ANALOG TRANSMITTER

PD 1611

Manual

Contents

1. General information.

The PD 1611 Analog Transmitter is a part of **Proces-Data**'s module series 1000. The PD 1611 module has been developed for storing and processing of up to 6 analog measurement signals.

1.1. Functions.

* Connection of analog measurement signals on 6 different channels. The measurement signals could be voltage (0-500 mV), current (0-20 or 4-20 mA) or output from temperature detector Pt-100.

* Analog current output 4-20 mA used for instance for displaying one of the 6 results of measurement on an indicating meter.

* Internal PID-regulator, used for regulating purposes via the analog output.

* 2 internal limit channels for use as alarm detectors, or for batching from a user defined measurement and a set point.

* Direct connection of the PD 230 display. Using the display unit it is possible to select various measurement results as well as setting the module for various functions.

* Connection to PD's local network, the P-net. Using the P-net, it is possible to carry out the same functions (as the display unit) from a central computer.

* Voltage output 24 V DC max. 150 mA. Useful to power passive digital in- and outputs or other external equipment.

* Advanced self testing facility which can be monitored using the display unit or through the P-net.

1.2. Block diagram of PD 1611.


```
                    ┌──────────┐         ┌──────────────┐
                    │   CPU    │         │  Display     │◄──────►
                    │          │         │  interface   │
                    └──────────┘         └──────────────┘
   ┌──────────┐                          ┌──────────────┐
──►│    6     │                          │  P-net       │◄──────►
   │ Analog   │◄─────                     │  interface   │
   │ inputs   │                          └──────────────┘
   └──────────┘
                    ┌──────────┐
                    │    2     │◄─────
                    │ Dosing   │
                    │ channels │
                    └──────────┘
                    ┌──────────┐         ┌──────────────┐
                    │   PID-   │◄────►   │ Current output│──────►
                    │regulator │         │  4-20 mA     │
                    └──────────┘         └──────────────┘
   ┌──────────┐     ┌──────────┐         ┌──────────────┐
──►│  Power   │     │ Service  │◄─────   │  Voltage     │──────►
   │ supply   │     │ channel  │         │  output      │
   └──────────┘     └──────────┘         │ 24 V DC/150 mA│
                                         └──────────────┘
```

                                                                    5020311RB
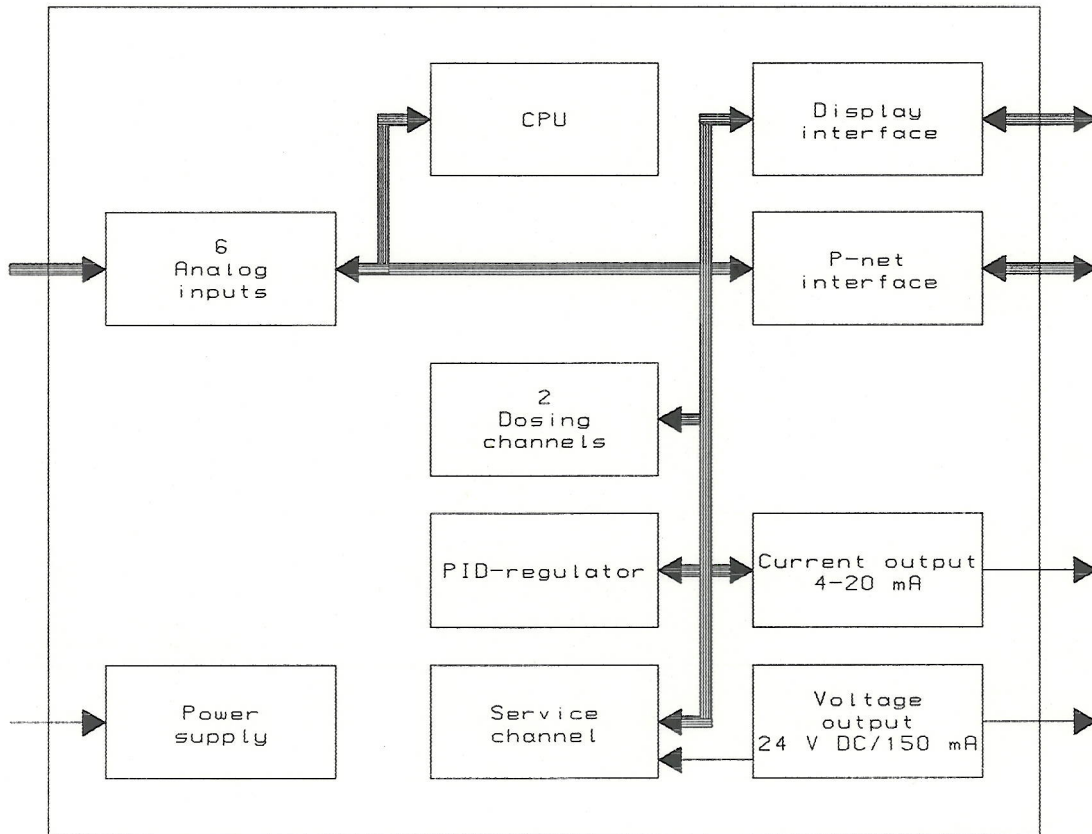
1.3. Connection to P-NET.

