

PLUTO Safety-PLC

Manual

Absolute Encoders

Table of contents:

1	General	3
1.1	Reaction time	3
1.2	Safety parameters	3
2	Electrical.....	4
2.1	Separation with Gateway in Bridge Mode	5
2.2	CAN bus data	5
3	Singleturn encoder RSA 597/RHA 597	6
3.1	Address setting.....	6
3.2	Software settings.....	7
3.2.1	Baudrate.....	7
3.2.2	Scale	7
3.2.3	Position adjustment	8
3.3	Speed limit	9
3.4	Technical data	9
3.4.1	Electrical data.....	9
3.4.2	Mechanical data	10
4	Multiturn encoder RSA 698/RHA 698	13
4.1	Connectors.....	13
4.2	Indicators.....	13
4.3	Address setting.....	14
4.4	Baudrate.....	14
4.5	Software settings.....	14
4.5.1	Scaling	15
4.5.2	Position adjustment	15
4.6	Technical data	16
4.6.1	Electrical data.....	16
4.6.2	Mechanical data	17
5	PLC Software - Description of function blocks	20
5.1	SafeEncoder	20
5.2	SafeEncoderMult.....	21
5.2.1	SafeEncoderMult32.....	22
5.3	EncoderCam	23
5.3.1	EncoderCam_R.....	23
5.3.2	EncoderCam32	23

1 General

Pluto can handle up to 16 absolute encoders connected to the CAN bus. However, the encoders themselves can only be addressed to one of 8 different addresses (for Multiturn encoders 9 addresses, but a safe solution requires that the encoders are mounted in pairs). This leads to that in practice a maximum of 8 encoders can be used.

One of the Pluto units on the bus must have software option for communication with the encoders. This special Pluto sends cyclically a sync telegram, which trigs all encoders on the bus to read the position and send a telegram with the position value back to the bus. The special Pluto can read the encoder telegrams and evaluate them. With block functions in the PLC code it is then possible to make a dual channel function with two encoders. Out of this the user gets failsafe values for position and speed including stand still monitoring and over speed detection.

The encoders are standard absolute encoders with modified software to meet the safety requirements.

1.1 Reaction time

The encoders are read every 10 ms. The system tolerates that a single reading of an encoder fails, but if two or more fails, it leads to failure alarm. This leads to the response times as below.

Response time:

Normal conditions: 11 ms + program execution time \approx 14 ms

By fault: 31 ms + program execution time \approx 34 ms

1.2 Safety parameters

A system with two encoders connected to a Pluto with a PLC program which uses the block functions described in this manual (see 5) meets the safety levels below.

The values are the same as for the use of other input types, such as normal digital I/O:s or AS-i bus sensors. The values are including the encoders.

SIL according to IEC 61508, EN 62061	SIL 3
Charge pump outputs* (Q2, Q3)	
PFD _{AV} (for proof test interval = 20 years)	1.5×10^{-4}
PFH _D according to IEC 61508, EN 62061	1.5×10^{-9}
Relay outputs* (Q0, Q1, Q4, Q5)	
PFD _{AV} (for proof test interval = 20 years)	1.5×10^{-4}
PFH _D according to IEC 61508, EN 62061	2.0×10^{-9}

PL according to EN ISO 13849-1	PL e
Category according to EN ISO 13849-1	4
MTTF _d according to EN ISO 13849-1	High/1500 years
DC _{avg} according to EN ISO 13849-1	High

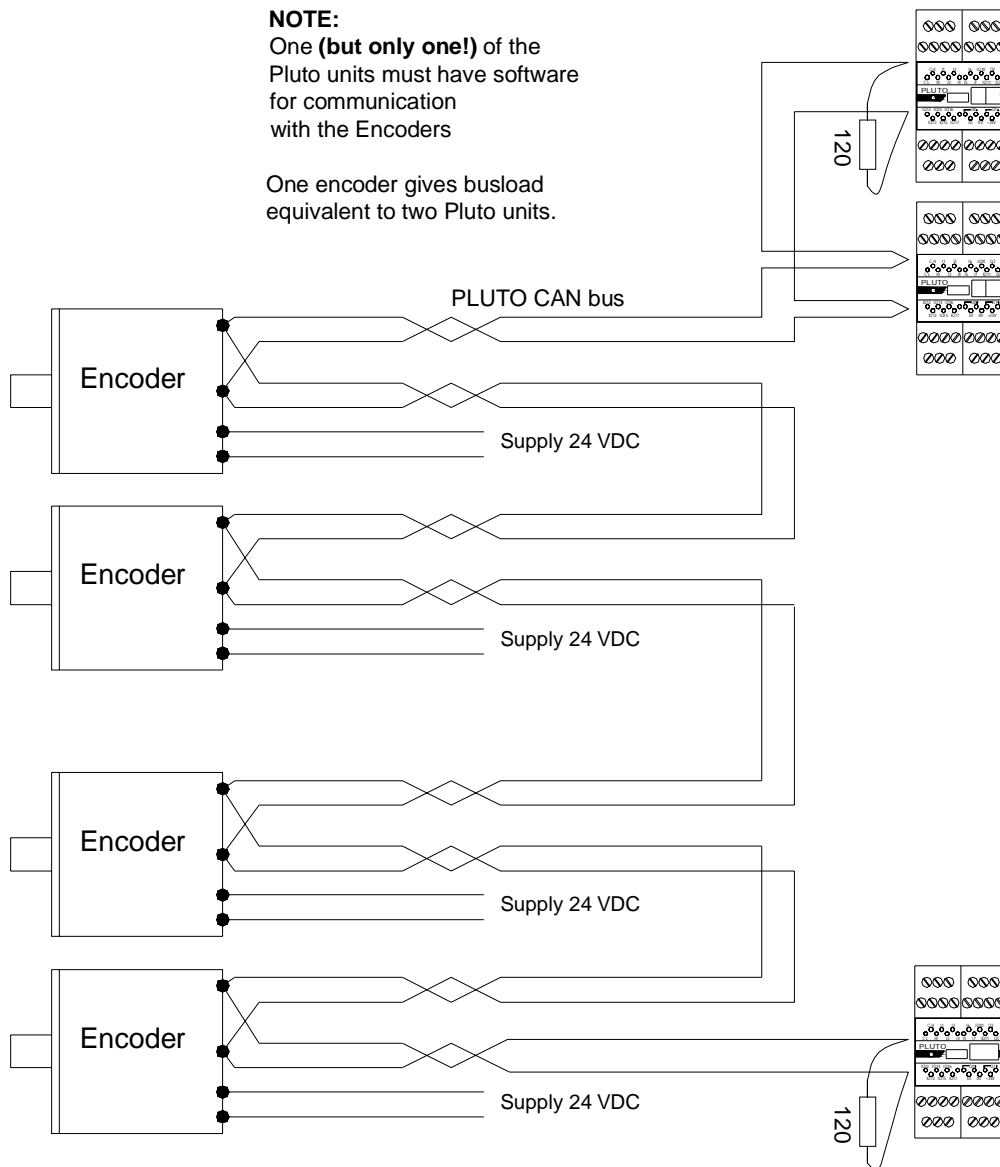
*Encoder to output.

For more details see Pluto hardware manual.

