JÖRGENSEN ••

Index

1	Mo	dule Configuration		3
2	Out	put conditions		5
	2.1	Logical Operators	6	
	2.2	Editing condition list	9	
	2.3	Editing specific conditions in the list	10	
3		gram		11
4		gs / Subroutines		
5	Cro	ss reference		14
6	PW	M/Danfoss Configuration		15
7	Ser	vo settings:		18
	7.1	Settings – Servoloop1 and Servoloop2	19	
	7.2	Settings – Servoloop 3 / Analog ut	21	
	7.3	Connections		
8	Spe	cial Features		24
	8.1	Simulate ID	24	
	8.2	Increase/Decrease	25	
	8.3	Counter		
	8.4	PID controllers	27	
9	Pri	nt:		29
10) Pro	gramming module:		30
11	Up	load Program:		32
12	-	alyse the CAN bus:		
	12.1	Analyse - Advanced		
	12.2	Analyse Specific – Bar graph		
	12.3	Analyse Specific – Histogram:		
13	G Cor	n-port		39
14	GS	M		40
15	i Loc	k Project		41
16	6 Pro	gram Settings		42
17		p		
	17.1	Manual		
	17.2	Updates		

Project Information

S CanPro	
File View Settings Insert Help	
🕠 Project information 🏋 Module configuration 🤣 Output conditions 🌷	Programming 🦙 Upload program 😕 Analyse 🍳 Analyse specific
Project information	
Information	eous
Project Name: Example project Rev nr:	
Date: 2008-12-22 V Rev Date:	2008-12-22
Signature:	
Comment	
	<u>^</u>
	×
GSM :	

Project name: Name of the project

Date:	When the project was created.
Rev no:	Revision no
Rev. Date:	Date of the last revision.
Signature:	Signature of the person who last revised the project.
Comment:	Possibility to save some comments about the project.
Flags:	Choose the current language to use.

1 Module Configuration

New Module

🗞 CanPro		
File View Settings Insert Help		
🕦 Project information 🥂 Module configuration 🤌 Output	conditions 🤳 Programming 🦙 Upload pro	gram 🥮 Analyse 🔍 Analyse specific
Module Configuration	8	
Add to project	Registered modules 1 Analog module Module Comment 1	
New Module	2 Digital module Module Comment 2 3 PWM64 module Module Comment 3	Delete Module
Module: Digital		
ID 1-25		
Comment: Module Comment 2		Danfoss
Function Port Comment		
I/O 1 Digital in 💌 Port Comment 1		Danfoss settings
1/0 2 Digital out V Port Comment 2		PWM
1/0 3 Freq. counter V Port Comment 3		PWM settings
1/0 4 -Not Connected 💌		
1/0 5 -Not Connected 💌		Servo settings
1/0 6 -Not Connected 💌		Convert PWM
1/0 7 -Not Connected 💌		
1/0 8 -Not Connected 💌		Saftey settings
Radio ID 1 (0-65535) Register Module		
TimeOut Sec (0,2-9,9) Cancel registration		
	N	
GSM :		

Add a new module (hardware representation) to the current project. Each module must have a ID to identify it on the bus. The lowest free ID will automatically be assigned, if not changed.

Delete the currently selected module from the project. If the Delete Module module is used in the programming logic, a warning will appear before the module is deleted. When a new module has been added and the different ports have Register Module been configured, this button will add the module to the project. The registered module will now appear in the "registered modules" list. PWM settings Show settings window for PWM-ports on the selected module. Danfoss-inställningar Show settings window for Danfoss-ports on the selected module. Shows setting window for the "servo loops" on the selected Servo settings module. Convert between different PWM-hardware. (used for legacy Convert PWM hardware) Show settings window for safety features on the safety module. Saftey settings

Module:	Choose what type of	hardware you want to	add to the project
	Digital module Radio module Servo module	Analogue module PWM-module	PWM-64 module Text module
ID: 2	Sets the ID to be used	d on the hardware mod	lule. (1-25)
Comment:	A short naming of the	e module.	
<i>Function:</i> enabled.	The selected module	has a number of ports	that has to be
enabled.	The different options	follow:	
	Digital module: Digital in Digital out Frequency counter - Not connected -	Analogue me Analogue in Analogue out - Not connect	t
	<u>Radio module:</u> Digital in Analogue in - Not connected -	PWM-modu PWM-out, - not connecte	
	<u>Text-module:</u> Text out - Not connected -	<u>Servo modul</u> Servo out - Not connect	
Port comment:	Each port on a modul E.g. Button1 or LED	e has a short descripti 1	on field.
Radio ID1TimeOut1,0		sed, then you can enter 535). This same code the sender.	1
	lost without the receiver enters the time known value from the	nfigures for how long ver to enter a TimeOu ne out state, it will no e sender, but instead so cted by in the program	t state. When the longer send its last et all values to 0.

Registered modules:

Registered module list shows all modules in the project. The modules are ordered by ID. To edit the settings of a module, the select the module from the list and the corresponding values will show.

2 Output conditions

🗞 CanPro										
File View Settir	ngs Insert	Help								
	6								_	
🔞 Project information 🔢 Module configuration 🤣 Output conditions 🌷 Programming 🦨 Upload program 🗭 Analyse 🍳 Analyse specific										specific
	ut cond	litions								
Module	2 Digital mo	dule Module Comment 2	*	🛛 🐬 Flag Commer	it 📄					
OutPort	📑 2 Digita	al out Port Comment 2	*	•			[Analy	se buss	
Conditions										
	module	ID Type Comment		/O Type Comment	~	is	> 🗸	100	v 🍕	
Active if AND	module	1 Analog module Module Comm 🗸	port	🗲 1 Analog in	~	is	< 🗸	200	~ 2	
OR	Module	2 Digital module Module Comme 💌	port	🗲 1 Digital in Port Comment	-	is	- ~	1	~	
TOGG.	module	*	port		~	is	Y		~ 🍫	
	💌 module	~	port		~	is	~		× 🍫	
	💌 module	×	port		~	is	~		× 🍫	
	💌 module	~	port		~	is	~		× 🍫	
	💌 module	· · · · · · · · · · · · · · · · · · ·	port		~	is	~		× 🍫	
				Cr	iss re	fere	ence	Process	Chart	
GSM :										

Module:	Select what module you want to edit
Out port:	Select on witch port on the selected module you want to program.
Conditions:	The eight rows are used to set the programming logic for the selected port. The programming logic can be built up by using I/Os, timers or by using sub-routines called Flags.

2.1 Logical Operators

AND 🔽	Logical AND operator.
OR 🔽	Logical OR operator. If a list of conditions shall be used with an OR operator, then write the conditions in a FLAG and then use OR on the flag.
SET 💌	Logical SET operator. Sets a value to the port (0-255).
FOLLOW	Logical FOLLOW operator. The current port will follow another port.
LIMIT	Logical LIMIT operator. The current port will follow another port, but with the boundary checks. The operators <i><</i> , <i>> is used to limit the</i> <i>ports values to a constant value</i> . To be able to limit the ports out value to a variable limit (e.g. if the limit value is set by a analogue input) use <i><</i> I/O, <i>></i> I/O.
OR FOLLOW 🔽	Logical ORFOLLOW operator. An extension of the operator FOLLOW. This operator allows for multiple ports to follow, but only one at the time. The ORFOLLOW that has a value that differs the most from the normal center value is the one to be used. (The modules PWM, PWM64 and Danfoss has a center value/default value that can be changed. All other modules have a centre value/default value = 0)
TIMER	Timer 1 has a 0.1 sec resolution and can range from 0.0 - 25.5 seconds. Timer 2 has a 1 sec resolution and can range from 0 - 255 seconds. Puls timer has a range from 0.2-25.5 seconds and a constant duty cycle of 50%.
TOGG. 💌	Logical TOGG. Operator. Toggle/changes between True and False, each time the value is true. Default value is False.
FOLLOW+ 🔽	Logical FOLLOW+ operator. The current port will be assigned the value from another port, and added a constant value. Some module support FOLLO + I/O (see datasheet for support)
FOLLOW- 🔽	Logical FOLLOW- operator. The current port will be assigned the value from another port, and subtracted a constant value. Some module support FOLLO - I/O (see datasheet for support)
FOLLOW.I 💌	Logical FOLLOW INVERTED operator. Same as FOLLOW, but inverted. (255 – value to follow)



Logical LIMIT INVERTED operator. Same as the limit function, but.(255 - value to follow/limit)

In column two in the condition group box you specify what module that should be in the condition.