The PD 1611 module can be connected to Proces-Data's local area network P-Net, which is a local area network intended for process control and collection of data.

The PD 1611 module is controlled via the P-Net. Setting the module for the functions required, and communication between the module and a control computer is also carried out via the P-Net.

A typical application of the P-Net is that of transmitting measurements from one or more modules to a central computer and displaying the results on a screen. Furthermore, it is possible to control the outputs (digital as well as analogue) via the computer and to check modules for internal & external errors.

More information about the operation and structure of P-Net can be obtained by writing to **Proces-Data.**

1.4. Channels/registers.

The PD 1611 module contains:

1 Service channel	(channel 0)	1 PID regulator	(channel 8)
6 Analog inputs	(channel 1-6)	2 Dosing channels	(channel 9-A)
1 Current output	(channel 7)	1 Display channel	(channel B)

A set of 16 registers numbered from 0 - F, is associated with each channel (see diagram on following page). For addressing a register within a particular channel, a symbolic address of 2 hexadecimal digits is used.

```
Example: Register 4 on channel 8 needs to be addressed.
         The symbolic address will then look like this:    8 4

                          Channel number  ─────────────
                          Register number ─────────────
```
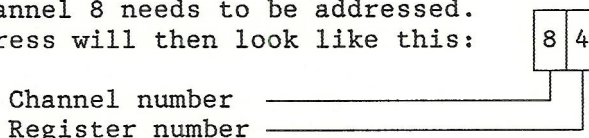
If attempting to read a non-defined register, the module will not respond. This will be indicated as an error in the P-net transmission.

Throughout the manual the register sets will be visualized in a table. The example below concerns the register sets belonging to channel 0.

The register names are standard identifiers defined in Process-Pascal.

Reg.no.	Register name	Write	Storage medium	Number of bytes	Read out
0					
1	DeviceType		P	2	Decimal
2	PrgVers		P	2	Decimal
3	Error3	X	R	1	Hexadec.
4					
5					
6					
7	WDTimer	X	R	2	Decimal
8	WDPreset	X	E	2	Decimal
9	Code9	X	E	4	Hexadec.
A					
B					
C					
D					
E					
F	ErrorF		R	1	Hexadec.

Storage media: R;RAM, E;EEPROM, P;PROM, BR;BATTERY RAM

Data stored in EEPROM is retained after a reset, or after a power failure. If this data needs to be modified, the "Program Enable" switch must be ON.

Data stored in Battery RAM is retained after a reset or a power failure.

Types of channel

The module consists of several channels which each have their own function. These channels can be combined to suit the needs of the user. The module contains channels of the following three types:

Input channel:

Direct electrical connection to input device. Input data is stored in a register of its own channel. Typically, it's possible to use scaling.
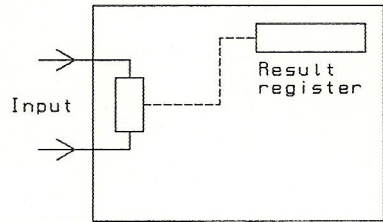
502025AI

Internal channel:

Channel without electrical connection. The channel takes input data from other channels, or the contents of registers which have been set by the display unit or P-net.

Input data is selected using a code register. The resultant data is stored in a associated output channel register.
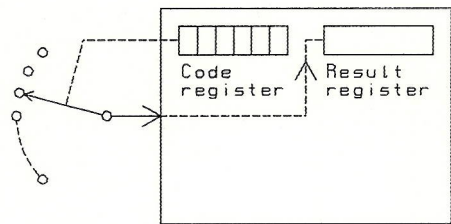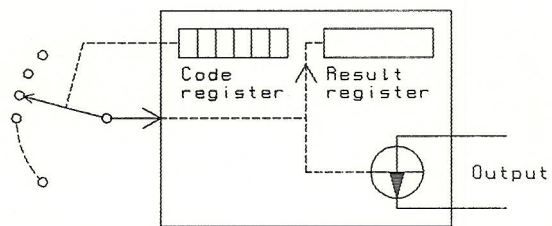

502025AI

Output channel:

Direct electrical connection to output device. The channel takes data from other channels or from registers which have been set via the display-unit or P-net.

