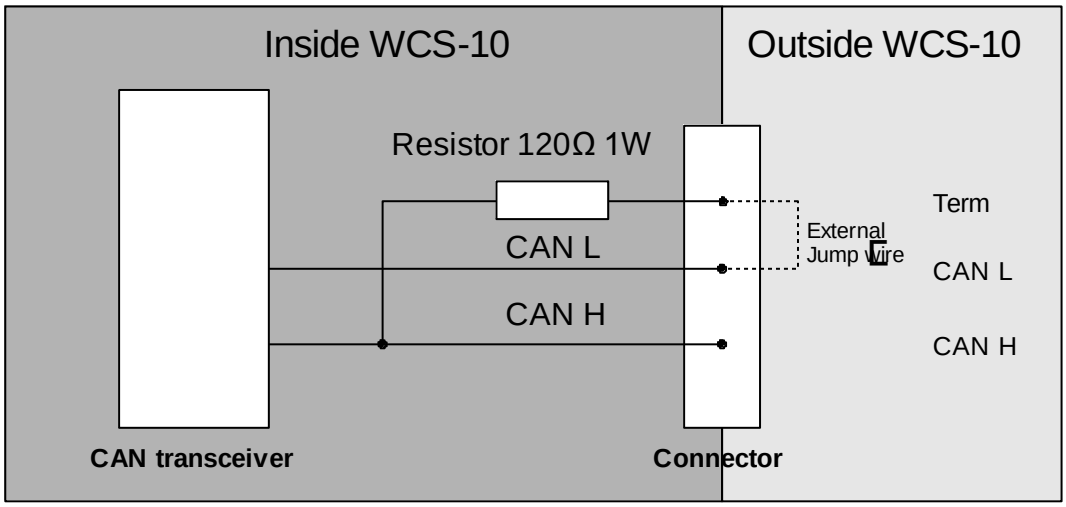


Figure 4: Internal termination

The WCS-10 has internal termination resistors. By installing a "jump" wire from CAN term to CAN L the CAN bus can be terminated. CAN 4 is always terminated at the WCS-10.

2.6 Power supply

The WCS-10 must be powered by DC voltage from 10 to 40 Volt. The WCS-10 has dual power inputs and can be powered on both simultaneous or on only one of the two. The max power consumption when stress tested was 3,5W. In normal operation the power consumption is about 3W.

Table 1: Power supply characteristics

Info	Max	Min	Recommended/Normal
Voltage supply	40VDC	10VDC	12VDC or 24VDC
Power usage	3,5W	2W	3W

2.7 Transceivers

The WCS-10 uses Philips TJA1050 CAN transceivers

2.8 Isolation

Four of the five CAN ports on the WCS-10 are galvanic isolated. The ports that are isolated are CAN 1, CAN 2, CAN 3 and CAN 4. The CAN A port is not isolated. The data lines are isolated with ADuM 1201. DC/DC isolations are done with C&D NTE0505.

2.9 PE Grounding

The WCS-10 X14 ground strip need to be connected to a high quality ground point for WCS-10 to fulfill EMC requirements.

2.10 Light Emitting Diodes (LED)

The WCS-10 have 6 led's, two green and four yellow. The green ones are for input voltage monitoring. When input voltage drops below 10VDC the power led's switches off.

The yellow CAN led's indicates different CAN controller states.

Table 2: Power led status

LED	Status
Off	Input voltage, Low
On	Input voltage, OK

Table 3: CAN controller led status

LED	Status
Off	No traffic
On	Traffic
Blink 1/s	TX Error
Blink 5/s	RX Error

Table 4: Priority of led indication states

Prio	Status
Prio 1	No traffic
Prio 2	TX Buffer overrun
Prio 3	RX Buffer overrun
Prio 4	Traffic

Table 5: Led blink times

State Change	Time
Traffic - Tx Overun – No Traffic	10sec - 10sec
Tx Overun Corrected	2min
Traffic - Rx Overun	10sec
Rx Overun Corrected	2min
No traffic - Traffic	1sec
Traffic – No traffic	10sec

The configuration port does not have any LED indications.

3 Device Overview (WCS-10)

3.1 Introduction

This document describes how to use the WCS-10 and all features. Several constraints has been met with the standard CANGW-firmware, when used as the CAN network switch.

3.2 WCS-10

WCS-10 is a product name for a product originally targeted to connect up to 4 CAN buses running different protocols with each other. The WCS-10 takes care of forwarding CAN-messages, CAN-ID, DLC and data contents according to the configuration. The WCS-10 support bit rate configurations on each CAN-port.

Forwarding software enhanced to the network switch function. The WCS-10 has a powerful processor called X-gate used for software forwarding of telegrams to give high performance.

3.3 CAN-Port usage

The WCS-10 consists of 5 CAN ports. Port 1-4 are used for CAN telegram forwarding. Port A is used for WCS-10 configuration. Port CAN A has implemented Micro CANopen for making the configuration via CANopen SDO telegrams.

3.4 Factory settings

The very first time when the WCS-10 is started up it has hardcoded factory settings in routing table and CAN-interfaces. The factory settings are as follow:

- **Acceptance Filter:** All CAN-interfaces has filter 0xFFFFFFFF. Additional information referenced to /2/ Chapter 13.4.3

- **Identifier Mask Register:** All CAN-interfaces has mask 0xFFFFFFFF. Additional information referenced to /2/ Chapter 13.3.2.18

- **Acceptance Control Register:** All CAN-interfaces has Control Register (IDAM0-IDAM1) value 0x00. Additional information referenced to /2/ Chapter 13.3.2.12
  NOTE: IDHIT0-IDHIT2 are never changed.