If the operator TIMER is used you can choose what type of timer to be used.

TIMER	~	modul	Timer1	0,1 s/step (0-25,5 s)	*
	~	modul	Timer2	0,1 s/step (0-25,5 s) 1,0 s/step (0-255 s)	
	~		Puls	0,1 s/step (0,2 25,5 s)	

If the operator SET is used some of the following dropdown boxes will be disabled.

SET	~	مار بام م	Constant Value	*	1	~	:-	=	✓ 123	×
561		module	Constant value		port	· · · · · · · · · · · · · · · · · · ·	IS	L	120	×.

If operator is FOLLOW the two last boxes in this row will be disabled.

MODIUE / TIMED

FOLLOW	💌 module	1 Digital module	~	port 🗲 1 Digital in	*	is	- ~	~

OP I			MODULE / IIWIER			PORT / SUBRUU	UPJ					
	OR	*	module	1 Digital module	*	port 🗲 1 Digital in	*	is	=	*	1	*

DODT / SUDDOUTINES OD

Port Select what port to be used in the current condition. Both in-ports and out-ports can be used. It's is also possible to use subroutines in the condition.

Operator2 (**OP2**)

OD 1

= > < <> Comparison operators, where the later operand (OP3) is a constant. =I/O >I/O <I/O <>I/O. Comparison operators, where the later operand (OP3) is an I/O.

Operand3 (OP3)

Define the constant value or the I/O to be used in the logical condition line. If a constant value is used, the value can range from 0 to 255. If operator 2(OP2) Uses a I/O comparison, then the the I/O is defined as follows in the box (OP3).

Example 1 OP3:	25 corresponds to ID	2 port 5
Example 2 OP3:	236 corresponds to ID	23 port 6

		Opera	tor 2	(OP2	2)
The following examples are TRUE if:				↓ ◆	
Is TRUE if the value on module 1, port 7 is equal to 130					
1 Analog module 🛛 🖌 port 🗲 1 Analog in 🗸	is	=	v	130	*
Is TRUE if the value on module 2, port 1 is less then 5					
2 Analog module v port 🗲 1 Analog in v	is	<	v 5	5	*
Is TRUE if the value on module 2, port 1 is greater then 200					
2 Analog module 🔽 port 🗲 1 Analog in 👻	is	>	v 2	200	¥
Is TRUE if the value on module 2, flagga 1 is anything other than	127	7			
2 Analog module v port 1 Flag/Subroutine1	is	\diamond	¥	127	*
Is TRUE if the value on module 2, port 3 equals the value on mod	ule	10 noi	•t 1		
2 Analog module value on module 2, port 5 equals the value on module 2, port 6 equals the value of module 2, port 6 equals	is	=1/0		104	*
Is TRUE if the value on module 2 port 5 is less than the value on	mo	dulo 1	1 nor	+ 7	
Is TRUE if the value on module 2, port 5 is less then the value on 2 Analog module	is			117	~
Is TRUE if the value on module 2, flag 8 is greater than the value	on	modu	le 4 po	ort 4	
2 Analog module 🛛 🗸 port 🗲 8 Analog in 🗸	is	>1/0	v	44	~
					Δ
Is TRUE if the value on module 2, port 2 is anything other than the port 1	le v	alue o	n mod	lule 1	U
2 Analog module v port C 2 Analog in v	is	<>I/0	v 1	101	*

2.2 Editing condition list

By right clicking on the spanner icon at the right of the "Out port box", a menu will show you the following:

Cut condition list

This will cut out all the conditions shown, and place it in the memory.

Copy condition list

Copy all conditions shown, and place them in the memory.

Paste condition list

Paste the condition list currently saved in the memory.

Be aware that if a condition uses flags, the flag on one module do not correspond to flags on other modules. All flags are local to the module and you will get a warning.

Clear condition list

Clear all conditions on the selected port.

OutPort	🕂 1 Digital out	Cut condition list
Conditions		🚯 Copy condition list 🚺 Paste condition list
	ID Type Comment	V 🗱 Clear condition list

2.3 Editing specific conditions in the list

By right clicking on the spanner icon at the right of the condition row, a menu will show you the following:

Cut condition

This will cut out all the conditions shown, and place it in the memory.

Copy condition

Copy all conditions shown, and place them in the memory.

Paste condition

Pastes the conditions list currently saved in the memory.

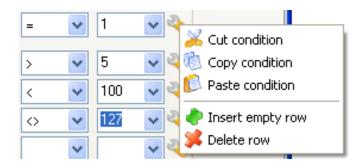
Be aware that if a condition uses flags, the flag on one module do not correspond to flags on other modules. All flags are local to the module and you will get a warning.

Insert empty row

This will insert an empty row into the condition list.

Delete row

Clear all conditions on the selected port.



3 Diagram

When programming in *CanPro* you can use a process chart over the program logic. This can make it easier to get an overview of the functions. It's also possible to print the process chart to a printer.

Here is an example of a program and the diagram connected to this condition list.

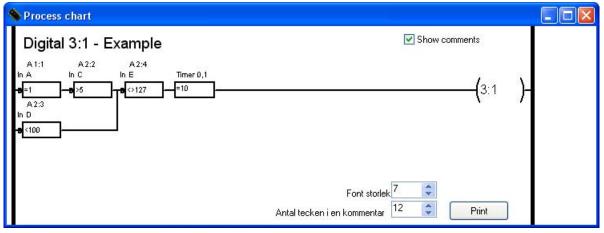
🗞 CanPro									
File View Settine	gs Insert	Help							
0 🖻 🖬 🍯									
Project informa	ation 🔢 I	Module configuration 🤣 Outp	ut conditions	🌷 Programming 🧳	👇 Upload pi	ogra	am 😕 Ana	alyse 🍳 Analys	e specific
	ut conc	litions							
Module [3 Digital mo	dule	*	🛛 🐬 Flag	g Comment				
OutPort		al out Example	~	4				🗌 Analyse buss	
Conditions									-
		ID Type Comment		1/O Type Com	ment				
Active if	module	1 Analog module	port	🗲 1 Analog in In A	*	is	= 🗸	1 💌 🏊	
AND	🖌 module	2 Analog module	Port	🗲 2 Analog in In C	*	is	> 🗸	5 💌 🌭	
OR	🞽 module	2 Analog module	💌 port	🗲 3 Analog in In D	*	is	< 💌	100 💌 🌭	
AND	🖌 module	2 Analog module	yort 🗹	🗲 4 Analog in In E	*	is	◇ ✓	127 🔽 🍑	
TIMER	🖌 module	Timer1 0,1 s/step (0-25,5 s)	yort		*	is	= 🗸	10 💌 🌭	
	🖌 module		yort 🔁		*	is	~	× 🌭	
	🖌 module		yort 🔁		*	is	~	💽 💌 🌭	
	🖌 module		yort		*	is	~	🔹 🗸	
120					Cross	refer	ence	Process Chart	
									,
GSM :						_			
don't						_			344

Diagram without the checkbox show comments disabled.

Services chart		
Digital 3:1 - Example	Show comments	
A 1:1 A 2:2 A 2:4 Timer 0,1 a =1 b >5 b (>127 c =10 c -10 A 2:3 b (<100 c -10 c -	(3:1) Font storlek 7 Antal tecken i en kommentar 12 Print	

Diagram with the checkbox *show* comments enabled.