Input data is selected using a code register. Output data is stored in a register which directly controls the output device.

502025AI

1.5. Combining Channels.

If, for example, the current output is required to be controlled by a measured temperature on one of the two analog inputs, it can be done by pointing to the measured value on the analog input, using a code reg, in the current output channel (see fig. 3).

As soon as a measured value is pointed to by the code register, the module will create an indirect register, where the measured value can be read. This means that the measurement can be read both in the result register in the analog input channel, and in the indirect register in the current output channel.

When pointing at address 00 by means of a code register, it is possible to write directly in the indirect register.

Note: It is not possible to point to an indirect register using a code register, as this will result in an error on that channel. The indirect registers are marked with an asterisk in the definition of registers for the various channels.
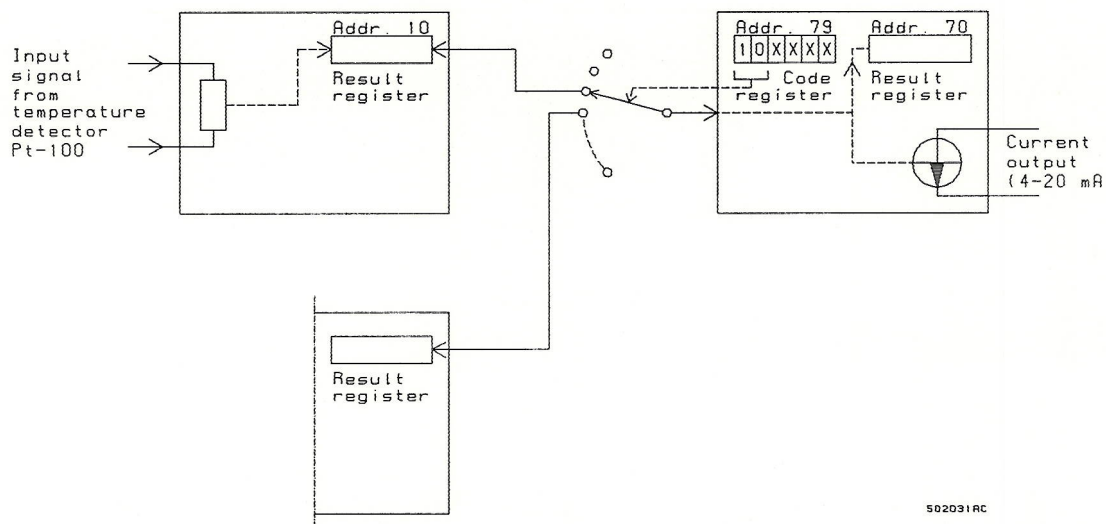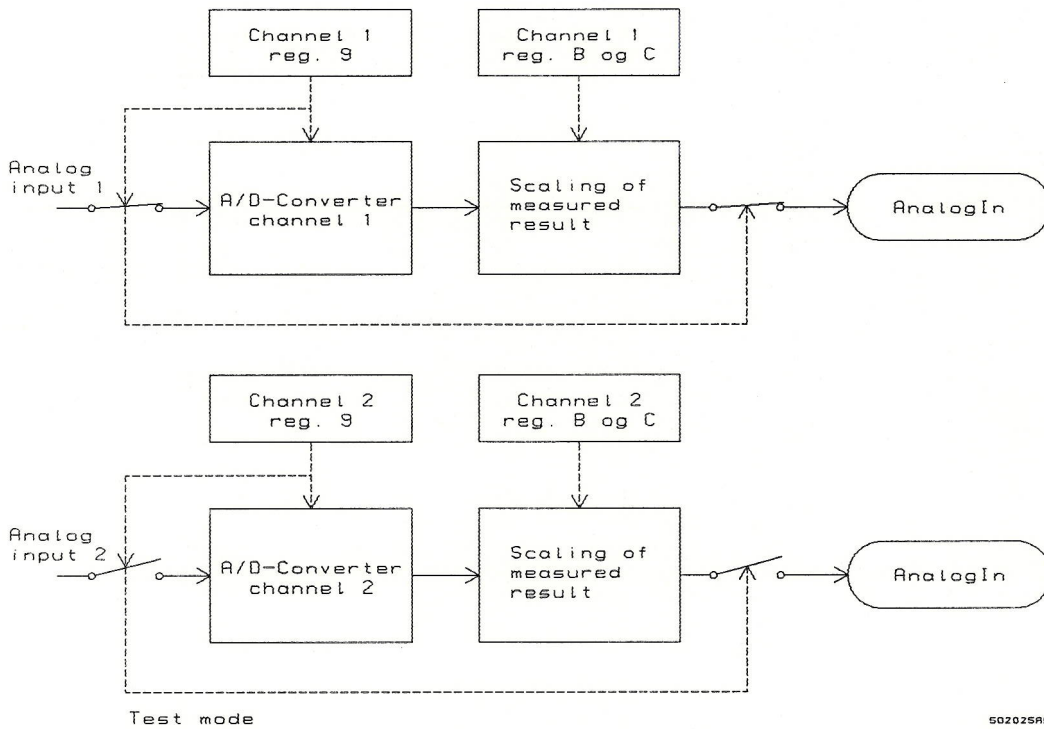