2 Electrical

NOTE:
One (**but only one!**) of the
Pluto units must have software
for communication
with the Encoders

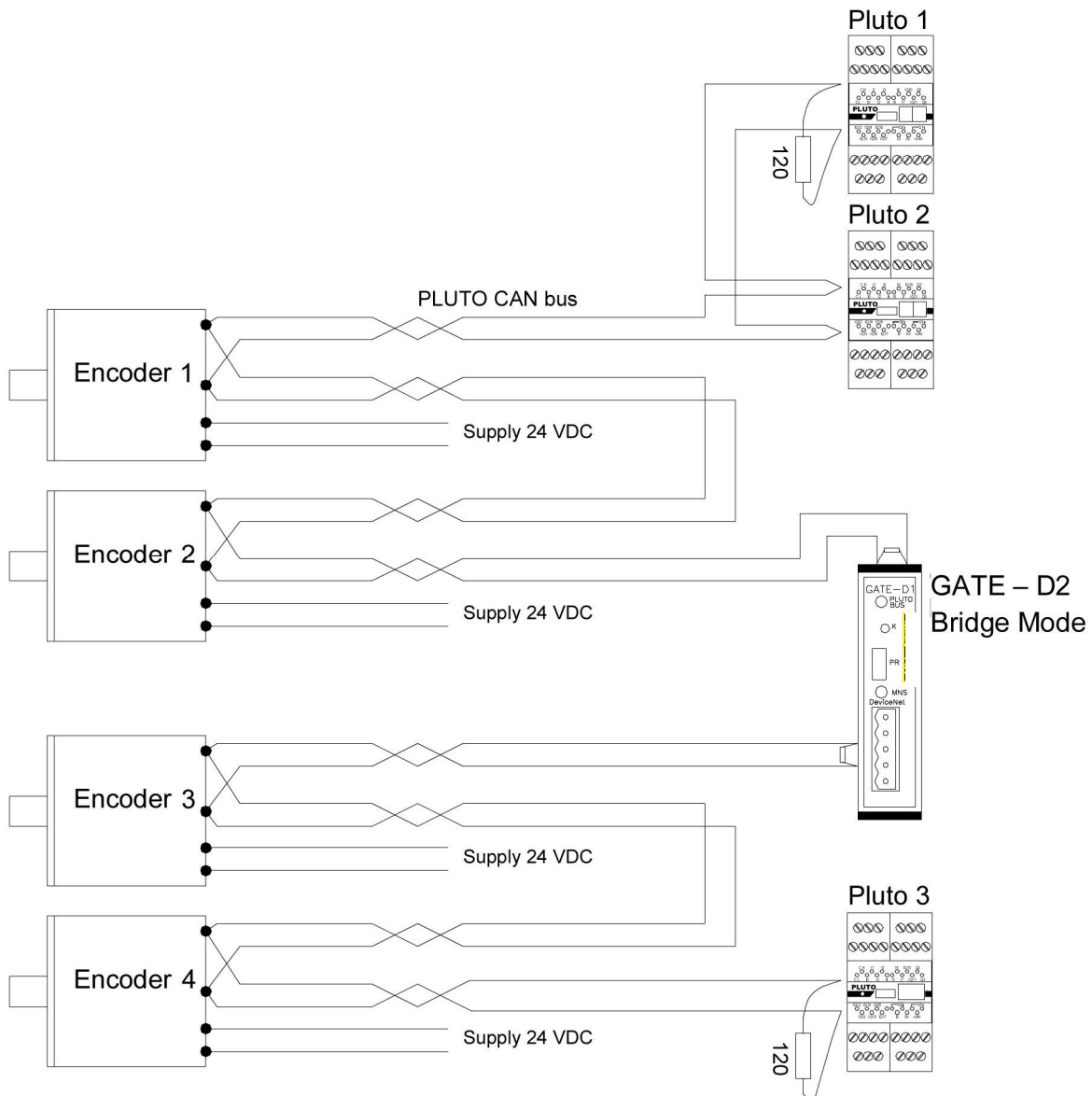
One encoder gives busload
equivalent to two Pluto units.



Connection of CAN bus

2.1 Separation with Gateway in Bridge Mode

By using a Gateway GATE-C2 or GATE-D2 in bridge mode it is possible to separate the Encoders so that different Pluto units communicate with different Encoders. The Gateway will filter out the Encoder telegrams, but let the other telegrams pass through. (See 8.1 “Pluto filter” in the Pluto_Gateway_Manual.)



In this example Pluto 1 communicates with Encoder 1 and 2, while Pluto 3 communicates with Encoder 3 and 4.

2.2 CAN bus data

Each encoder gives bus load equivalent to two Pluto units. This leads to that with a certain CAN speed, the maximum amount of Pluto units will be reduced by two for each encoder. For example a Pluto-bus running with 250 kbit/s and 4 encoders can have 14-24 Pluto units instead of 22-32. For bus data as cable length / baudrate, amount of bus nodes, etc., see Pluto - Operating instructions - Hardware.

3 Singleturn encoder RSA 597/RHA 597



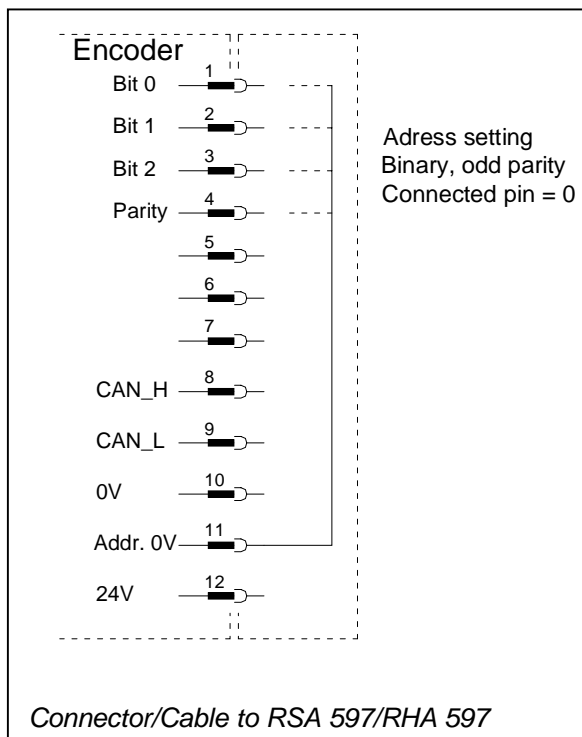
The singleturn encoder comes in four variants:

- RSA 597 with 12-pin connector.
- RSA 597 with 1.5 meter cable.
- RHA 597 with hollow shaft and 2 meter cable.
- RHA 597 with hollow shaft and 10 meter cable.

Besides connection of power supply and CAN bus the connector/cable is also used for setting the address with jumpers (see Address settings below).

3.1 Address setting

The encoder must be addressed 1-8 by connecting pin 1...4 with pin 11 in the connector/cable. By exchange of encoder the addressing will automatically be made by fitting the connector.



Connection type	Cable
<i>Function</i>	<i>Colour</i>
Address Bit 0	White
Address Bit 1	Brown
Address Bit 2	Grey
Parity	Pink
Address 0 Volt	Blue/Red
CAN_H	Green
CAN_L	Yellow
CAN_GND	Grey/Pink
+E Volt	Red
0 Volt	Blue
Shield	Housing

Addr.	Pin 4 Parity	Pin 3 Bit 2	Pin 2 Bit 1	Pin 1 Bit 0
1	0	0	0	1
2	0	0	1	0
3	1	0	1	1
4	0	1	0	0
5	1	1	0	1
6	1	1	1	0
7	0	1	1	1
8	1	0	0	0

1 = Not connected
0 = Connected to pin 11.