The fonts and text length can be manipulated with the two boxes, *font size* and *max no of char in a comment*.



4 Flags / Subroutines

将 Flag Comment

By first selecting a module, you can get an overview of all flags/subroutines in a module. You can then edit the name of the different flags/subroutines.

All flags/subroutines are local to the module, and the information in a flag is not shared over the CAN-bus to other modules.

💊 Flag							
	Flags						
7	Modules 💈	2 Digital mod	ule MM1	~	-		
					-		
Flag 1	Aktiverar V1	Flag 17	LastningSJAutoOF				
Flag 2	Aktiverar V1	Flag 18	V1>Ramp]			
Flag 3	V1 (boggi lås)	Flag 19	V1>Svartjord				
Flag 4	V1>Tankvagn	Flag 20	V1>Skivmyllare]			
Flag 5	BomUpp SvängV	Flag 21	V1>24m Ramp]			
Flag 6	BomUpp StickaUpp	Flag 22	V1>24m Ramp				
Flag 7	BomUpp	Flag 23]			
Flag 8	BomUpp SvängH	Flag 24	isAutoUtOn				
Flag 9	SpridninglAuto	Flag 25	RegleringFörF26]			
Flag 10	Loss.AutoRampUT	Flag 26	ManRampKörning				
Flag 11	Lossn.AutoDelyUT	Flag 27	RampAutoUt]			
Flag 12	Loss.AutoDelyUT	Flag 28	RampAutoIn				
Flag 13	Last.AutoRampIN	Flag 29	SJM_Start]			
Flag 14	Last.AutoSM	Flag 30	SJM_STOP				
Flag 15	LossAutSJPå	Flag 31	SM_START				
Flag 16	LossAutoSM	Flag 32	SM_STOP]			
							Close

5 Cross reference

The cross reference window is a good tool, when you want to know where in the program a specific I/O is used.

Scross reference	
Cross reference	Show all ports
ID 3:6 (PDAKonstantvärde) is used as an output condition in : ID 17:5 Ramp In ID 17:6 Ramp Ut	•
ID 8:1 (SIMBoggieGivareOK) is used as an output condition in : ID 10: Flag 11 V_Manuell vidFel ID 10: Flag 12 H_Manuell vidFel	
ID 8:2 (SIMBoggieGivareFel) is used as an output condition in : ID 10:5 SVÄNG VÄNSTER ID 10:6 SVÄNG HÖGER ID 24: Flag 19 Bogg Autostymin	
ID 8:3 (SIMBoggieFörLångsam) is used as an output condition in : ID 9:6 BoggiFörLångsam ID 11: Flag 26 BoggieGivareFel ID 11: Flag 27 BoggieFörLångsam ID 24: Flag 19 Bogg Autostyrnin	
ID 9:1 (MM3VINK.GIVARE VAGN) is used as an output condition in : ID 9:4 BoggiGivareOK ID 9:5 BoggiGivareFEL ID 10:5 SVANG VÄNSTER ID 10:6 SVÄNG HÖGER ID 9: Flag 3 HögerHyst ID 9: Flag 4 VänsterHyst ID 10: Flag 2 Höger hyst ID 10: Flag 2 Höger hyst	
Update Print	Close

By checking the box *Show all ports*, all ports will show in the list. Not only referenced ports but also ports that are not used in other ports/flags will show.

By checking the box *Show all flags*, all flags will show in the list. Not only referenced flags but also flags that are not used in other ports/flags will show.

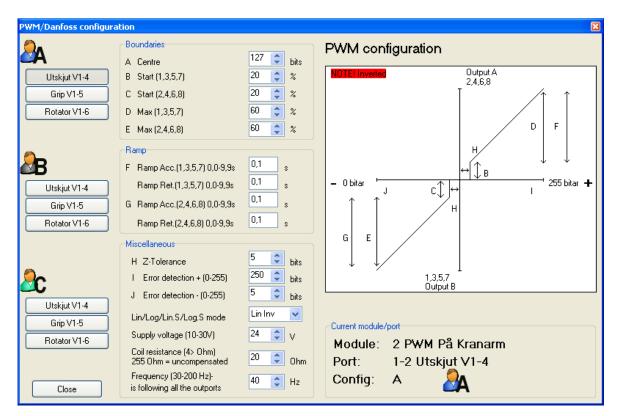
Update

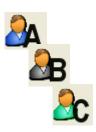
This button will update the list. This could be useful if the window is open at the same time as the program is being written.

Print

Print the information listed in the window.

6 PWM/Danfoss Configuration





3 different user settings can be programmed. For example if different users wants different characteristics on the machine. It can also be used for different modes different modes, where you have a fast, medium and slow mode.

If flag 30 is true on the current module, then setting A is used. If flag 31 is true on the current module, then setting B is used. If flag 32 is true on the current module, then setting C is used.



With the buttons to the left, you can choose what port to configure.

CanPro 4.35 – Documentation

Boundaries		
A Centre	127 文 bits	In signal centre value (e.g. Joystick center)
B Start (1,3,5,7)	20 🛟 %	Start current for proportional valves (min
C Start (2,4,6,8)	20 📚 🕺 🦟	flow)
D Max (1,3,5,7)	60 🛟 %	,
E Max (2,4,6,8)	60 🛟 🐒 🔪	Max current for proportional valves (max
		flow)
		,
Ramp		
F Ramp Acc.(1,3,5,7) 0,0-9,9s	0,1 s	
Ramp Ret.(1,3,5,7) 0,0-9,9s	0,1 s	Separate acceleration and "negative
G Ramp Acc.(2,4,6,8) 0,0-9,9s	0,1 s	acceleration" settings in both directions. Range 0.0 - 9.9 seconds.
	0.1	Runge o.o 5.5 seconds.
Ramp Ret.(2,4,6,8) 0,0-9,9s	0,1 \$	
Miscellaneous		
H Z-Tolerance	5 😂 bits	Dead band on e.g. a joystick.
I Error detection + (0-255)	250 😂 bi <u>ts</u>	Defines what is a valid value. This is
J Error detection - (0-255)	5 🛟 bits_	used to detect cable faults.
Lin/Log/Lin.S/Log.S mode	Lin Inv 🛛 🔽	
Supply voltage (10-30V)	24 🗘 V	——— Characteristics of the output signal.
Coil resistance (4> Ohm)	20	The systems nominal supply voltage.
255 Ohm = uncompensated	20 🛟 Ohm	
Frequency (30-200 Hz)- is following all the outports	40 文 Hz	Nominal resistance in the proportional valve
		Ripple frequency (Used on all ports on
		the module)

Lin/Log/Lin.S/Log.S mode	Lin Inv 🗸 🗸
	Lin.
	Log.
	Lin.S
	Log.S
	Lin Inv
	Log Inv Lin.S Inv
	Lin.S Inv
	Log.S Inv

Lin / Log / Lin.s / Log.s / Lin Inv / Log Inv / Lin.s Inv / Log.s Inv Logarithmic or linier scale, dual or single proportional, inverted signal. The image to the right in the program will illustrate the different signal characteristics.

<i>Lin</i> Linier output.	Two valves for one section.
---------------------------	-----------------------------

- *Log* Logarithmic output. Two valves for one section. More precise precision at lower speed, and faster response at higher speed.
- *Lin.s* Linear output. One valve with a directional valve.
- *Log.s* Logarithmic output. One valve with a directional valve. More precise precision at lower speed, and faster response at higher speed.

Same characteristics as above, but the input signal is inverted. This is useful for example if for example right and left should change when the driver turns his seat around in the vehicle. The invert function mirrors the <u>in signal</u> so therefore it is very important that the centre value is nearly 127bit.