- **Bit rate:** CAN1-4 bit rate are defined to 500kbit, CANA initial bit rate is 500kbit.

- **Route table:** All CAN IDs are forwarded to all ports except the port where it was received.

- **CANopen node ID:** 0x7F

- **Heartbeat time:** 1000

- **Operational mode:** The CANopen stack starts in Operational mode.

- **EMCY Inhibit time:** The EMCY inhibit time factor is set to 1. The factor is multiples of 100micro seconds. Ref /3/

Note: The factory configuration for acceptance filer, identifier mask register and acceptance control register behaves as “pass all”.

4 Device Configuration

While the WCS-10 is CAN networking device, configuration is object-dictionary oriented to enable use of standard CANopen configuration tools, EDS- and DCF-files.

4.1 Object dictionary overview

Table 6: Communication profile area

Index	Sub-idx	Type		Description
0x1000	0x00	U32		Device type (0x0000 012D)
0x100B	0x00	U8		Node ID
0x1010		Record		Store parameters
	0x00	U8		Largest sub-index supported
	0x01	U32		Store all parameters (“EVAS” = 0x65 76 61 73)
	0x02	U32		Store communication parameters (0x1000..0x1FFF)
	0x03	U32		Store application parameters (0x2000..0x9FFF)
0x1015	0x00	U16		Inhibit time EMCY (multiple of 100μs)
0x1017	0x00	U16		Producer heartbeat time (multiple of 1ms)
0x1018		Identity		Identity object
	0x00	U8		Largest sub-index supported
	0x01	U32		Vendor-ID
	0x02	U32		Product code
	0x03	U32		Revision number
	0x04	U32		Serial number

Table 7: Manufacturer-specific area

Index	Sub-idx	Type		Description
0x2000	0x00	U8		Heartbeat Enable/Disable
0x5010				CAN1 bit rate
	0x00	U8		Largest sub-index supported
	0x01	U8		CAN1 register BTR0
	0x02	U8		CAN1 register BTR1
0x5011	0x00	U32		CAN1 acceptance filter, bank 0
0x5012	0x00	U32		CAN1 acceptance filter, bank 1
0x5015	0x00	U32		CAN1 acceptance mask, bank 0
0x5016	0x00	U32		CAN1 acceptance mask, bank 1
0x5019	0x00	U8		CAN1 filter configuration
0x5020				CAN2 bit rate
	0x00	U8		Largest sub-index supported
	0x01	U8		CAN2 register BTR0
	0x02	U8		CAN2 register BTR1
0x5021	0x00	U32		CAN2 acceptance filter, bank 0
0x5022	0x00	U32		CAN2 acceptance filter, bank 1
0x5025	0x00	U32		CAN2 acceptance mask, bank 0
0x5026	0x00	U32		CAN2 acceptance mask, bank 1
0x5029	0x00	U8		CAN2 filter configuration

0x5030				CAN3 bit rate
	0x00	U8		Largest sub-index supported
	0x01	U8		CAN3 register BTR0
	0x02	U8		CAN3 register BTR1
0x5031	0x00	U32		CAN3 acceptance filter, bank 0
0x5031	0x00	U32		CAN3 acceptance filter, bank 1
0x5035	0x00	U32		CAN3 acceptance mask, bank 0
0x5036	0x00	U32		CAN3 acceptance mask, bank 1
0x5039	0x00	U8		CAN3 filter configuration
0x5040				CAN4 bit rate
	0x00	U8		Largest sub-index supported
	0x01	U8		CAN4 register BTR0
	0x02	U8		CAN4 register BTR1
0x5041	0x00	U32		CAN4 acceptance filter, bank 0
0x5042	0x00	U32		CAN4 acceptance filter, bank 1
0x5045	0x00	U32		CAN4 acceptance mask, bank 0
0x5046	0x00	U32		CAN4 acceptance mask, bank 1
0x5049	0x00	U8		CAN4 filter configuration
0x5050				CANA Configuration port bit rate
	0x00	U8		Largest sub-index supported
	0x01	U8		CANA register BTR0
	0x02	U8		CANA register BRT1

Table 8: Device-profile specific area

Index	Sub-idx	Type	Bits	Description
0x6800	0x01	U32		Universal ID route settings (Ignored)
	0x02	U16		Array index 0x000 routing
			0..3	Destinations for message received from port 1
			4..7	Destinations for message received from port 2
			8..11	Destinations for message received from port 3
			12..15	Destinations for message received from port 4
0x6801	0x01	U32		CAN-ID x
	0x02	U16		Array index 0x001 routing
			0..3	Destinations for message received from port 1
			4..7	Destinations for message received from port 2
			8..11	Destinations for message received from port 3
			12..15	Destinations for message received from port 4
⋮				
0x6864	0x01	U32		CAN-ID x
	0x02	U16		Array index 0x64 routing
			0..3	Destinations for message received from port 1
			4..7	Destinations for message received from port 2
			8..11	Destinations for message received from port 3
			12..15	Destinations for message received from port 4

4.2 Global parameters

4.2.1 Device type (0x1000)

Device type indicates rough category of the device. WCS-10 equals pure CANopen-device without any device-profile specific functionality or application programmability. No additional information is provided.