Example: Encoder address 3 – Pin 3 connected to Pin 11 and Pin 1, 2, 4 not connected.

3.2 Software settings

An encoder can be adjusted from Pluto by using Terminal Window in Pluto Manager.

By connection the prompt Pluto_A> shall be visible.

Type “encoder” for entering the encoder menu where it is possible to change baudrate (B), set the current position (A) and rescale the encoder (I).

NOTE: If an encoder with wrong baud rate is connected to a running Pluto buss the CAN-bus communication fails.

```

PlutoManager Terminal Window
Send File Close Copy
Pluto_a>
Pluto_a>encoder
Safe encoder commands:

p   display Position
s   Status
v   Version info
b   Baudrate set
a   Position adjust
i   Scale encoder in Increment/rev
h   Help text
<Esc> Back to monitor

Encoder>

```

3.2.1 Baudrate

Baudrate: 125, 250, 500, 800 kbit/s or 1 Mbit/s. (Default Pluto setting 400kbit/s is not possible.) At delivery the encoders are set to 500 kbps. Note that the encoder must be powered off/on before the new setting is valid.

```

Encoder>
Encoder>baud rate set
Encoder node address? (1-16) 1
Desired baudrate ?(125/250/500/800/1000) 500
Command was successful - encoder must be powered on/off
Encoder>

```

3.2.2 Scale

Example of setting the scale to degrees.

```

Encoder>
Encoder>increment/rev scaling
Encoder node address? (1-16) 1
Wanted increments/rev? 360
Direction: ClockWise/AntiClockWise? (c/a)c
Encoder is singleturn
This will rescale the encoder - Are you sure? (y/n)y
Encoder rescaled!
Total resolution=          360 (ClockWise direction)
Encoder>

```

3.2.3 Position adjustment

The current position is set to 90.

```
Encoder>  
Encoder>adjust encoder position  
Encoder node address? (1-16) 1  
Wanted encoder position 90  
This will adjust the encoder position - Are you sure? (y/n)y  
Encoder adjusted!  
Encoder>
```


3.3 Speed limit

By use of single turn encoder for speed monitoring the maximum allowed rotating speed is limited to 50 rev/s = 3000 rpm. If the speed is exceeded the speed value will be negative.

Note: It must be secured by limitations in application that the maximum speed is not exceeded or that no dangerous situations can occur if it does.

Examples of such limitations are: An induction motor which speed is limited by the frequency or a hydraulic cylinder which is limited by a maximum pump capacity.

3.4 Technical data

Type designation	Variant	Interface	Article number, ABB
RSA 597	With 12-pole connector	CAN04	2TLA020070R3600
RSA 597	With 1.5 m. cable	CAN04	2TLA020070R3300
RHA 597	Hollow shaft, with 2 m. cable	CAN04	2TLA020070R3400
RHA 597	Hollow shaft, with 10 m. cable	CAN04	2TLA020070R5900

Encoder data	
Type	RSA 597, RHA 597
Operating temperature	-40°C .. +70°C
Storage temperature	-30°C .. +70°C
Ingress protection class	IP-67 according to IEC 60529
At shaft inlet	IP-66 according to IEC 60529
Vibration (55 to 2000Hz)	< 300 m/s ² according to IEC 60068-2-6
Shock (6ms)	< 2000 m/s ² according to IEC 60068-2-27
Cover material	Aluminium
Cover surface treatment	Coated and cromated or anodized
Weight	Approx. 300g
Accuracy and resolution	
Resolution	13 Bit, 8192 positions per revolution
Accuracy	± ½ LSB

3.4.1 Electrical data

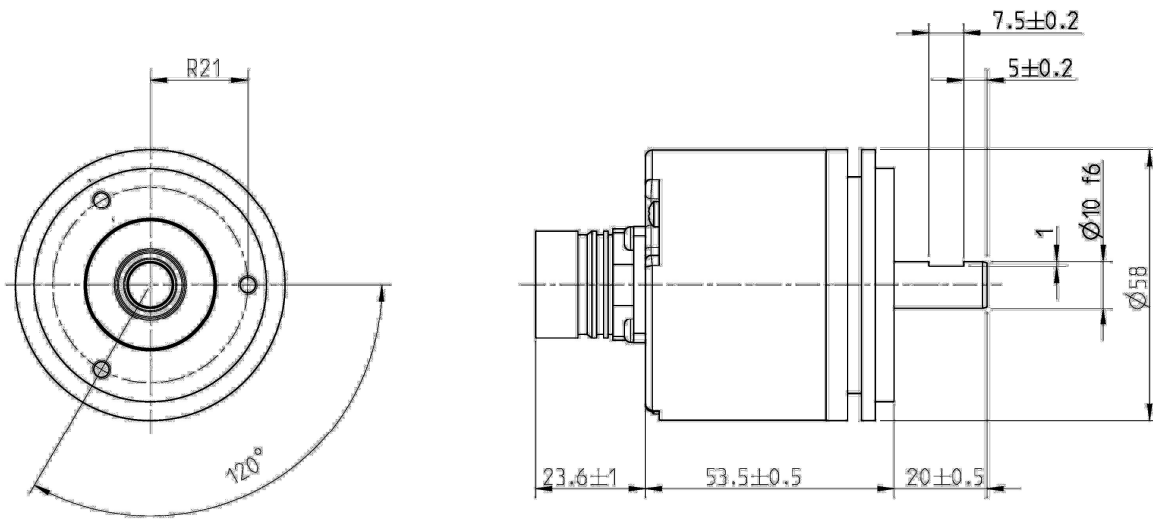
Default baud rate is 500kbit/s.

Power supply	9-36 Vdc
Polarity protected	Yes

Output interface	CANPluto
Short circuit protected	Yes
Interface	CAN specifications 2.0 part A and B
Application layer	Safe Encoder
Baud rate	5 kbit/s - 1 Mbit/s
CAN identifier	3 bit hardware adjustable
Address input	Active low
Code type	Binary
Programmable functions	Resolution, Preset
	Direction, Baud rate
	Node ID
Current consumption	50mA @ 24Vdc
Max current consumption	100mA