Fig. 1.5.a.

2. Analog input signals.

Up to 6 analog input signals may be connected to the PD 1611, being current (0-20 or 4-20 mA), voltage (0-500 mV) or a Pt-100 temperature detector. Signal type etc. is selected separately for each channel by means of a code.

2.1. Block diagram showing Analog input channels.

Test mode

502025RK

2.2. Registers on analog input channel (channel 1-6).

In the 6 analog input channels (channel 1-6) the following registers have been defined.

Reg.no.	Contents	Write	Storage medium	Number of bytes	Read out
0	AnalogIn		R	4	Decimal
1					
2					
3					
4					
5					
6					
7					
8					
9	Code9	X	E	4	Hexadec.
A					
B	Fullscale	X	E	4	Decimal
C	ZeroPoint	X	E	4	Decimal
D					
E					
F	CHError		R	1	Hexadec.

*   Indirect register

Reg. 0: AnalogIn.

The measurement result, being temperature, current or voltage, can be read out from register 0.

Reg. B: FullScale.

FullScale is only used with current or voltage signals. The value expected in register 0, at full signal on the analog input (20 mA / 500 mV), will be the same as the value placed in the FullScale register.

Reg. C: ZeroPoint.

Current and voltage signals, the value read in register 0 at minimum signal on the analog input (0 mA / 4 mA / 0 mV) will be the same as the value placed in the ZeroPoint register. If the module has been programmed as a Pt-100 input, ZeroPoint is used for offset-adjustments of the temperature detector.

Reg. 9: Code9.

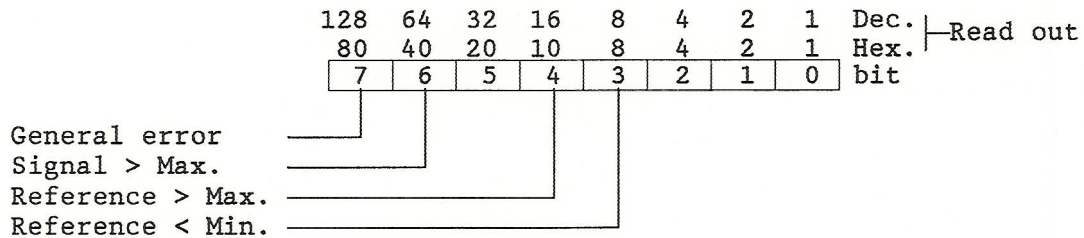The code consists of 6 digits each with its own meaning.

```
Code register for Analog        |Cif.1|Cif.2|Cif.3|Cif.4|Cif.5|Cif.6|
input channel (Reg. 9).

        Not in use

    0: No 3 mA alarm
    1: 3 mA alarm limit

    0: Normal mode
    1: TEST-mode

    0: Not in use
    1: Temperature Pt-100
    2: Current 0-20 mA
    3: Current 4-20 mA
    4: Voltage, 0-500 mV

        Not in use
```
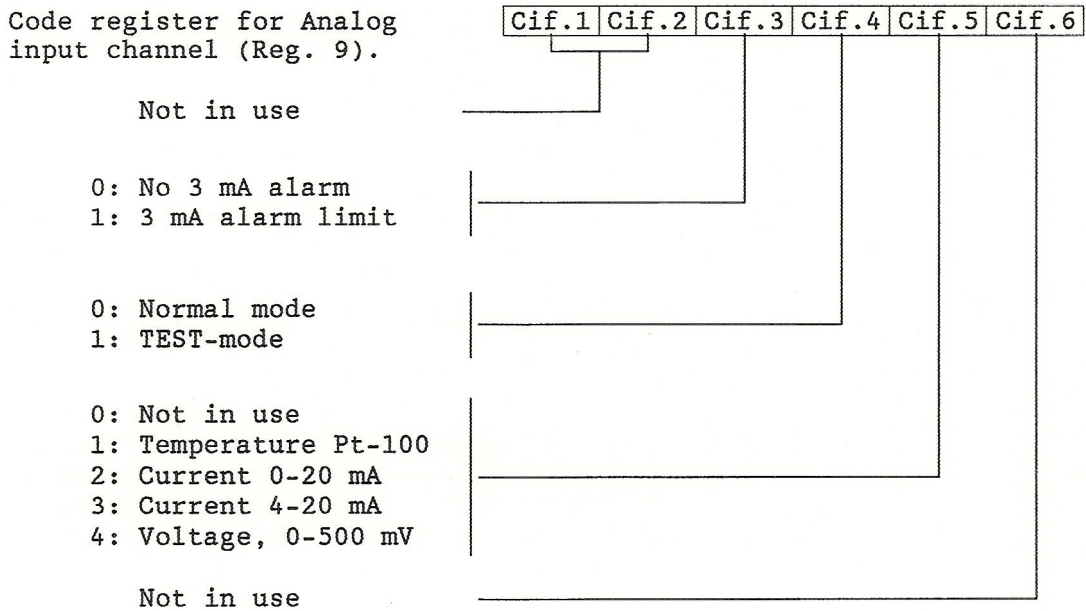
Note: If the channel is not in use, write "0" in digit 5 ("channel not in use"), otherwise errors can occur.