Table 9: 0x1000 object description

Attribute	Value
Index	0x1000
Name	Device type
Object code	Variable
Data type	U32
Category	Mandatory

Table 10: 0x1000 entry description

Attribute	Value
Sub-index	0x00
Access	Constant
PDO mapping	No
Value range	U32
Default value	0x0000 012D (DS-301)

4.2.2 Node-ID (0x100B)

Table 11: 0x100B object description

Attribute	Value
Index	0x100B
Name	Node-ID
Object code	Variable
Data type	U8
Category	Optional

Table 12: 0x100B entry description

Attribute	Value
Sub-index	0x00
Access	RW
PDO mapping	No
Value range	0x01..0x7F
Default value	0x7F

Node-ID applies only to the configuration port (port CAN A).

4.2.3 Store parameters (0x1010)

When storing parameters, the device stores the configuration and routing-table in non-volatile memory (flash). The device does not support automated storing. When storing data, the device follows the standard Ref /3/. The value to send is ASCII value for “SAVE”.

Table 13: 0x1010 object description

Attribute	Value
Index	0x1010
Name	Store parameters
Object code	Array
Data type	U32
Category	Optional

Table 14: 0x1010 entry description

Attribute	Value
Sub-index	0x00
Description	Highest sub-index supported
Access	RO
PDO mapping	No
Value range	0x01..0x7F
Default value	0x03
Sub-index	0x01
Description	Save all parameters
Access	RW
PDO mapping	No
Value range	See Table 15
Default value	profile- or manufacturer-specific
Sub-index	0x02
Description	Save communication parameters
Access	RW
PDO mapping	No
Value range	See Table 15
Default value	profile- or manufacturer-specific
Sub-index	0x03
Description	Save application parameters
Access	RW
PDO mapping	No
Value range	See Table 15
Default value	profile- or manufacturer-specific

Table 15: Storage write access signature

Byte3 (MSB)	Byte2	Byte1	Byte0 (LSB)
“E”	“V”	“A”	“S”
0x65	0x76	0x61	0x73

4.2.4 Inhibit time of EMCY (0x1015)

Table 16: 0x1015 object description

Attribute	Value
Index	0x1015
Name	Inhibit time EMCY
Object code	Variable
Data type	U16
Category	Optional

Table 17: 0x1015 entry description

Attribute	Value
Sub-index	0x00
Access	RW
PDO mapping	No
Value range	U16 (multiple of 100µs)
Default value	0x0000

Inhibit time is a minimum time between two consecutive emergency objects produced by node.

4.2.5 Producer heartbeat time (0x1017)

Table 18: 0x1017 object description

Attribute	Value
Index	0x1017
Name	Producer heartbeat time
Object code	Variable
Data type	U16
Category	Mandatory

Table 19: 0x1017 entry description

Attribute	Value
Sub-index	0x00
Access	RW
PDO mapping	No
Value range	U16 (multiple of 1ms)
Default value	0x03E8 (1000ms)

The value for heartbeat time is given in multiples of 1ms. The value 0 disables the producer heartbeat. Dynamic heartbeat time is affected instantly. But in order to keep configuration after reboot a save is needed. Node-ID is affected after reboot.

4.2.6 Device identity (0x1018)

Table 20: 0x1018 object description

Attribute	Value
Index	0x1018
Name	Identity object
Object code	Record
Data type	Identity
Category	Mandatory

Table 21: 0x1018 entry description

Attribute	Value
Sub-index	0x00
Description	Highest sub-index supported
Access	RO
PDO mapping	No
Value range	0x01..0x04
Default value	0x04
Sub-index	0x01
Description	Vendor-ID
Access	RO
PDO mapping	No
Value range	U32
Default value	0x00000209 (TK Engineering Oy)
Sub-index	0x02
Description	Product code
Access	RO
PDO mapping	No
Value range	U32
Default value	0x00000FCB
Sub-index	0x03
Description	Revision number
Access	RO
PDO mapping	No
Value range	U32
Default value	0x00010025
Sub-index	0x04
Description	Serial number
Access	RO
PDO mapping	No
Value range	U32
Default value	No

Table 22: Revision number structure

Byte3	Byte2	Byte1	Byte0
Major revision MSB	Major revision LSB	Minor revision MSB	Minor revision LSB

4.3 Hardware configuration

4.3.1 Disable/Enable Heartbeat

The WCS-10 has a configuration port that is using the CANopen stack specifications. This port uses to configure the node, and it also transmits heartbeat and EMCY messages. This port can disable and enable the heartbeat by writing to SDO object 0x2000. Writing 0 disables the heartbeat, writing 1 enables, which is default.