3.4.2 Mechanical data

RSA 597 with 12-pin connector



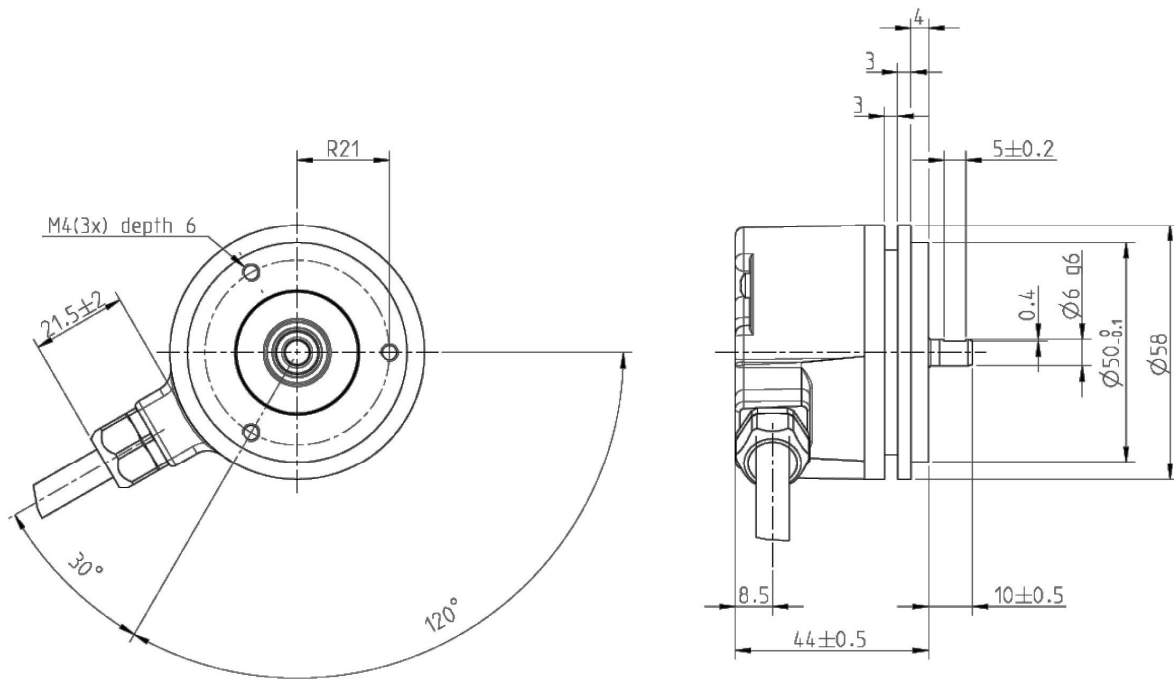
Shaft specification

Shaft type	Ø 10 with face
Axial shaft load	50 N
Radial shaft load	60 N
Mech. permissible speed	6000 rpm (12 000)
Shaft material	Stainless steel
Moment of inertia	$2,0 \times 10^{-6} \text{ kgm}^2$

Flange specification

Flange type	63, Synchro
Outer diameter	ø58 mm
Mounting holes	3 x M4
Flange material	Aluminium
Surface treatment	Anodized

RSA 597 with cable



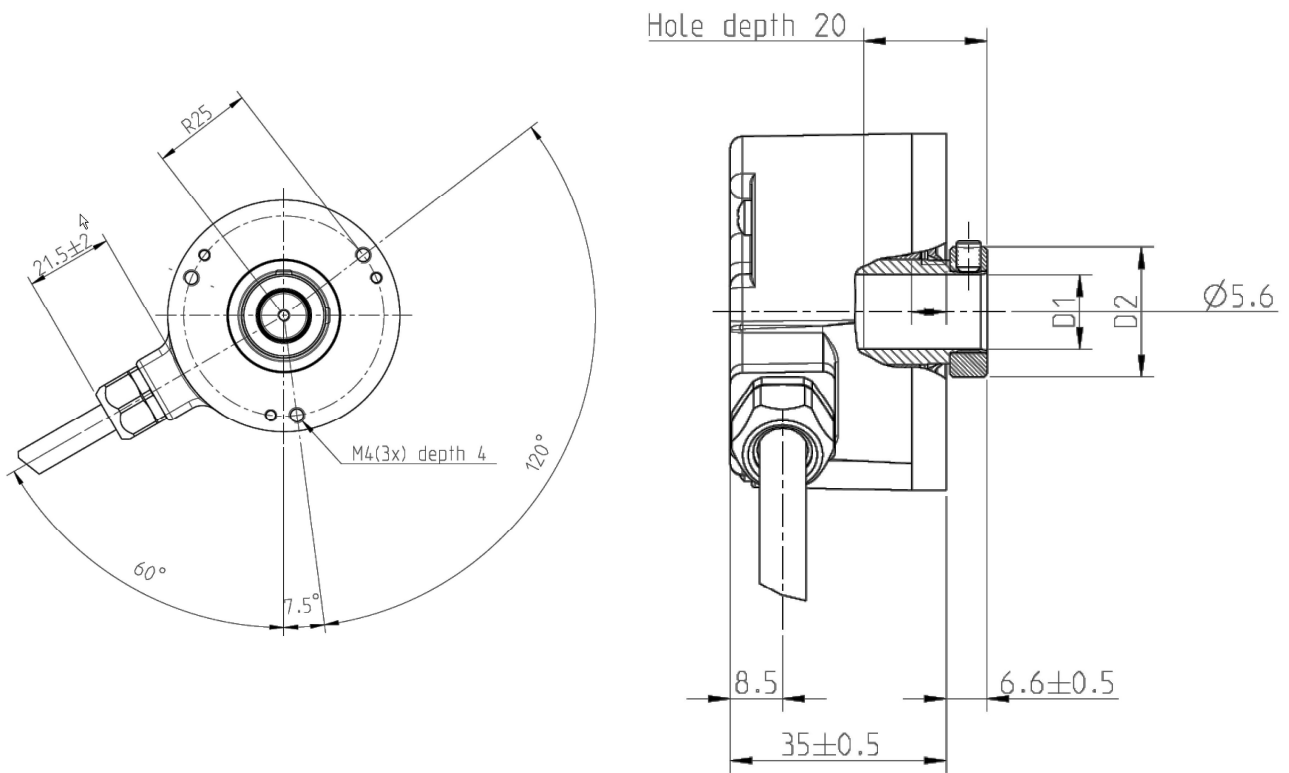
Shaft specification

Shaft type	Ø 6 with face
Axial shaft load	50 N
Radial shaft load	60 N
Mech. permissible speed	6000 rpm (12 000)
Shaft material	Stainless steel
Moment of inertia	$1,9 \times 10^{-6} \text{ kgm}^2$

Flange specification

Flange type	63, Synchro
Outer diameter	Ø58 mm
Mounting holes	3 x M4
Flange material	Aluminium
Surface treatment	Anodized

RHA 597 hollow shaft with cable



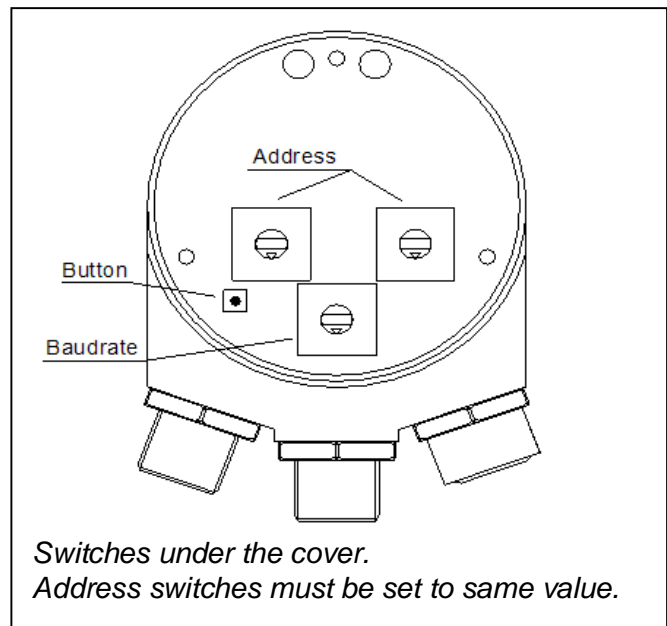
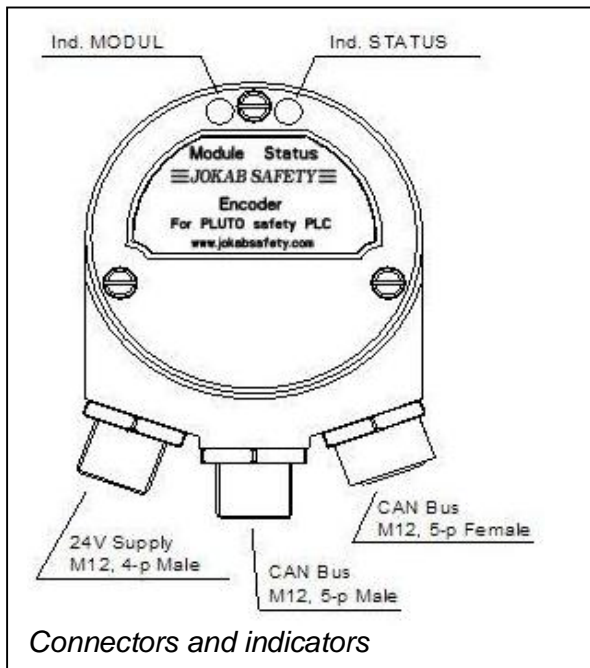
Shaft specification