When current is being measured (4-20 mA), an extra function can be used (3 mA alarm limit). When this function is required, the module will generate an error as soon as the current on the input falls below 3 mA. (see register F).

When the channel is in TEST-mode, no measuring result will be calculated but it is possible for the user to insert a random value in register 0.

Reg. F: CHError.

The CHError register shows whether there is currently an error on the channel concerned, or whether there is a general error on the module which might lead to the channel being defective.

```
        128  64  32  16   8   4   2   1  Dec. ┐
         80  40  20  10   8   4   2   1  Hex. ├Read out
        | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | bit ┘

General error      ─────
Signal > Max.      ─────────
Reference > Max.   ─────────────
Reference < Min.   ─────────────────
```

If bit 7 (General error) is set, then the other bits are insignificant, as the general error can be derived from error codes on the other channels (see Service Channel).

2.2.1. Description of error codes on channel 1 - 6.

Bit	Meaning	Type of signal	Error source
6	Signal > 400°C	Temperature	Input signal too high
5	Signal < -30°C	Temperature	Input signal too low Temp. detector short circuit
4	Int. reference > max.	Temperature	Internal fault
3	Int. reference < min.	Temperature	Temp. detector disconnected Internal fault
6	Signal > 500 mV	Voltage	Input signal too high
5		Voltage	
4	Int. reference > max.	Voltage	Internal fault
3	Int. reference < min.	Voltage	Internal fault
6	Signal > 20 mA	Current (4-20 mA)	Input signal too high
5	Signal < 3 mA	Current (4-20 mA)	Input signal too low Internal fault
4	Int. reference > max.	Current (4-20 mA)	Internal fault
3	Int. reference < min.	Current (4-20 mA)	Internal fault
6	Signal > 20 mA	Current (0-20 mA)	Input signal too high
5		Current (0-20 mA)	
4	Int. reference > max.	Current (0-20 mA)	Internal fault
3	Int. reference < min.	Current (0-20 mA)	Internal fault

2.3. Connection to analog input channels.

2.3.1. Temperature measurements.

The PD 1611 module can be connected to temperature detectors of the Pt-100 type. The detector is connected to one of the two sets of terminals marked "A-B-C-D" as shown in fig. 2.3.1.a.

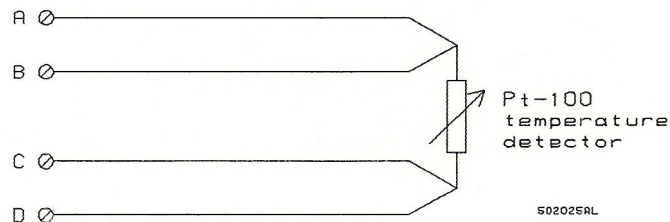Fig. 2.3.1.a.