Table 23: 0x2000 object description

Attribute	Value
Index	0x2000
Name	Disable/Enable Heartbeat
Object code	Variable
Data type	U8
Category	Mandatory

Table 24: 0x2000 entry description

Attribute	Value
Sub-index	0x00
Access	rw
PDO mapping	No
Value range	U8
Default value	1 = Enabled (Default), 0 = Disabled

4.3.2 Bit rates (0x50x0)

Table 25: 0x5010..0x5050 object description

Attribute	Value
Index	0x5010 (CAN1), 0x5020 (CAN2), 0x5030 (CAN3), 0x5040 (CAN4), 0x5050 (CANA)
Name	Port [1..4, A] Bit rate
Object code	Variable
Data type	U8
Category	Optional

Table 26: 0x5010..0x5050 sub 0x01 BTR0 entry description

Attribute	Value
Sub-index	0x01
Access	RW
PDO mapping	No
Value range	See Table 28
Default value	0x01 (500 kbps)

Table 27: 0x5010..0x5050 sub 0x02 BTR1 entry description

Attribute	Value
Sub-index	0x3A
Access	RW
PDO mapping	No
Value range	See Table 28 to Table 34
Default value	0x1C (500 kbps)

These tables below about bit rates are examples and pre-calculated bit rates for the MSCAN controller and with oscillator clock 16 MHz. If you want some other sampling point and bit rates please then contact TK Engineering for re-calculation.

Table 28: Bit rate register enumeration for 1000kbps

Nr Time quanta	Sample Point %	Register BTR0	Register BTR1
8	87,5	0x00	0x1C
8	75	0x00	0x3A
8	62,5	0x00	0x58

Table 29: Bit rate register enumeration for 800kbps

Nr Time quanta	Sample Point %	Register BTR0	Register BRT1
20	85	0x00	0x2F
20	75	0x00	0x4D
20	65	0x00	0x6B

Table 30: Bit rate register enumeration for 500kbps

Nr Time quanta	Sample Point %	Register BTR0	Register BRT1
16	87,5	0x01	0x1C
16	75	0x01	0x3A
16	62,5	0x01	0x58

Table 31: Bit rate register enumeration for 250kbps

Nr Time quanta	Sample Point %	Register BTR0	Register BRT1
16	87,5	0x03	0x1C
16	75	0x03	0x3A
16	62,5	0x03	0x58

Table 32: Bit rate register enumeration for 125kbps

Nr Time quanta	Sample Point %	Register BTR0	Register BRT1
16	87,5	0x07	0x1C
16	76	0x07	0x3A
16	62,5	0x07	0x58

Table 33: Bit rate register enumeration for 100kbps

Nr Time quanta	Sample Point %	Register BTR0	Register BRT1
16	87,5	0x09	0x1C
16	75	0x09	0x3A
16	62,5	0x09	0x58

Table 34: Bit rate register enumeration for 50kbps

Nr Time quanta	Sample Point %	Register BTR0	Register BRT1
16	87,5	0x13	0x1C
16	75	0x13	0x3A
16	62,5	0x13	0x58

The new settings will apply after made a "Save configuration" request and when made a software/hardware reboot.

4.3.3 Acceptance filter configurations first filter bank (0x50x1)

Table 35: 0x50x1 object description

Attribute	Value
Index	0x5011 (CAN1), 0x5021 (CAN2), 0x5031 (CAN3), 0x5041 (CAN4)
Name	Port [1..4] acceptance filter configuration
Object code	Variable
Data type	U32
Category	Optional

Table 36: 0x50x1 entry description

Attribute	Value
Sub-index	0x00
Access	RW
PDO mapping	No
Value range	U32, see Table 37
Default value	0xFFFFFFFF

HW acceptance filter is unsigned 32. See ref /2/ chapter 13.3.2.17 data sheet for details. The values from the CANopen telegram are inserted into the register as follow:

Table 37: Acceptance register mapping

Data Byte	Acceptance Registers
Byte[0]	CANIDAR3
Byte[1]	CANIDAR2
Byte[2]	CANIDAR1
Byte[3]	CANIDAR0

NOTE: The hardware and software uses both banks. The new settings will apply after made a "Save configuration" request and when made a software/hardware reboot.

4.3.4 Acceptance filter configurations second filter bank (0x50x2)

Table 38: 0x50x2 object description

Attribute	Value
Index	0x5012 (CAN1), 0x5022 (CAN2), 0x5032 (CAN3), 0x5042 (CAN4)
Name	Port [1..4] acceptance filter configuration
Object code	Variable
Data type	U32
Category	Optional

Table 39: 0x50x2 entry description

Attribute	Value
Sub-index	0x00
Access	RW
PDO mapping	No
Value range	U32, see Table 40
Default value	0xFFFFFFFF

HW acceptance filter is unsigned 32. See ref /2/ chapter 13.3.2.17 data sheet for details. The values from the CANopen telegram are inserted into the register as follow:

Table 40: Acceptance register mapping

Data Byte	Acceptance Registers
Byte[0]	CANIDAR3
Byte[1]	CANIDAR2
Byte[2]	CANIDAR1
Byte[3]	CANIDAR0

NOTE: The hardware and software uses both banks. The new settings will apply after made a “Save configuration” request and when made a software/hardware reboot.

4.3.5 Acceptance mask configurations first filter bank (0x50x5)

Table 41: 0x50x5 object description

Attribute	Value
Index	0x5015 (CAN1), 0x5025 (CAN2), 0x5035 (CAN3), 0x5045 (CAN4)
Name	Port [1..4] acceptance mask configuration
Object code	Variable
Data type	U32
Category	Optional

Table 42: 0x50x5 entry description

Attribute	Value
Sub-index	0x00
Access	RW
PDO mapping	No
Value range	U32, see Table 43: Acceptance mask register mapping
Default value	0xFFFFFFFF (Don't care all)

HW acceptance mask is unsigned 32. See ref /2/ chapter 13.3.2.18 data sheet for details. The values from the CANopen telegram are inserted into the register as follow:

Table 43: Acceptance mask register mapping

Data Byte	Acceptance Registers
Byte[0]	CANIDMR3
Byte[1]	CANIDMR2
Byte[2]	CANIDMR1
Byte[3]	CANIDMR0

NOTE: The hardware and software uses both banks, and the software is not programmed to work with extended identifiers. The new settings will apply after made a "Save configuration" request and when made a software/hardware reboot.