Hollow shaft type	Ø 12 hs
Axial shaft load	10 N
Radial shaft load	20 N
Mech. permissible speed	6000 rpm
Shaft material	Stainless steel
Moment of inertia	$2,0 \times 10^{-6} \text{ kgm}^2$

Flange specification

Flange type	56, hollow shaft
Diameter	Ø58 mm
Flange material	Aluminium
Surface treatment	Anodized
Torque support	Torque arm

4 Multiturn encoder RSA 698/RHA 698



4.1 Connectors

Power supply 24 VDC	Pin
+24 Volt	1
-	2
0 Volt	3
-	4

CAN Bus	Pin
CAN Shield	1
(CAN V+)*	2
(CAN GND)*	3
CAN High	4
CAN Low	5

*Normally not used

4.2 Indicators

STATUS: Follows the CANopen standard.

Green	OK
Flashing red/green	Fault Example: Wrong baudrate Baudrate conflict
Flashing green/short red	No contact with bus

MODULE: Indicates sensor status.

Green	OK
Flashing red	Fault Example: Address switches set to different values Baudrate switch set to 9. Sensor fault

4.3 Address setting

The address is set by the two rotary switches under the cover. The two address switches shall be set to the same value.

Note: The encoder must be powered off/on before the new setting is valid.

Address	Switch 1, 2
1	1, 1
2	2, 2
3	3, 3
4	4, 4
5	5, 5
6	6, 6
7	7, 7
8	8, 8
9	9, 9

4.4 Baudrate

The baudrate is set by the switch under the cover.

Note: The encoder must be powered off/on before the new setting is valid.

Baud rate	Baudrate switch
10 Kbit	0
20 Kbit	1
50 Kbit	2
125 Kbit	3
250 Kbit	4
500 Kbit	5
800 Kbit	6
1000 Kbit	7
400 Kbit	8
Error	9

4.5 Software settings

Via Terminal window in Pluto Manager it is possible to scale, set actual position and get information.

- Connect the computer to Pluto and start Pluto Manager. Start the terminal window.
- Start by typing `encoder` and `v` and `s` to see if the encoders are present on the bus:

```

PlutoManager Terminal Window
Send File Close Copy
Pluto_a>encoder
Safe encoder commands:

p  display Position
s  Status
v  Version info
b  Baudrate set
a  Position adjust
i  Scale encoder in Increment/rev
h  Help text
<Esc> Back to monitor

Encoder>ver:
ENC SERIAL-NO  OP-TIME NAM, HW_V, SW_V
05  99999999  44 60X, 1.00, 0.10
06  99999999  44 60X, 1.00, 0.10

Encoder>status:
ENC  POSITION ALARM  SCALED RES = INCR*TURNS DIR  HARDWARE RES = INCR*TURNS
05   2 298 0000    4 000 = 04000*00001 ACW  33 554 432 = 08192*04096
06   1 002 0000    4 000 = 04000*00001 CW   33 554 432 = 08192*04096

Encoder>
  
```

In this case encoder 5 and 6 are present on the bus.

The “v” command (version) shows serial number, version of hardware and software and operating time.

The status command “s” gives scaled and hardware resolution, position. Here the encoders are configured to emulate singleturn encoders (scaled res turns=1) with 4000 increments/rev.

The encoders must be scaled so that the total range is with 0 to 31999.

4.5.1 Scaling

The encoder is rescaled by the command "i".

In below example the encoder is set to 100 increments/rev in multiturn mode. This means we can have up to 320 turns within the total range. Rotation direction is clockwise:

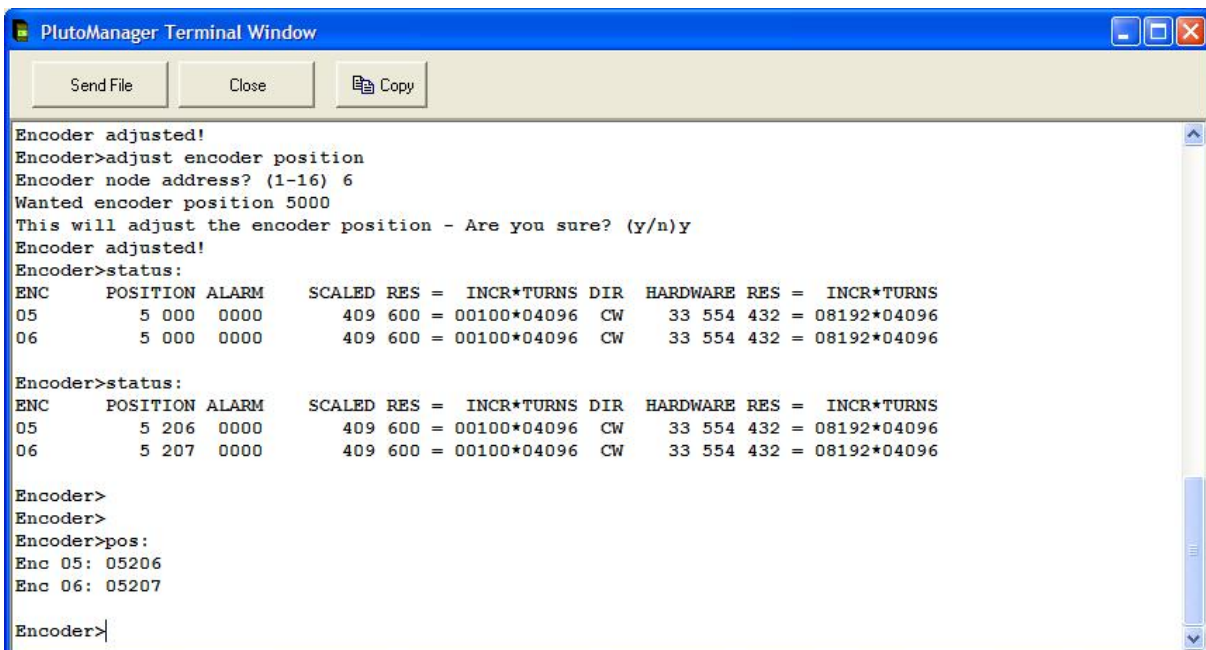
```
Encoder>increment/rev scaling
Encoder node address? (1-16) 5
Wanted increments/rev? 100
Direction: ClockWise/AntiClockWise? (a/c)c
Singleturn/Multiturn? (s/m)m
This will rescale the encoder - Are you sure? (y/n)y
Encoder rescaled!
Total resolution=      409 600 (ClockWise direction)
Encoder>increment/rev scaling
Encoder node address? (1-16) 6
Wanted increments/rev? 100
Direction: ClockWise/AntiClockWise? (a/c)c
Singleturn/Multiturn? (s/m)m
This will rescale the encoder - Are you sure? (y/n)y
Encoder rescaled!
Total resolution=      409 600 (ClockWise direction)
Encoder>status:
ENC    POSITION ALARM    SCALED RES = INCR*TURNS DIR  HARDWARE RES = INCR*TURNS
05     322 044 0000     409 600 = 00100*04096 CW   33 554 432 = 08192*04096
06     370 768 0000     409 600 = 00100*04096 CW   33 554 432 = 08192*04096

Encoder>
```