- *Lin Inv* Linier output. Two valves for one section. Inverted signal.
- *Log Inv* Logarithmic output. Two valves for one section. More precise precision at lower speed, and faster response at higher speed. Inverted signal.
- *Lin.s Inv* Linear output. One valve with a directional valve. Inverted signal.
- *Log.s Inv* Logarithmic output. One valve with a directional valve. More precise precision at lower speed, and faster response at higher speed. Inverted signal.

7 Servo settings:

A servo module consists of two feedback loops, with an additional out signal. The first servo loop uses PWM, whiles the other loop can choose between PWM, H-Bridge(actuator).

The third out port can be used in one of the following:

- PWM: 0-100%
- Voltage: 0-5V
- Current: 4-20mA
- PWM@533Hz: 8-92% duty cycle

Servo			Servo	
Servo Configuration	PWM-freque	ency 125 Hz	Servo Configuration	PWM-frequency 125 Hz
Servoloop 1 Servoloop 2 Servoloop 3/ Analog ut			Servoloop 1 Servoloop 2 Servoloop 3/ Analo	g ut
Type : Locked to PWM			Type PwM	×
Input signal B C Setup A B C Min 15 15 15 Center 127 127 127 Max 240 240 240 240 Start 10 10 10 10 80 80 80 80 80 80 \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ <th>Feed back signal</th> <th>Center enabled</th> <th>Input signal B C Setup A B C Min 15 15 15 Center 127 127 127 Max 240 240 240 Start 10 10 10 Stop 80 80 80</th> <th>Center 127 Center enabled</th>	Feed back signal	Center enabled	Input signal B C Setup A B C Min 15 15 15 Center 127 127 127 Max 240 240 240 Start 10 10 10 Stop 80 80 80	Center 127 Center enabled
Control parameters	Error detection		Control parameters	Error detection
Z Start 6	Error detection Feed back signal	Action	Z Start 6	Error detection Action Feed back signal Emergency mode
Z Stop 3 🛟	reed back signal		Z Stop 3 🛟	
P 30 🗘	✓ Input signal	 ○ Min ○ Center ○ Max ○ Freeze 	P 30 🗘	│ Min └ Input signal │ Center │ Max ○ Freeze
		Close		Close
Servo Servo Configuration Servaloop 1 Servaloop 2 Servaloop 3/ Analog ut	PwM-freque	ncy 125 Hz		
Type H-Brygga				
Input signal B C Setup A B C Min 15 15 15 Center 127 127 127 Max 240 240 240 240 Start 10<	Feed back signal	Center enabled		
Control parameters	Error detection			
Z Input 6	Error detection Feed back signal	Action		
Z Feedback 3 🛟	E i coa baok signal			
	✓ Input signal	 Min O Center Max Freeze 		
		Close		

7.1 Settings – Servoloop1 and Servoloop2

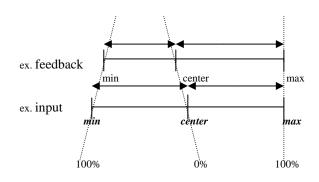
Start, Max

The PWM output signal is 0-100% when not reduced.

(To limit an actuators working span/length, use *feed back signal min*, and *feedback signal max*)

Min, Center, Max for both input signals and feedback signals

Configure the working ranged for the input signal and feedback signal using *max* and *min*. If a center position exits, enable the checkbox *center enable*. For example, a steering wheel has a center, but a accelerator pedal does not have a center. If *min* is larger than *max* the function will be inverted.



Z, Start, Stopp

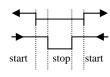
Z-Start sets how big the difference between the input signal and the feedback signal is when a movement starts. If the difference is smaller then Z-start, nothing will happen.

Z-Stop set how big the difference between the input signal and the feedback signal should be before stopping the current movement.

If the system seams nervous, and never find a idle position, the *Z*-*start*/*Z*-*stop* parameters can be changed to get the system to stop within a correct value. The max current for the actuator/valve have also relevance to these settings.

The Z Start should be greater or equal to Z Star

The *Z*-Start should be greater or equal to *Z*-Stop.



P,I

These parameters is only used with PWM.

P is the amplification of the error offset.

I is how fast the movement should accelerate if the error/offset remains constant.

As long as the error/offset is present, I will be added to the movement, each 25 ms. The longer a error/offset is present, the actuator/valve will becom stronger/faster to correct the error/offset.

The resolution of the PWM signal is 460800/PWM-frequency

At 125 Hz the resolution equals 3686 steps.

Example **P**. At 20 bits offset the correction should be at full power. 3686/20 = 184. Example **I**. The acceleration should go from 0 to 100% in 1 second. I= (3686*0,025)/1 = 92

Error detection input:

Error detection can be individually activated for both the input signal and the feedback signal. An error is detected when a value goes outside a given boundary. This boundary is defined as 10 bits under *min*, and 10 bits over *max*.

If the *min* value is smaller then 10 the erroneous value is set to 0.

If the max value is set to 245 or larger, the erroneous value is set to 255.

If an error is detected on the input signal, it's possible to choose how the output signal will behave.

The different modes are *min*, *max*, *center*, or *freeze*. This action will also start if the CAN-communication goes down.

Error detection feedback:

When an error is detected on the feedback signal (H-bridge or PWM) it's possible to enter an emergency mode. In emergency mode the 30 bits closest to max and the 30 bits closest to min will activate the output in that direction, for at most 20 seconds at a time.

Example: The feedback signal is missing on a boat with a steering wheel controlling a rudder. To steer right in emergency mode, the steering wheel must be turned to the 30 last bits at the right direction. This will activate the rudder to steer right as long as you keep the steering wheel in this position, but as most for 20 seconds. When the steering wheel is turned against the center the rudder will stop turning and remain at its current position.

PWM-Frequency

Shared PWM frequency on both servoloop1 and servoloop2.

7.2 Settings – Servoloop 3 / Analog ut

The following output input signals can be used on

- H-bridge
- Voltage: 0-5V
- Current: 4-20mA
- PWM@533Hz: 8-92% duty cycle

Min, Center, Max for both input signals and feedback signals

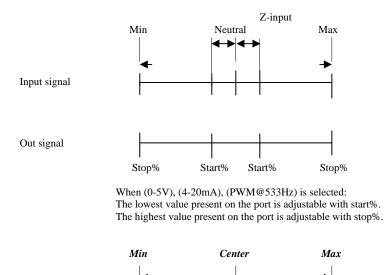
Configure the working ranged for the input signal. If H-bridge is used the feed back signal is configured with min and max. If a center position exits, enable the checkbox *center enable*.

For example, a steering wheel has a center, but a accelerator pedal does not have a center. If *min* is larger than *max* the function will be inverted.

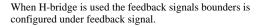
Z input is the deviation in bits from the neutral bit value before more throttle is given than the idle throttle.

Z-Feedback configures how sensitive the regulator is (H-bridge).

(To low value can give a nervous system, to high can give a late start on the regulator)



Feedback Signal



Error detection input:

Error detection can be individually activated for both the input signal and the feedback signal. An error is detected when a value goes outside a given boundary. This boundary is defined as 10 bits under *min*, and 10 bits over *max*.

If the *min* value is smaller then 10 the erroneous value is set to 0.

If the max value is set to 245 or larger, the erroneous value is set to 255.

If an error is detected on the input signal, it's possible to choose how the output signal will behave.