4.3.6 Acceptance mask configurations second filter bank (0x50x6)

Table 44: 0x50x6 object description

Attribute	Value
Index	0x5016 (CAN1), 0x5026 (CAN2), 0x5036 (CAN3), 0x5046 (CAN4)
Name	Port [1..4] acceptance mask configuration
Object code	Variable
Data type	U32
Category	Optional

Table 45: 0x50x6 entry description

Attribute	Value
Sub-index	0x00
Access	RW
PDO mapping	No
Value range	U32, see Table 46
Default value	0xFFFFFFFF (Don't care all)

HW acceptance mask is unsigned 32. See ref /2/ chapter 13.3.2.18 data sheet for details. The values from the CANopen telegram are inserted into the register as follow:

Table 46: Acceptance mask register mapping

Data Byte	Acceptance Registers
Byte[0]	CANIDMR3
Byte[1]	CANIDMR2
Byte[2]	CANIDMR1
Byte[3]	CANIDMR0

NOTE: The hardware and software uses both banks. The new settings will apply after made a "Save configuration" request and when made a software/hardware reboot.

4.3.7 Hardware filter configuration (0x50x9)

There are 4 different filter configurations which the WCS-10 supports. More detailed information is found in reference /2/ Chapter 13.3.2.12. Filter configuration changes register CANIDAC fields IDAM0-IDAM1.

Table 47: 0x50x9 object description

Attribute	Value
Index	0x5019 (CAN1), 0x5029 (CAN2), 0x5039 (CAN3), 0x5049 (CAN4)
Name	Port [1..4] filter configuration register
Object code	Variable
Data type	U8
Category	Optional

Table 48: 0x50x9 entry description

Attribute	Value
Sub-index	0x00
Access	RW
PDO mapping	No
Value range	U8, see Table 49
Default value	0x00

Table 49: HW filter configuration description

Value	Description
0x00	Two 32-bit registers
0x10	Four 16-bit registers
0x20	Eight 8-bit registers
0x30	Filter closed

The new settings will apply after made a “Save configuration” request and when made a software/hardware reboot.

4.3.8 Filter example for two 32-bit filters using extended ID

The picture below shows the function of one of the two per channel 32-bit filters when using extended CAN IDs. For a filter to produce a “hit” the ID in *CAN controller register* is masked with the value in the *CAN controller acceptance mask register* against the value in the *CAN controller acceptance register*. A “1” in *CAN controller acceptance mask register* is masked as don’t care

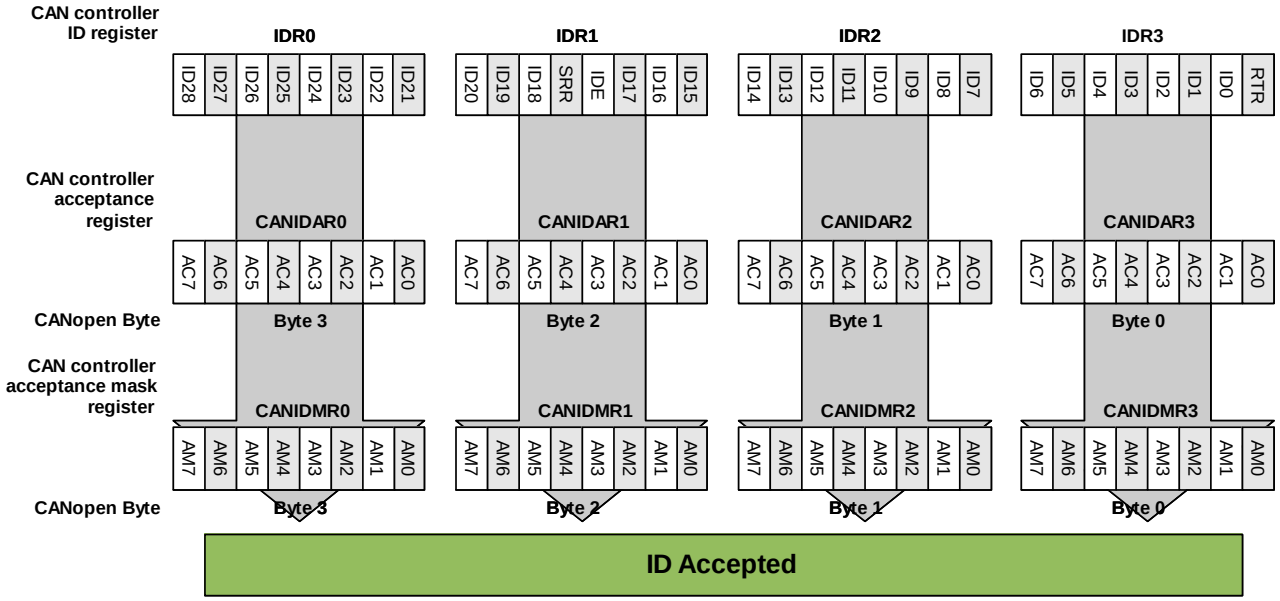


Figure 5: 32-bit filter using extended ID register

The following picture shows the filter settings for only letting trough extended CAN ID 0x1.