By typing "s" after the rescaling we can see that the total range is 409600, but since Pluto only has 16-bit arithmetic only the part between 0 and 31999 is usable.

4.5.2 Position adjustment

In below example the actual encoder position is adjusted to 5000.



```
PlutoManager Terminal Window
Encoder adjusted!
Encoder>adjust encoder position
Encoder node address? (1-16) 6
Wanted encoder position 5000
This will adjust the encoder position - Are you sure? (y/n)y
Encoder adjusted!
Encoder>status:
ENC    POSITION ALARM    SCALED RES = INCR*TURNS DIR  HARDWARE RES = INCR*TURNS
05     5 000 0000     409 600 = 00100*04096 CW   33 554 432 = 08192*04096
06     5 000 0000     409 600 = 00100*04096 CW   33 554 432 = 08192*04096

Encoder>status:
ENC    POSITION ALARM    SCALED RES = INCR*TURNS DIR  HARDWARE RES = INCR*TURNS
05     5 206 0000     409 600 = 00100*04096 CW   33 554 432 = 08192*04096
06     5 207 0000     409 600 = 00100*04096 CW   33 554 432 = 08192*04096

Encoder>
Encoder>
Encoder>pos:
Enc 05: 05206
Enc 06: 05207

Encoder>
```

By typing "s" after the rescaling we can see that the position is changed to 5000.

If the encoder is turned and "s" is typed again we can see that the position is changed.

It is also possible to type "p" (position) to get the actual position. If position is displayed by the status command but not for the "p" command, probably the position is outside range 0-31999.

4.6 Technical data

Type designation	Variant	Interface	Article number, ABB
RSA 698	Multiturn	CAN04	2TLA020070R3700
RSA 698	Multiturn, 6mm shaft	CAN04	2TLA020071R7800
RHA 698	Multiturn, Hollow shaft	CAN04	2TLA020071R7900

Encoder data	
Type	RSA 698, RHA 698
Operating temperature	-40°C .. +70°C
Storage temperature	-30°C .. +70°C
Ingress protection class	IP-67 according to IEC 60529
At shaft inlet	IP-66 according to IEC 60529
Vibration (55 to 2000Hz)	< 100 m/s ² according to IEC 60068-2-6
Shock (6ms)	< 2000 m/s ² according to IEC 60068-2-27
Cover material	Aluminium
Cover surface treatment	Anodized
Weight	Approx. 400g
Accuracy and resolution	
Total Resolution	25 Bit
Single turn resolution	13 Bit, 8192 positions per revolution
Multiturn resolution	12 Bit, 4096 absolute number of revolutions
Accuracy	± ½ LSB

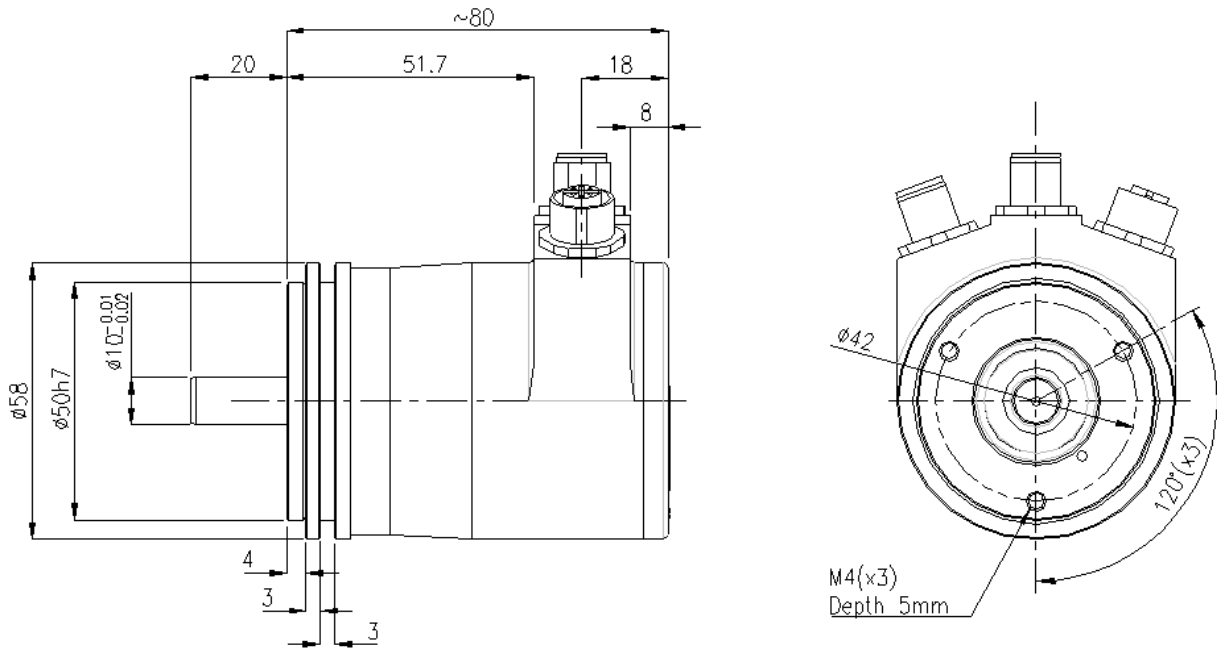
4.6.1 Electrical data

Power supply	9-36 Vdc
Polarity protected	Yes

Output interface	CANPluto
Node address	Settable via DIP Switch
Baud rate	Settable via switch, Max 1MBit/s
Default Baud rate	125 kBit/s
Code type	Binary
Programmable functions	Scaling, Preset
	Code sequence
Current consumption	90 mA @ 24Vdc
Max current consumption	150 mA @ 24Vdc