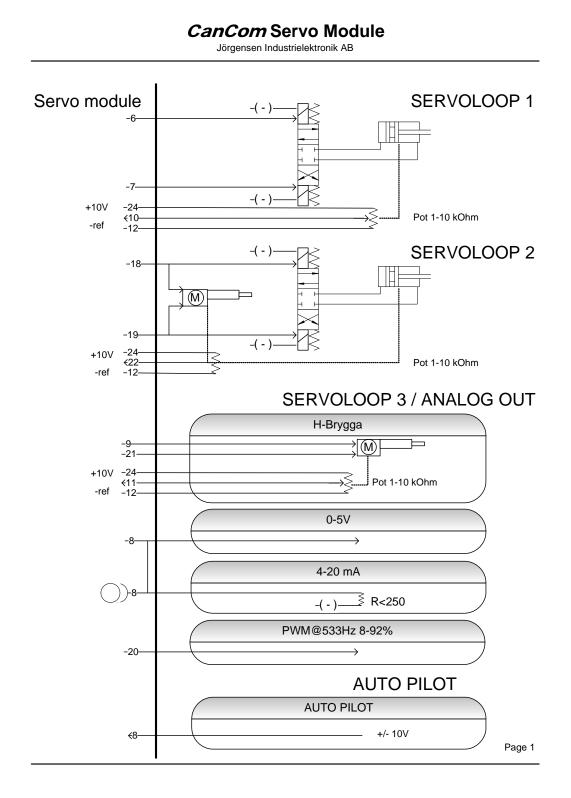
The different modes are *min*, *max*, *center*, or *freeze*. This action will also start if the CAN-communication goes down.

Error detection feedback:

When an error is detected on the feedback signal (H-bridge or PWM) it's possible to enter an emergency mode. In emergency mode the 30 bits closest to max and the30 bits closest to min for the input signal will activate the output in that direction as on/off, for at most 20 seconds at the time.

Example: The feedback signal is missing on a boat with a steering wheel controlling a rudder. To steer right in emergency mode, the steering wheel must be turned to the 30 last bits at the right direction. This will activate the rudder to steer right as long as you keep the steering wheel in this position, but as most for 20 seconds. When the steering wheel is turned against the center the rudder will stop turning and remain at its current position.

7.3 Connections



8 Special Features

When a module is selected that have special features, the button special features will be enabled. When the button is clicked a new window will show the configuration possibilities.

8.1 Simulate ID

Below is an example of the function "Simulate ID". To simulate an ID on the CAN bus you first need to enable the function and assign an ID number to the simulated module. After this has been done, the eight ports of the simulated module must be connected to a value. Possible values to connect is the internal flags (sub routines) that the current module have.

🗞 Special features		
Special features, ID:1		
Simulate ID		
Only for:		
Digital64 (Multimodul-64) PWM64 (Multimodul-64)		
-Simulate ID-		
Simulate values from flags to a :	simulated ID	
Simulate on ID	5	
Simulated port 1 follows:	-// 1	¥
Simulated port 2 follows:	1 2	*
Simulated port 3 follows:	47 3	*
Simulated port 4 follows:	4 4	*
Simulated port 5 follows:	475	*
Simulated port 6 follows:	476	¥
Simulated port 7 follows:	47 10	¥
Simulated port 8 follows:	🍠 64	~
		Close
👫 Module configuration 🤣 Output	conditions 🌷 Programming 🏠	
Configuration	Registered modules	
	1 Digital module 64	
New Module	5 Simulated module	
jital64 🗸 🗸		

In this example a new simulated module is created on ID 5. The values of the simulated module is fetched from ID 1:s flags (sub routines).

This is a good way to send internal flag values out on the bus so that other modules can use this value.

8.2 Increase/Decrease

The function Increase/Decrease can be used to change a value up and down, by just having push buttons. The module will remember this value until it is restarted.

Example:

A concrete mixer want to rotate the concrete at a certain speed. They have one button to increase the speed and one button to decrease the speed of rotation. The third button is used to stop the rotation.

Statements for the flags: Flag1: Button 1= 1 (inc) Flag2: Button 2= 1 (dec) Flag3: Button 3= 1 (stop)

Flags can be chosen freely between 1 and 64

Special features	
Special features, ID:19	
Simulate ID Increase/Decrease Counter PID controllers	
Only for: PWM64 (Multimodule 64)	
Settings	
ms between inc/dec:	_
Send value to port/flag:	×
Increase if flag active: 🌌 1 Inc	×
Decrease if flag active: 🌌 2 Dec	×
Reset if flag active: 🛹 3 Reset	~
	Close
	CIUSE

8.3 Counter

The function Counter can be used to count how many times a signal has been activated. First you select what flag that will trigger the counter to increase. This flag will also get the value of the counter. Then you select a flag that will reset the counter. Last you select how high the counter can count. When the maximum value is reached the counter will stay on the maximum value.

Example:

In the example below, the counter will increase each time flag 10 is active. The counter will reset as soon as flag 20 is active. The value of the counter is saved in flag 10.

The counter can be useful in sequence programs or to count.

Flags can be chosen freely between 1 and 64.

Special features	
Special features, ID:19	
Simulate ID Increase/Decrease Counter PID controllers	
Only for: Digital64 (Multimodule 64) PWM64 (Multimodule 64)	
Settings	
Increase when flag active: 10 Reset when flag active: 20 Maximum 255 📚	~
	Close

8.4 PID controllers

The PID controller can be used to keep e.g. a speed, a position or a temperature at a desired value. It measures the actual value, process value, and compares it with the desired value, setpoint. The difference between process value and setpoint is called control error. The output is calculated and controls e.g. a motor, an actuator or a heater, in order to minimize the control error.

The PID controller output is the sum of three parts:

- The proportional part is the control error, it becomes greater the more process value and setpoint differ. Often, the proportional part alone can not completely remove the control error.
- The integral part is the sum of all previous control errors, the output signal will increase or decrease as long as there is a control error. The integral part makes any remaining control error disappear.
- The derivative part senses the rate of change of the process value. It tries to predict changes in the process value, so that the controller can compensate for them before they become too big. The derivative part has a dampening effect and can make the control both faster and more stable.

The behavior of the controller can be changed by adjusting the three parameters: Gain (Kp), Integral time (Ti) and Derivative time (Td).

Special features	
Special features, ID:19	
Simulate ID Increase/Decrease Counter PID controllers	
Only for: PWM64 (Multimodule 64)	
PID 1 PID 2 PID 3 PID 4	
Output: 🕞 1 Output PID1	~
Process value: 16 Analog module 🛛 💽 🗲 1 ProcessValuePID1	~
Set point: Constant value 💽 127	~
Gain, Kp: 1,0	
Integral time, Ti [s]: 15,0	
Derivative time, Td [s]: 5,0	
	Close

Output

Selects where the controller output is sent, or if the controller should be turned off. The output can be sent to any port or flag in the PWM64 module, and takes precedence over any conditions in the selected port or flag. The controller output has its center value at 127.

Process value

The actual, measured value for the system to be controlled. Can be obtained from any port or flag in any module in the project, or be set to a constant value.

Setpoint

The desired value for the system to be controlled. Can be obtained from any port or flag in any module in the project, or be set to a constant value.

Gain, Kp

Gain of the controller, range 0.0-25.5 times.

A gain of 1.0 means that a change of the process value gives an equally large change of the output value (disregarding integral and derivative part). A gain of 2.0 gives a twice as large change of the output value.

Integral time, Ti

Integral time of the controller, range 0.0-25.5 seconds.

Specifies the time it will take before the integral part has contributed as much to the output as the proportional part has, at a constant control error. A larger value gives less integral action, while a low value gives more. The exception is if 0.0 is specified, then the integral part is turned off completely.

Derivative time, Td

Derivative time of the controller, range 0.0-25.5 seconds.

Specifies how far ahead in time the derivative part predicts changes. A larger value gives more derivative action, while the value 0.0 turns off the derivative part completely.

9 Print:

File View Settings Inser	Print 🛛
🗅 New	Print
😂 Open	◯ Singel module 13 Digital module 64 🗸
🔚 Save	All modules
Save as	
Printer settings	Print Printer setting Close
Exit	

Single module: Print a single module with the ID nr entered in the box.

All modules: Print all modules in the project.

Printer settings: Shows a printer dialog, where you can select witch printer to use.

Print: Starts the print.

Close: Closes the printer window.