2.3.2. Current measurement, 0-20 or 4-20 mA.

The PD 1611 module can be connected to analog current signals, 0-20 mA or 4-20 mA.

The signal is connected to one of the two sets of terminals marked "A-B-C-D" as shown in fig. 2.3.2.a.
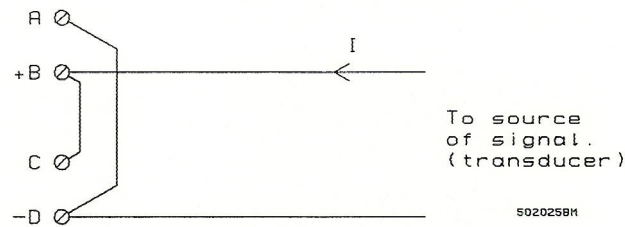

Fig. 2.3.2.a.

Example of the connection of two-wire transducers, supplied from the module's voltage output. See fig. 2.3.2.b.
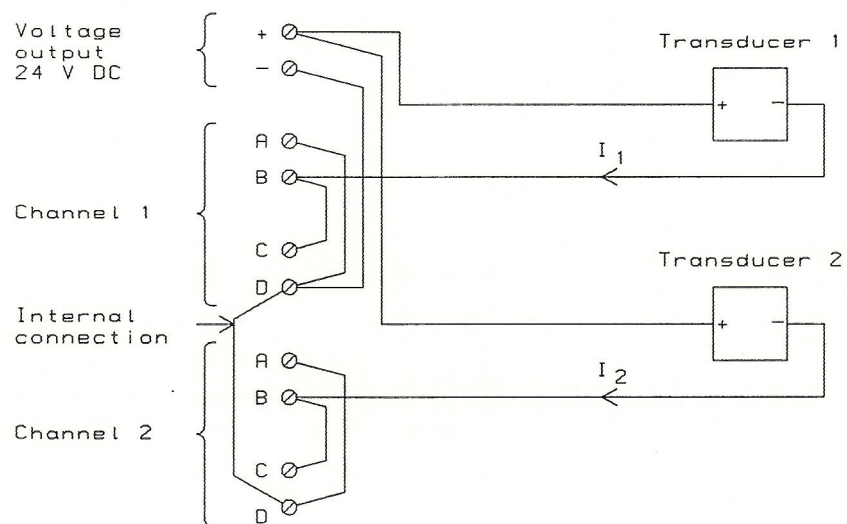
Fig. 2.3.2.b.

2.3.3. Voltage measurement, 0-500 mV.

The PD 1611 module can be connected to analog voltage signals, 0-500 mV.

The signal is connected to one of the two sets of terminals marked "A-B-C-D" as shown in fig. 2.3.3.a.

```
A  ⊘
       ⟍
+B  ⊘————————————————————————
                              To source
                              of signal.
                              (transducer)
C  ⊘
       ⟍
-D  ⊘————————————————————————              50202SAM
```
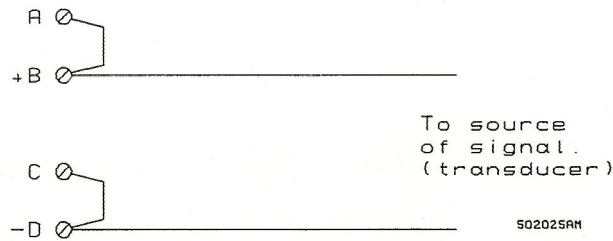
Fig. 2.3.3.a.

The modules input impedance is so high, that ordinary voltage division with two resistors can be used, if the signal to be measured is higher than 0-500 mV.

Example: Input signal 0-5 V. See fig. 2.3.3.b.

```
                        90  kOhm
A  ⊘
       ⟍    ┌────[======]────
+B  ⊘──────┤
           │
           [ ]  10 kOhm          Signal, 0-5 V
C  ⊘       │
       ⟍    │
-D  ⊘──────┴────────────────              50202SBO
```

Fig. 2.3.3.b.

3. Current Output.

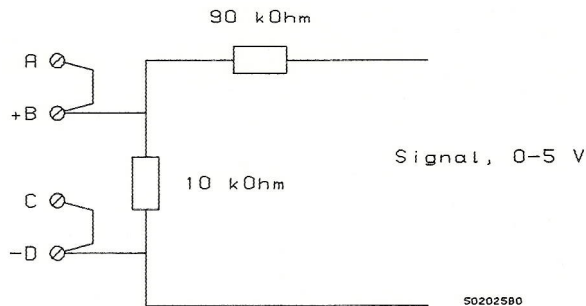The PD 1611 module has a current output (4-20mA) which is an output channel (channel 2). The current output can be the result of a measurement, or a value set by the display unit PD 230, or the P-net.