4.6.2 Mechanical data

RSA 698 Multiturn



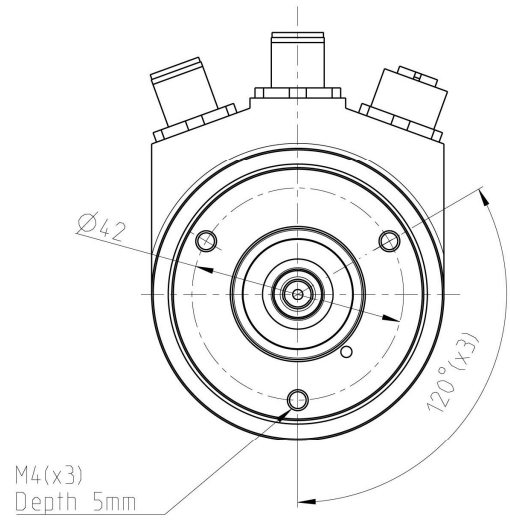
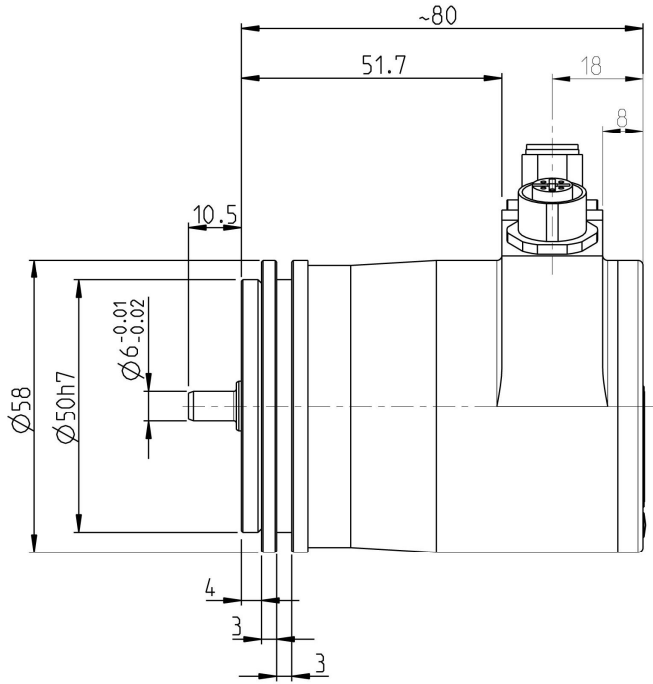
Shaft specification

Shaft type	Ø 10 round
Axial shaft load	50 N
Radial shaft load	60 N
Mech. permissible speed	12000 rpm
Shaft material	Stainless steel
Moment of inertia	$2,0 \times 10^{-6} \text{ kgm}^2$

Flange specification

Flange type	63, Synchro
Outer diameter	Ø58 mm
Mounting holes	3 x M4
Flange material	Aluminium

RSA 698 Multiturn, 6 mm shaft



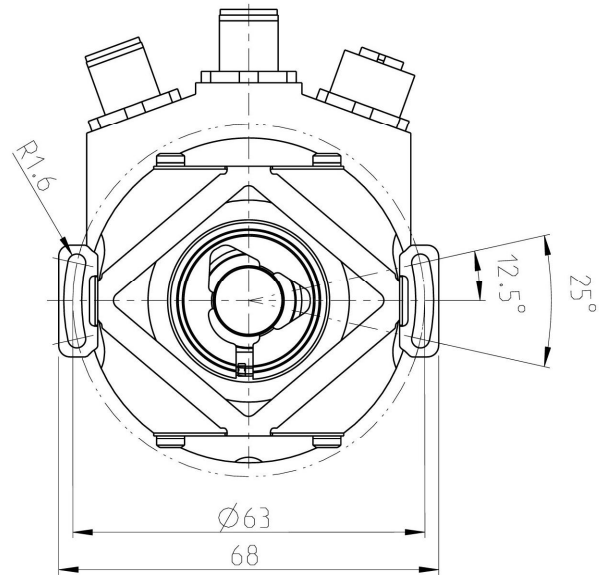
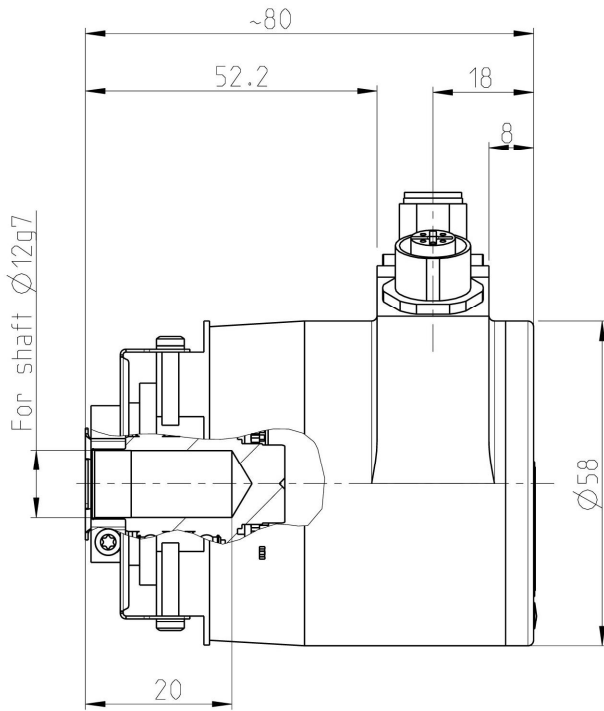
Shaft specification

Shaft type	Ø 6 round
Axial shaft load	50 N
Radial shaft load	60 N
Mech. permissible speed	12000 rpm
Shaft material	Stainless steel
Moment of inertia	$1,9 \times 10^{-6} \text{ kgm}^2$

Flange specification

Flange type	63, Synchro
Outer diameter	ø58 mm
Mounting holes	3 x M4
Flange material	Aluminium

RHA 698 Multiturn, Hollow shaft



Shaft specification

Shaft type	Ø 12 hs
Axial shaft load	10 N
Radial shaft load	20 N
Mech. permissible speed	6000 rpm
Shaft material	Stainless steel
Moment of inertia	$4,3 \times 10^{-6} \text{ kgm}^2$

Flange specification

Flange type	58, HS
Outer diameter	ø58 mm
Torque support	Stator coupling
Flange material	Aluminium

5 PLC Software - Description of function blocks

The safety blocks for reading encoders are located in file "encoder01.fps".

5.1 SafeEncoder

Function block for singleturn encoders generating safe position and speed value out of two absolute encoders.

----- Function -----

The function block reads and evaluates two absolute encoders. The average of the two encoders is calculated and set to the output "Position". The output "Speed" is also an average value given in increments/10ms. It is also monitored that the two encoder values do not deviate more than the value set by the input "MaxDiff".

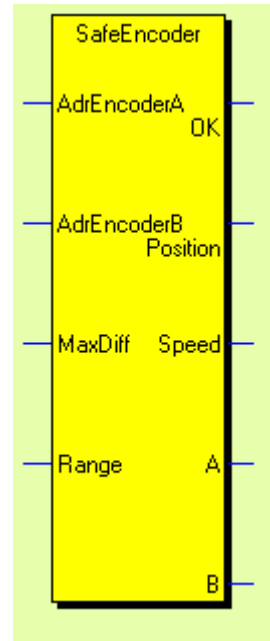
If something is wrong the output "OK" is set "0". In some application the values "Position" and 'Speed' must be used together with the "OK" output.

----- Description of in- and outputs -----

- AddrEncoderA: Encoder A node address
- AddrEncoderB: Encoder B node address
- MaxDiff: Max allowed deviation between the encoders (max 2% of Range)
- Range: Number of increments per revolution
- OK: Set when encoders are working OK and the position values are within the margin set by "MaxDiff"
- Position: Position value
- Speed: Speed value as increments/10ms
- A: Encoder A position. Must not be used in PLC program!
- B: Encoder B position. Must not be used in PLC program!