	Re Sig	v. datum:	2001-01-12 2004-09-07 Morten			
JÖRGENSEN						
<i>CanPro</i> - Moduldo			nn Ioduldokum	entation		
	G:\cancom\Ca	nPro jobh	_V30-\U442	.ca3		
	ModulID: 3	Modulty	p :PWM-mod	lul Komm	entar :	
	Port	Тур		Kommenta	r	
	1,2 (1A,1B)	PWM ut		Styrning		
	3,4 (2A,2B)	PWM ut		Drivning		
	5 6 (33 39)			P-automa		
Materiallä	7,8 (4A,4B)	PWM ut		Parkering	gsbroms	
Materiana	-					
Filnamn: G:\cancom\CanPro jobb_V3	, Konfig. A	Utgång 1	Utgång 2	Utging 3	Utging 4	
Filnamn: G:\cancom\CanPro jobb_V30 Kommentar :	Konfig. A	Utgång 1 127	Utgång 2	Utging 3	Utging 4	bits
Filnamn: G:\cancom\CanPro jobb_V3(Kommentar : Driftmagneten år ställd på dubbla		127		127 98		
Filnamn: G:\cancom\CanPro jobb_V3(Kommentar : Driftmagneten är ställd på dubbla spolresistansen pga. att det är en	Center Start 1,3,5,7 Start 2,4,6,8	127 20 20	127 20 20	127 98 98	127 98 98	4
Filnamn: G:\cancom\CanPro jobb_V3(Kommentar : Driftmagneten är ställd på dubbla spolresistansen pga. att det är en Haveventil (spolarna har en	Center Start 1,3,5,7 Start 2,4,6,8 Max 1,3,5,7	127 20 20 80	127 20 20 80	127 98 98 99	127 98 98 99	4
Filnamn: G:\cancom\CanPro jobb_V30 Kommentar : Drifmagneten är ställd på dubbla spolresistansen pga, att det är en Haveventi (spolarna har en gemensam punkt)	Center Start 1,3,5,7 Start 2,4,6,8	127 20 20 80 80	127 20 20	127 98 98	127 98 98	4
Filnamn: G:\cancom\CanPro jobb_V30 Kommentar : Driftmagneten år ställd på dubbla spolresistansen pga, att det är en Haveventil (spolarna har en gemensam punkt) Sväng: utgång 1 och 2	Center Start 1,3,5,7 Start 2,4,6,8 Max 1,3,5,7 Max 2,4,6,8	127 20 20 80 80 0,0	127 20 20 80 80	127 98 98 99 99	127 98 98 99 99	* * *
Filnamn: G:\cancom\CanPro jobb_V3(Kommentar : Driftmagneten år ställd på dubbla spolresistansen pga. att det är en Haveventil (spolarna har en gemensam punkt) Sviag: utgång 1 och 2 Drift: Utgång 3 och 4	Center Start 1,3,5,7 Start 2,4,6,8 Max 1,3,5,7 Max 2,4,6,8 RampAce 1,3,5,7 RampAce 1,3,5,7	127 20 20 80 80 0,0 0,0 0,0 0,0	127 20 20 80 0,5 0,5 0,5	127 98 99 99 99 99 0,0 0,0 0,0	127 98 99 99 99 0,0 0,0 0,0	****
Filnamn: G:\cancom\CanPro jobb_V30 Kommentar : Driftmagneten är ställd på dubbla spolresistansen pga. att det är en Haveventil (spolarna har en gemensam punkt) Sväng: utgång 1 och 2 Drift: Utgång 3 och 4 P.automatik: Utgång 5	Center Start 1,3,5,7 Start 2,4,6,8 Max 1,3,5,7 Max 2,4,6,8 RampAcc 1,3,5,7 RampAcc 1,3,5,7 RampAcc 2,4,6,8 RampAcc 2,4,6,8	127 20 20 80 0,0 0,0 0,0 0,0 0,0	127 20 80 80 0,5 0,5 0,5 0,5	127 98 99 99 93 0.0 0.0 0.0 0.0 0.0	127 98 99 99 0,0 0,0 0,0 0,0 0,0	****
Filnamn: G:\cancom\CanPro jobb_V3(Kommentar : Driftmagneten år ställd på dubbla spolresistansen pga. att det är en Haveventil (spolarna har en gemensam punkt) Sviag: utgång 1 och 2 Drift: Utgång 3 och 4	Center Start 1,3,5,7 Start 2,4,6,8 Max 1,3,5,7 Max 2,4,6,8 RampAce 1,3,5,7 RampAce 1,3,5,7	127 20 20 80 80 0,0 0,0 0,0 0,0	127 20 20 80 0,5 0,5 0,5	127 98 99 99 99 99 0,0 0,0 0,0	127 98 99 99 99 0,0 0,0 0,0	e e e e e e e biti
Filnamn: G:\cancom\CanPro jobb_V30 Kommentar : Driftmagneten är ställd på dubbla spolresistansen pga. att det är en Haveventil (spolarna har en gemensam punkt) Sväng: utgång 1 och 2 Drift: Utgång 3 och 4 P.automatik: Utgång 5	Center Start 1,3,5,7 Start 2,4,6,8 Max 1,3,5,7 Max 2,4,6,8 RampAcc 1,3,5,7 RampAcc 1,3,5,7 RampAcc 2,4,6,8 RampRet 2,4,6,8 ITol.	127 20 20 80 0,0 0,0 0,0 0,0 0,0 10	127 20 20 80 0,5 0,5 0,5 10	127 98 98 99 99 0,0 0,0 0,0 0,0 0,0 10 255 0	127 98 99 99 0,0 0,0 0,0 0,0 0,0 10	tite tite tite
Filnamn: G:\cancom\CanPro jobb_V30 Kommentar : Driftmagneten är ställd på dubbla spolresistansen pga. att det är en Haveventil (spolarna har en gemensam punkt) Sväng: utgång 1 och 2 Drift: Utgång 3 och 4 P.automatik: Utgång 5	Center Start 1, 3, 5, 7 Start 2, 4, 6, 8 Max 1, 3, 5, 7 Max 2, 4, 6, 8 RampAcc 1, 3, 5, 7 RampAcc 1, 3, 5, 7 RampAcc 2, 4, 6, 8 RampAcc 2, 4, 6, 8 UTol. Feldet.+ Feldet Lin/Log(S/D/INV	127 20 20 80 0,0 0,0 0,0 0,0 10 250 5 5 12 10 10 10	127 20 20 80 0,5 0,5 0,5 0,5 0,5 10 250 5 Lin.	127 98 98 99 99 0,0 0,0 0,0 0,0 0,0 0,0 0,0 0,0 0	127 98 99 99 99 0,0 0,0 0,0 0,0 0,0 10 255 0 Lin.	% % s s bits bits
Filnamn: G:\cancom\CanPro jobb_V30 Kommentar : Driftmagneten är ställd på dubbla spolresistansen pga. att det är en Haveventil (spolarna har en gemensam punkt) Sväng: utgång 1 och 2 Drift: Utgång 3 och 4 P.automatik: Utgång 5	Center Start 1, 3, 5, 7 Start 2, 4, 6, 8 Max 1, 3, 5, 7 Max 2, 4, 6, 8 RampAcc 1, 3, 5, 7 RampAcc 2, 4, 6, 8 RampAcc 2, 6, 8 RampAcc 2, 6, 8 RampAcc 2, 6, 8 RampAcc	127 20 20 30 0,0 0,0 0,0 0,0 10 250 5	127 20 20 80 0,5 0,5 0,5 0,5 10 250 5	127 98 98 99 99 0,0 0,0 0,0 0,0 0,0 10 255 0	127 98 99 99 0,0 0,0 0,0 0,0 0,0 10 255 0	

10 Programming module:

ScanPro 4.31			
Eile <u>V</u> iew <u>S</u> ettings Insert <u>H</u> elp			
	BENSEN		
🔞 Project information 🥦 Module configuration 🧔 Out	put conditions 🦊 Programming 🗛	Upload program 😕 Analyse	🔍 Analyse specific
Downlod the program to the modules	📫 13 Digital64	Status Module Status	13 Programming
ID Module		Version	14
I3 Digital module 64 Enginero		CAN-interface Checksum	49 9F3D
Program all available modules			
Program specific modules Eg. 2,3,5-8 Program X Cancel			
Module modification			
Lock module with PIN for upload PIN Diversified and the pine of t			
PIN			
GSM : Programming			

Program: There is 3 different ways to program the system.