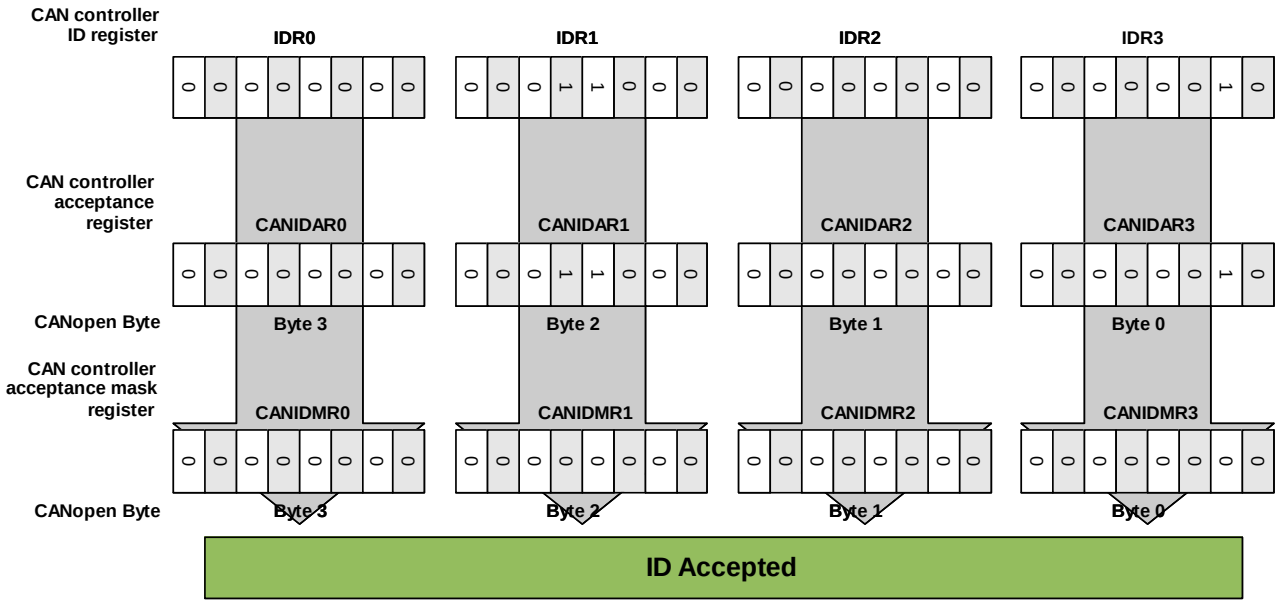


Figure 6: 32-bit filter using extended ID register only accepting extended CAN ID 0x1

To set WCS-10 CAN 1 to use two 32-bit filters with filter one only accepting an extended ID with CAN ID 0x1 and filter two accepting only extended CAN ID 0x2

Table 50: 32-bit filter extended ID example

Index	Value
0x5011	0x180002 (SRR and IDE always 1 in a extended message)
0x5012	0x180004 (SRR and IDE always 1 in a extended message)
0x5015	0x0 (All bits need to match)
0x5016	0x0 (All bits need to match)
0x5019	0x0 (Setting for two 32-bit filters)

Values need to be saved and WCS-10 rebooted for the settings to be taken into account.

4.3.9 Filter example for two 32-bit filters using standard ID

The picture below shows the function of one of the two per channel 32-bit filters when using standard CAN IDs. For a filter to produce a "hit" the ID in *CAN controller register* is masked with the value in the *CAN controller acceptance mask register* against the value in the *CAN controller acceptance register*. A "1" in *CAN controller acceptance mask register* is masked as don't care