The current output can also represent the output signal from an internal PID Regulator and can be used for control purposes.

The signal on the current output can be read via display-unit or P-net as a number between 0-100%.

Block diagram of current output function:

502025AP
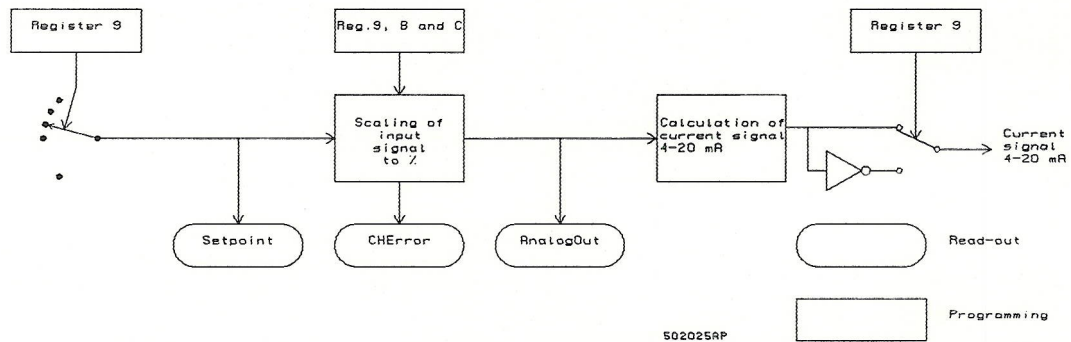
Registers in the current output channel (channel 7).

Reg.no.	Register name	Write	Storage medium	Number of bytes	Read out
0	AnalogOut		R	4	Decimal
1					
2					
3					
4					
5					
6					
7	Setpoint *	X	BR	4	Decimal
8					
9	Code9	X	E	4	Hexadec.
A					
B	FullScale	X	E	4	Decimal
C	ZeroPoint	X	E	4	Decimal
D					
E					
F	CHError		R	1	Hexadec.

*   Indirect register.

Reg. 0: Analog Out

Using this register the output signal can be displayed as a number between 0-100%.

Reg.7: Setpoint *

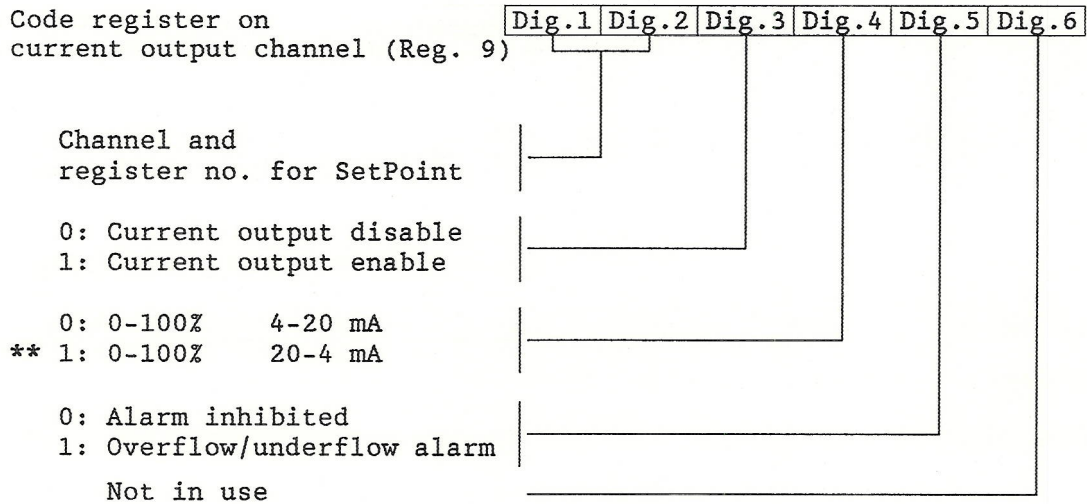The value which the current output is required to attain (if settled) can be seen in this register.

If the code register points to address 00, then this register can be written to.

Reg. 9: Code

Code register for the current output channel (Reg. 9)

```
Code register on               | Dig.1 | Dig.2 | Dig.3 | Dig.4 | Dig.5 | Dig.6 |
current output channel (Reg. 9)


      Channel and
      register no. for SetPoint

      0: Current output disable
      1: Current output enable

      0: 0-100%    4-20 mA
  **  1: 0-100%    20-4 mA

      0: Alarm inhibited
      1: Overflow/underflow alarm

         Not in use
```

** Inverting the current output.

If an address in digit 0 and 1 is defined, then the corresponding indirect register can be written to.