 ID Module 10 Analog module 	This selection can be used when programming a single module. Select the module to program and press the program button.
🔵 Program all available modules	Program all available modules is used when you want to program all modules in the project. Press the program button, and a list of modules will appear to the right.
Program specific modules Eg. 2,3,5-8	This is a flexible way to program a set of modules from the project. It's very much like how you enter what pages you want to print. Example: 3,5,7,8,9,10 (can also be written as: 3,5,7-10).
🗙 Cancel	If CanPro is in the process of sending its program to the hardware module, it's possible to cancel this action.

Lock module with PIN for upload		
PIN		

You can lock the program in the module by entering a PIN number, and then program this to the hardware. It's now not possible to upload the program from the hardware back to CanPro without the right PIN.

 Event module with PIN for programing
 By locking the ID with a 6-digit PIN code, programming

 PIN
 By locking the module by unauthorized personel is avoided. The modules can be programmed with different PIN codes.



If a module in the project has an ID different than the hardware you want to send the program to it's possible to change the ID of the hardware.

Check the box and enter the current ID of the hardware. After you have programmed the module, it will have changed its ID to the ID of the module in the project.

Example: In my program I have a module assigned to ID 5. My hardware already has an ID assigned to it, ID 10. I select ID 5, check the box *change ID*, and set the value to 10. Then I program. Now the hardware has the ID 5.

Status	
Module	10
Status	Programming
Version	35
CAN-interface	49
Checksum	589A
	•••

Module: Shows witch ID currently being programmed. *Status:* Show the status of the current action. *Version:* Version of the module being programmed. *CAN*-interface: Version of the CAN interface. *Checksum:* Checksum of the program being sent.

۲	PIN			
PIN	N Co	de		
		e is locked with to program the	Pin-code! module, the correct Pin-code must be entered.	
Pin	n code		Unlock Cancel	

In those cases when the CanCom module has been programmed with the programming-PIN, will this dialog show. PIN code has to be entered to program the CanCom module again.

11 Upload Program:

e View Settings Insert Help		
Project information 🔢 Module configuration 🧔	Output conditions 🧔 Programming 🏹	Upload program 😕 Analyse 🍳 Analyse specific
	🔿 10 Analog	Status Module 10
ID Module type	•	Status Uploading program Version 35 CAN-interface 49 Checksum
Upload all registered modules		
Upload specific modules		
Ex: 2,3,5-8		

Uploading of program:



This selection can be used when uploading a single module.

Select the module to upload and press the upload button.

Upload all registered modules is used when all modules on the CAN bus shall be uploaded to CanPro.

This is a flexible way to upload a set of modules from the CAN bus to the project. A requirement is that the project already has the ID:s that you want to upload registered in the project.

After the upload the modules in the project will have the exact code of the modules on the bus.

To enter witch modules to upload I similar to how you enter what pages you want to print Example: 3,5,7,8,9,10 (can also be written as: 3,5,7-10).



If CanPro is in the process of uploading a program from the hardware module, it's possible to cancel this action.

Status	
Module	10
Status	Uploading program
Version	35
CAN-interface	49
Checksum	

Module: Shows witch ID currently being rogrammed. *Status:* Show the status of the current action. *Version:* Version of the module being programmed. *CAN-interface:* Version of the CAN interface. *Checksum:* Checksum of the program being sent.

Note:

If you have an empty project and register a set of modules, it's possible to upload only one of the registered modules. However if the uploaded module interacts with any other module, this will not show in the condition list, since the information from the other modules is missing. In this case the correct thing would be to upload all modules registered in the project.

🔦 PIN	
PIN Code	
The project will then be v	m the module, the correct PIN must be entered.
Open write protective	
Open without protection	Unlock Abort

If the hardware module has been programmed with a PIN code, an upload will result in a dialog window opening. Here it's possible to upload the module to the project as Read-Only or upload the module to the project as normal, with no limitations.

If you want to open up a locket module with no limitations, you have to contact the supplier of the hardware to get the master PIN.

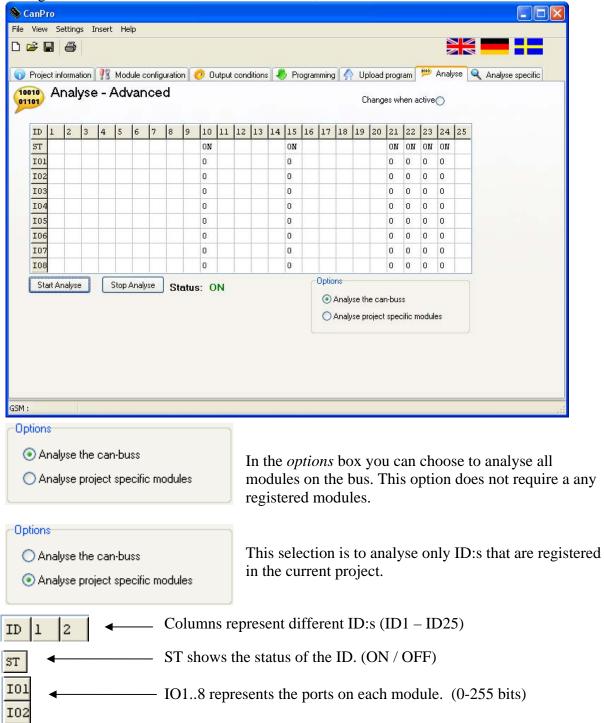
If you upload a module with read-only permissions, you can still change the PWM-settings. This allows you to calibrate the different settings, but will not allow you to change any program logic.

12 Analyse the CAN bus:

This chapter describes how it's possible to analyse the data on the CAN bus in CanPro. This enables you to easily detect errors and get information about inputs and outputs.

12.1 Analyse - Advanced

This grid shows all information on the CAN bus, that are sent between the modules.



12.2 Analyse Specific – Bar graph

ScanPro CanPro			
File View Settings Insert Help)		
		s 🔎 Programming 🛆 Upload program 🤓 A	nalvse
	le configuration 🦉 Output condition	s 🌗 Programming 🥋 Upload program 🗭 A	nalyse 🔍 Analyse specific
	Port	Bar Graph Histogram Ch	anges when active
x 10 Analog module	1 Analog in X-axis	75	
		75	
📁 🕺 10 Analog module 🛛 🔽	2 Analog in Y-axis 🛛 🔽	202	
📁 10 Analog module 🛛 💌	3 Analog in Z-axis 🔽	168	
📁 🗱 10 Analog module 🛛 🔽	4 Analog in motor 1 rev 💉	0	
🔰 10 Analog module 🛛 💌	5 Analog in motor 2 rev 💌	0	
📁 🔰 10 Analog module	6 Analog in motor 3 rev 💌	0	
🗯 10 Analog module 💽	7 Analog in temp 💽	72	
渊 10 Analog module 🛛 🔽	8 Analog in laser distance 🔽	0	
Start Analyse Stop Ar	nalyze Status: Analyse ON		
GSM :			

Bar Graph

Select *Bar graph* to show the selected ports in real time with green bars. Its possible to show up to eight bars at the same time. To select a port, drop down the boxes at the left side, and select the port from your project.

Start Analyse

%

Starts the analyser.

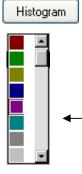
Stop Analyze Stops the analyser. If you leave the Analyse specific tab, the analyse will automatically shut off.