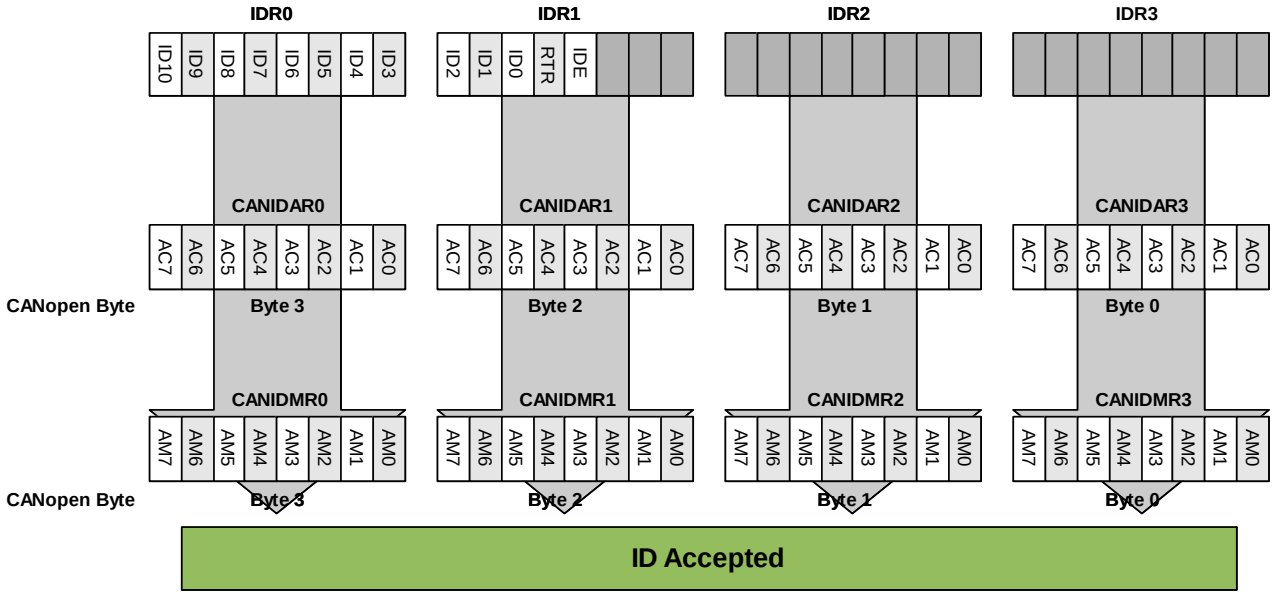


Figure 7: 32-bit filter using standard ID register

The following picture shows the filter settings for only letting trough standard CAN ID 0x1

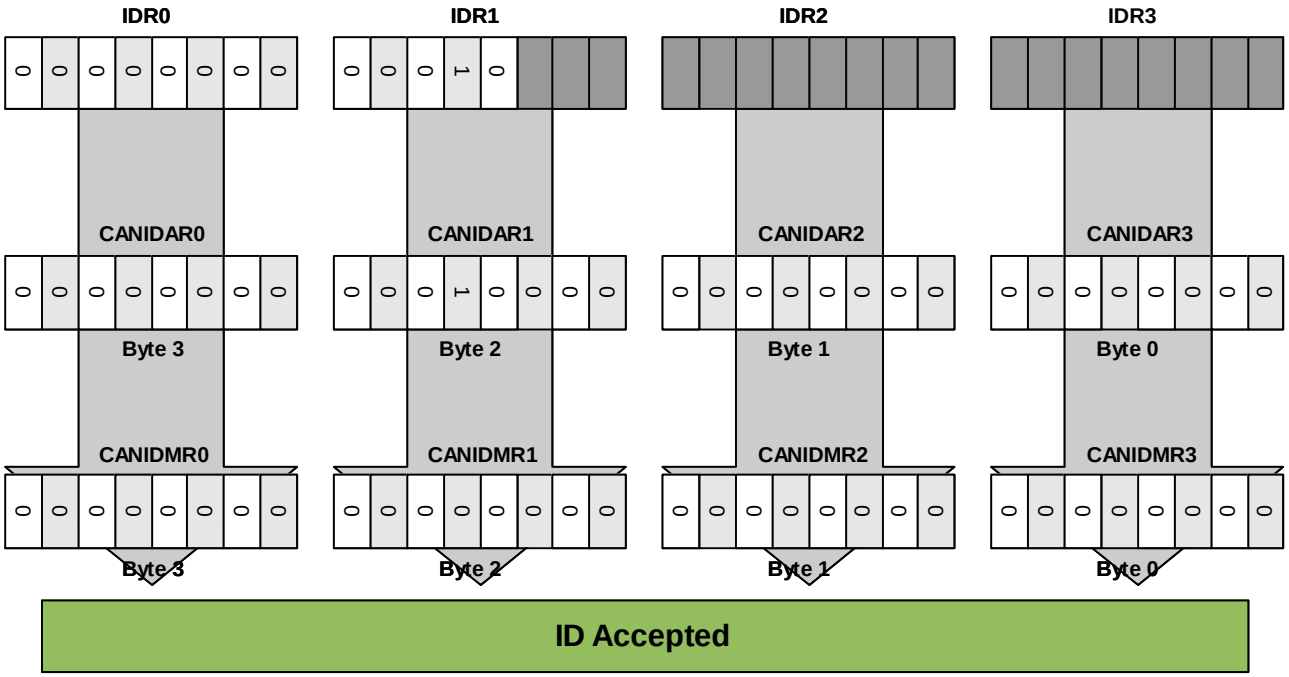


Figure 8: 32-bit filter using standard ID register only accepting standard CAN ID 0x1

To set WCS-10 CAN 1 to use two 32-bit filters with filter one only accepting an standard ID with CAN ID 0x1 and filter two accepting only standard CAN ID 0x2

Table 51: 32-bit filter standard ID example

Index	Value
0x5011	0x100000 (IDE 0 in a standard message)
0x5012	0x200000 (IDE 0 in a standard message)
0x5015	0x0 (All bits need to match)
0x5016	0x0 (All bits need to match)
0x5019	0x0 (Setting for two 32-bit filters)

Values need to be saved and WCS-10 rebooted for the settings to be taken into account.

4.4 Universal ID routing setting

This is used if you want all can-id to have the same routing settings. If value of routing settings is something else than 0x0000 then this routing settings will be used and the routing table 0x6801 to 0x6864 will be ignored.

Table 52: Universal ID routing settings

Attribute	Value
Index	0x6800
Name	Universal ID routing
Object code	Variable
Data type	U16
Category	Optional

Table 53: 0x6800 sub 1 entry description CAN-ID

Attribute	Value
Sub-index	0x01
Description	CAN id NOTE! This ID will be ignored
Access	RW
PDO mapping	No
Value range	U32
Default value	0x00000000

Table 54: 0x6800 sub2 entry description Route

Attribute	Value
Sub-index	0x02
Description	Route description
Access	RW
PDO mapping	No
Value range	U16, see Table 58
Default value	0x7BDE (for all entries)

4.5 Route table configuration (0x6801..0x6864)

If object dictionary index 0x6800 sub index 2 is equal to 0x0000 then this routing table of 100 CAN-ID will be used for routing.