NOTE! Position values from single encoders are only available for adjustment purposes and must NOT be used for safety.

NOTE! When error occurs "Position" = -1, "Speed" = -32768 and the OK output will be reset.



5.2 SafeEncoderMult

Function block for multiturn encoders generating safe position and speed value out of two absolute encoders.

----- Function -----

The function block reads and evaluates two absolute multiturn encoders. The average of the two encoders is calculated and set to the output "Position".

The output "Speed" is also an average value given in increments/10ms.

It is also monitored so the two encoder values do not deviate more than the value set by the input "MaxDiff".

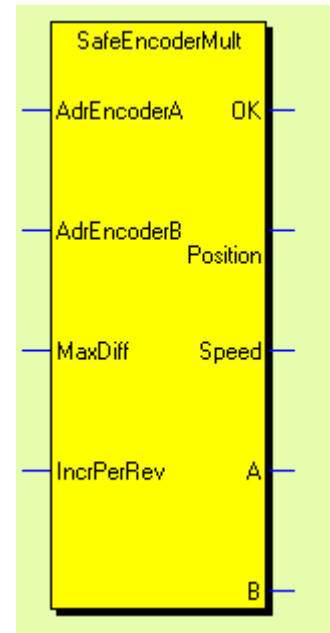
If something is wrong the output "OK" is set "0". In some application the values "Position" and "Speed" must be used together with the "OK" output.

----- Description of in- and outputs -----

- AdrEncoderA: Encoder A node address
- AdrEncoderB: Encoder B node address
- MaxDiff: Max allowed deviation between the encoders (max 2% of IncrPerRev)
- IncrPerRev: Number of increments per revolution
- OK: Set when encoders are working OK and the position values are within the margin set by "MaxDiff"
- Position: Position value
- Speed: Speed value as increments/10ms
- A: Encoder A position. Must not be used in PLC program!
- B: Encoder B position. Must not be used in PLC program!

NOTE! Position values from single encoders are only available for adjustment purposes and must NOT be used for safety.

NOTE! When error occurs "Position" = -1, "Speed" = -32768 and the OK output will be reset.



5.2.1 SafeEncoderMult32

Same as SafeEncoderMult but for Pluto with “Instruction set 3” (see Pluto Programming Manual). Can handle double registers.

----- Function -----

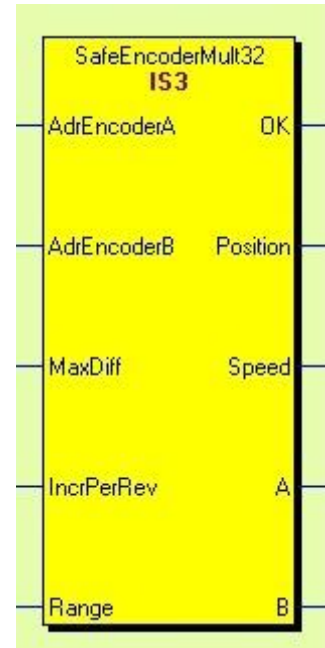
The function block reads and evaluates two absolute multiturn encoders. The average of the two encoders is calculated and set to the output “Position”.

The output “Speed” is also an average value given in increments/10ms. It is also monitored so the two encoder values do not deviate more than the value set by the input “MaxDiff”.

If something is wrong the output “OK” is set “0”. In some application the values “Position” and “Speed” must be used together with the “OK” output.

----- Description of in- and outputs -----

- AdrEncoderA: Encoder A node address
- AdrEncoderB: Encoder B node address
- MaxDiff: Max allowed deviation between the encoders (max 2% of IncrPerRev)
- IncrPerRev: Number of increments per revolution
- Range: This value should be set to the total number of increments (IncrPerRev * number of turns). *See example below.
- OK: Set when encoders are working OK and the position values are within the margin set by “MaxDiff”
- Position: Position value
- Speed: Speed value as increments/10ms
- A: Encoder A position. Must not be used in PLC program!
- B: Encoder B position. Must not be used in PLC program!



*Example: An encoder has a resolution of 1000 increments per revolution (IncrPerRev) and 4096 turns. In this case “Range” should be set to 4096000 and “IncrPerRev” to 1000. In terminal window this can be seen by typing “encoder”, followed by “s”.

```
Encoder>status:
ENC      POSITION ALARM      SCALED RES = INCR*TURNS DIR  HARDWARE RES = INCR*TURNS
01      4 095 900 0000      4 096 000 = 01000*04096 CW   33 554 432 = 08192*04096
03      4 095 900 0000      4 096 000 = 01000*04096 ACW  33 554 432 = 08192*04096
```

*Example: If IncrPerRev = 1000, Range shall be set to 4096000 (1000*4096).*

NOTE! Position values from single encoders are only available for adjustment purposes and must NOT be used for safety.

NOTE! When error occurs “Position” = -1, “Speed” = -2147483648 and the OK output will be reset.

5.3 EncoderCam

Function block for electronic cam limit switch.

----- Function -----

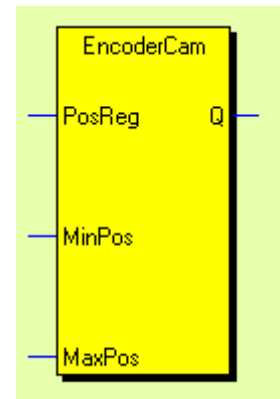
The output Q is set when the value in the input register PosReg is within the limits MinPos and MaxPos. The input value is normally Generated by the function block SafeEncoder.

NOTE! It is possible to specify a cam which passes through the zero position of an encoder. Position<0 is forbidden.

Example: If MinPos=3000 and MaxPos=200, Q is set when the position is greater than 2999 or less than 201.

----- Description of in- and outputs -----

- PosReg: Register where the position is stored
- MinPos: Min allowed value
- MaxPos: Max allowed value



5.3.1 EncoderCam_R

Function block for electronic cam limit switch where inputs MinPos and MaxPos also can be registers.

Function

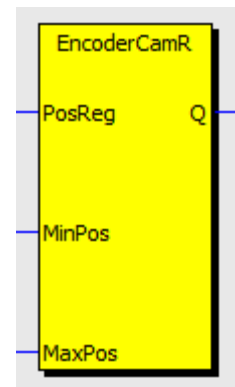
The output Q is set when the value in the input register PosReg is within the limits MinPos and MaxPos.

NOTE! It is possible to specify a cam which passes through the zero position of an encoder. Position<0 is forbidden.

Example: If MinPos=3000 and MaxPos=200, Q is set when the position is greater than 2999 or less than 201.

Description of in- and outputs

- PosReg: Register for the encoder position.
- MinPos: Min/start value. Constant or register
- MaxPos: Max/stop value. Constant or register
(Only positive values are allowed)



5.3.2 EncoderCam32

Same as EncoderCam but can handle double registers.
Needs "Instruction set 3" (see Pluto Programming Manual).

----- Function -----

The output Q is set when the value in the input register PosReg is within the limits MinPos and MaxPos. The input value is normally Generated by the function block SafeEncoder.

NOTE! It is possible to specify a cam which passes through the zero position of an encoder. Position<0 is forbidden.

Example: If MinPos=3000 and MaxPos=200, Q is set when the position is greater than 2999 or less than 201.

----- Description of in- and outputs -----

- PosReg: Register where the position is stored
- MinPos: Min allowed value
- MaxPos: Max allowed value