Clear the current row from a selected IO.

Changes when active This radio button will shift(on/off) if data is coming in.

12.3 Analyse Specific – Histogram:

🗞 CanPro		
File View Settings Insert Help	•	
Project information 😗 Modu	le configuration 🤣 Output conditions	👂 Programming 🦙 Upload program 🕮 Analyse 🔍 Analyse specific
Analyse		Bar Graph Histogram Changes when active
Module	Port	
📁 10 Analog module 🛛 🔽	1 Analog in X-axis 🛛 🔽 💌	
📁 10 Analog module 🛛 💌	2 Analog in Y-axis 🛛 💌 📕 👻	200-
📁 10 Analog module 🛛 🔽	3 Analog in Z-axis 🛛 🔽 💌	150
🗱 10 Analog module 🛛 💌	4 Analog in motor 1 rev 🛛 🔽 💙	
🗱 10 Analog module 🛛 💌	5 Analog in motor 2 rev 🛛 🖌	
🗱 10 Analog module 🛛 🔽	6 Analog in motor 3 rev 🛛 🖉 💌	50-
🗱 10 Analog module 🛛 🔽	7 Analog in temp 🔽 🔳 😪	٥ <u>أ</u>
🗱 10 Analog module 🛛 🔽	8 Analog in laser distance 🔽 📕 💙	
Start Analyse Stop A	nalyze Status: Analyse ON	Zoom
		Initial Display 🔿 Pan
		Clear O Zoom Drag
		O Zoom WindPos
GSM :		



Click on the *Histogram button* to enable the histogram graph. It's possible to record up to eight different I/Os at the same time. As with the bar-graph, the IDs ports are selected from the boxes at the left of the window. Each port has also a color assigned to it, showed in the graph.

Color to show in the histogram.

Start Analyse Starts the analyser.

Stops the analyser. If you leave the Analyse specific tab, the analyse will

Stop Analyze

automatically shut off.

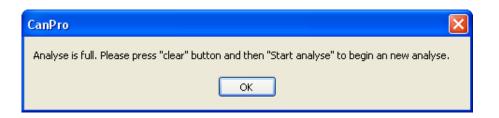
Clear the current row from a selected IO.

Changes when active This radio button will shift(on/off) if data is coming in.

0.0	5.0		10.		15.0		20.0		25.0
	Ū								
1	1	1	1	1	1	1	1	1	1

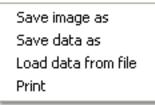
Timebase in seconds. Can be changed before under and after analysis.

O Pan-	Pan the histogram.
OZoom Drag	Stretch the window in both X and Y axis
O Zoom WindPos	Zoom a selected area.
Initial Display	Reset the histogram to its initial window settings.
Clear	Clear the histogram from previous log information.
*	Clear the current row from a selected IO.



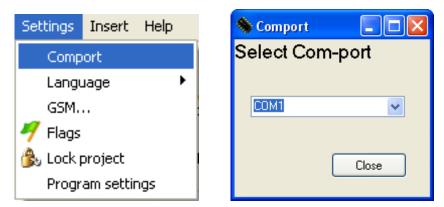
A log file can at most hold 19.000 log points. The logs are saved ever 20 ms, for as many as 8 channels at the time. In other words, more channels equals lesser time before the log is full.

By right clicking on the histogram the following popup menu will show:



Save image as:	Save the histogram as a bmp image.
Save data as:	Save the data in a log file. This file can be opened in CanPro at a later time.
Load data from file:	Load in data from a log file.
Print:	The current histogram is send to a printer with information about what ports has been analysed and their colour code. If a log file is loaded into CanPro information about colour and what ports

13 Com-port



Select the serial port on which the CAN-interface is connected.

14 GSM

is Insert Help
import nguage im ags ck project ogram settings

When a CanCom GSM modem is connected to a CanCom control system a remote PC can:

- Remotely update a program
- Remotely upload the program from the hardware
- Remotely analyse information on the bus to detect errors.

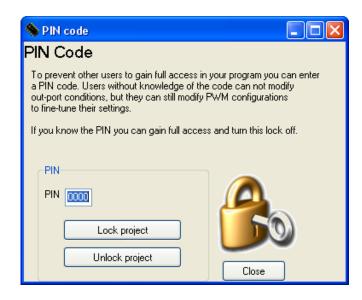
Depending on the telecom operator and country different protocols apply. You have the possibility to change from V.110 to V.32.

🚸 Phone bool	k			
일 Phone	e book			
Name	Number	Misc.	~	
Name	Number	Machine XY		
			Use N	r Close

To open the phone book click on the phonebook icon. Here you have the possibility to store up to 100 numbers.

15 Lock Project





The CanPro project can be protected by a 4 digit code. This will protect the projects from other users to change settings/code in the project. How ever, the project can be opened with a "read only" protection. This enables you to release the project to otherwise less trusted customers. They will then have the possibility to program their machine without you having to worry about them changing any settings or program logic.

Also read about locking program in the hardware modules.

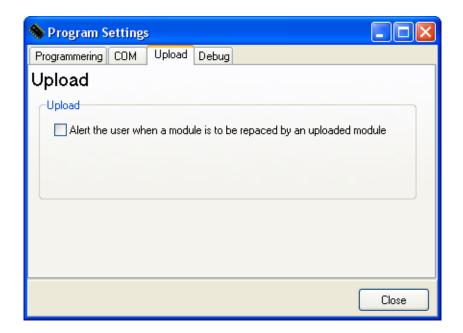
16 **Program Settings**

📏 Program Settings		
Programmering COM Upload Debug		
Download		
Settings		
Pause before downloading next ID 1000 🗘 ms		
Compability mode for Servomodules 3.x		
	O Programmera alla tillgängliga modu	ıler
[Close Programmera utvalda moduler	
	Ex: 2,3,5-8	

When programming more than one ID at the time, there is a pause time between the different ID:s. It's possible to change this pause time to optimize programming, or to raise the time if problems occur when programming.

🗞 Program Settings	
Programmering COM Upload Debug	
Communication	
Communication protocol	
 Automatic detection 	
◯ Ver 3.x	
◯ Ver 4.x	
Close COM port	
	Close

CanPro can be set in three different communication moods. This is due to backward capability with old hardware. If the radio button is set to automatic, the hardware version is detected and the appropriate algorithm is used. The two later radio buttons can lock the communication into ver3 or ver4 communication protocol. This could be used if problems occur when programming with unknown hardware.



When uploading a module from hardware into the CanPro project, and there already exists a registered module in the project with the same ID, a warning will be showed. This gives the user the chose to either continue with the upload and overwrite the previously registered module with the uploaded module, or to abort.

If many modules shall be uploaded you can disable this warning and the modules will be overwritten.

Section Settings	
Programmering COM Upload Debug	
Debug	
Debug	
☑ Debug "Upload"	
☑ Debug "Programming"	
✓ Debug "GSM connection"	
	Close

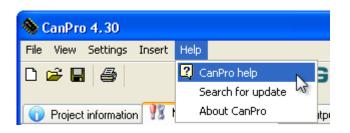
If having problem with the CanPro software, there is a debug functionality built in. To enable this functionality, pleas check the boxes as showed above.

Collected debug information can then be emailed to our software engineers for further investigation.



17 Help

17.1 Manual



Opens this manual. This requires a PDF reader to be installed on the computer. If CanPro uses swedish language, the swedish version of this manual will be opened. Otherwise the english version vill be opened.

17.2 Updates



If connected to the internet, you can search for newer versions of CanPro. You can also visit <u>www.cancom.se</u> to download the software.

🔷 Update		
Updates	- You are now running CanPro 4.28	
		<u>_</u>
	CanPro.exe	
	CanPro ver 4.30 3,1 MB	
	Manual CanPro 4 1,2 MB	
	Information om CanCom/CanPro version 4	
	······································	
		~
Sök	efter nya uppdateringar Stäng	