Table 55: Route table Configuration

Attribute	Value
Index	0x6801..0x6864
Name	Route table
Object code	Variable
Data type	U16
Category	Optional

Table 56: 0x6801..0x6864 entry description CAN-ID

Attribute	Value
Sub-index	0x01
Description	CAN id
Access	RW
PDO mapping	No
Value range	U32
Default value	0x00000000

NOTE: Standard id 0x3f and extended id 0x3f are handled separately, to define an extended id, bit 30 in ID field must be set to 1. To define a remote id, bit 31 in ID field must be set.

Table 57: 0x6801..0x6864 sub2 entry description Route

Attribute	Value
Sub-index	0x02
Description	Route description
Access	RW
PDO mapping	No
Value range	U16, see Table 58
Default value	0x0000 (for all entries)

Table 58: Route descriptor structure

Bit15 (MSb)															Bit0 (LSb)
Route descriptor entry for CAN-ID															
Nibble3				Nibble2				Nibble1				Nibble0			
Target flag set to message received from CAN4				Target flag set to message received from CAN3				Target flag set to message received from CAN2				Target flag set to message received from CAN1			
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0: Not forwarded to CAN4 1: Forwarded to CAN4	0: Not forwarded to CAN3 1: Forwarded to CAN3	0: Not forwarded to CAN2 1: Forwarded to CAN2	0: Not forwarded to CAN1 1: Forwarded to CAN1	0: Not forwarded to CAN4 1: Forwarded to CAN4	0: Not forwarded to CAN3 1: Forwarded to CAN3	0: Not forwarded to CAN2 1: Forwarded to CAN2	0: Not forwarded to CAN1 1: Forwarded to CAN1	0: Not forwarded to CAN4 1: Forwarded to CAN4	0: Not forwarded to CAN3 1: Forwarded to CAN3	0: Not forwarded to CAN2 1: Forwarded to CAN2	0: Not forwarded to CAN1 1: Forwarded to CAN1	0: Not forwarded to CAN4 1: Forwarded to CAN4	0: Not forwarded to CAN3 1: Forwarded to CAN3	0: Not forwarded to CAN2 1: Forwarded to CAN2	0: Not forwarded to CAN1 1: Forwarded to CAN1

The switch software is capable of handle all telegrams having 11-bit (standard) or 29-bit (extended) CAN-IDs to be forwarded to one or more interfaces or to be discarded. Route table consists of a Word (16bit) array, length 100. Each Word correspond to a table index between 0 to 100.

Example 1:
0x6801:0x01 = 0x3f: (array index 0)
0x6801:0x02 = 0x0356: (array index 0)
0: Messages with ID 0x3f received from CAN4 are discarded
3: Messages with ID 0x3f received from CAN3 are copied to CAN1 and CAN2
5: Messages with ID 0x3f received from CAN2 are copied to CAN1 and CAN3
6: Messages with ID 0x3f received from CAN1 are copied to CAN2 and CAN3

Example 2:

0x6802:0x01 = 0x2000003f: (array index 0)
0x6802:0x02 = 0x7BDE: (array index 0)
7: Messages with EXT ID 0x3f received from CAN4 are copied to CAN1, 2 and 3
B: Messages with EXT ID 0x3f received from CAN3 are copied to CAN1, 2 and 4
D: Messages with EXT ID 0x3f received from CAN2 are copied to CAN1, 3 and 4
E: Messages with EXT ID 0x3f received from CAN1 are copied to CAN2, 3 and 4

Row 1 (CAN Channel) defines from which channel (CAN-Interface) the telegram is received. Row 2 defines to what channel(s) the telegram is to be forwarded. The forwarding method is always ASAP.

Route table configuration is done by CANopen SDO. OD-index starts from 0x6801, which corresponds to array index 1.

The new settings will apply directly. In order to remain same route configuration after reboot, a "Save configuration" request is needed.

Standard id 0x3f and extended id 0x3f are handled separately, to define a extended id, bit 30 in ID field must be set. To define a remote id, bit 31 in ID field must be set.

5 Error message

The error messages (EMCY) are using CANopen protocol as template. Software is producing tree types of error messages for each CAN interface. The error messages are defined in table below.

Table 59: Error message descriptions

Error code	Error message	Description
0x8119	TX Overrun	This error message occurs when target CAN channel has dropped telegrams, because of overrun. Overrun is when 3-stage HW buffer and 16-stage SW buffer is full.
0x8111	RX Overrun	This error message occurs when the receiving CAN channel 5-stage HW buffer is full.
0x0000	Error-Active	Sent after a error state is cleared and transmission is working without errors.
0x8120	Error-Warning	RX or TX counter >96 and ≤ 127
0x8121	Error-Passive	RX or TX counter >127 and ≤ 255
0x8122	Bus-off	RX and TX counter > 255

In table below the bytes in CANopen telegram are defined.

Table 60: Error message structure

Byte	Description
0	Error code LSB
1	Error code MSB
2	Error-register object (0x1001)
3	Target CAN interface error occurred
4	Error counter, counter of how many times the error occurred LSB
5	Error counter, MSB
6	not used
7	not used