An indirect register must not be pointed to by the two first digits in the code.

The current output can be disabled using digit 3 (current output disabled).

Using digit 4, it can be decided whether 100% will correspond to 20 mA or 4 mA.

The two alarms "overflow" and "underflow" (see register F) can be disabled by digit 5.
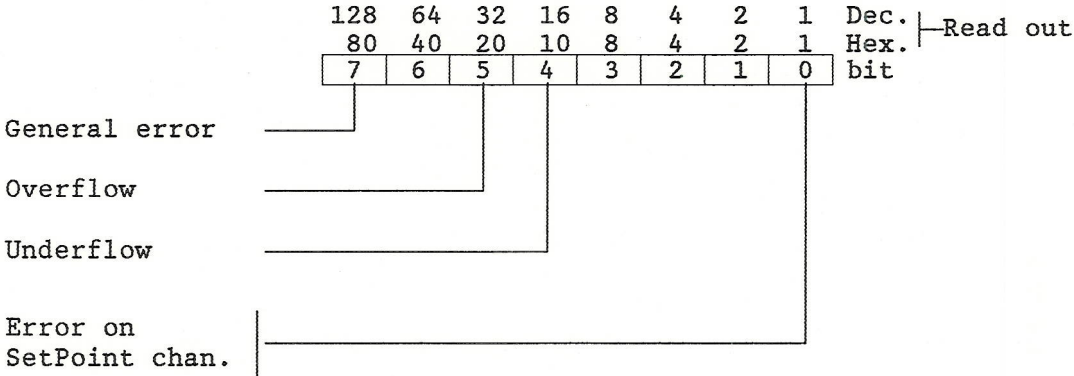
Reg. B: Fullscale

Fullscale is the value for which the max.output signal (100%) is required on the current output.

<u>Reg. C: Zero Point</u>

Zeropoint is the value for which the min.output signal (0%) is required on the current output.

<u>Reg. F: CHError</u>

The CHError register shows whether there is currently an error on the channel, or whether there is a general error in the module which might lead to the channel concerned being defective.

```
          128  64  32  16   8   4   2   1  Dec. ┐
           80  40  20  10   8   4   2   1  Hex. ├─Read out
          ┌───┬───┬───┬───┬───┬───┬───┬───┐      │
          │ 7 │ 6 │ 5 │ 4 │ 3 │ 2 │ 1 │ 0 │ bit ┘
          └───┴───┴───┴───┴───┴───┴───┴───┘

General error  ─────────┘       │

Overflow       ─────────────────┘

Underflow      ─────────────────────┘

Error on
SetPoint chan. │─────────────────────────────┘
```

If bit 7 (general error) is set, then the other bits are insignificant, as the general error can cause additional errors on the channels (see for example "service channel" part 8).

Bit 5 (overflow) is set if the output is more than the fullscale value (Reg. B).
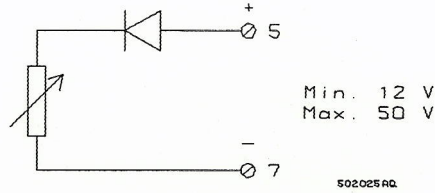
Bit 4 (underflow) is set, if the output is lower than the zeropoint value (Reg. C).

If Bit 0 is set, the module has found an error in the channel from where the output is derived.

3.1. Current Output, Electrical.

The current output is designed as a passive current generator and the voltage must therefore be applied from an external circuit (or perhaps from the built in voltage output in the module).

Fig. 3.1.a.: Current output, schematic.


```
                              +
                   ⯈|⎯⎯⎯⎯⎯⎯ ⊘ 5

                              Min.  12  V
                              Max.  50  V

                              ⎯
                   ⎯⎯⎯⎯⎯⎯⎯⎯⎯ ⊘ 7
                                      502025AQ
```

The current output is galvanically separated from the other part of the electronics by an opto-coupler. Furthermore, the output is protected against wrong polarization by a zenerdiode and a current limiting resistor as shown in the figure below. This resistance is rated so that limitation occurs at a current of approximately 35 mA. Before the output can be used again, the output must be disconnected totally for a few seconds.

The external control equipment must be rated such that the voltage over the current output range (terminal point 5 and 7 on the base circuit board) is always is min.12.v., the internal control circuitry, being supplied by this voltage.

Fig. 3.1.b.: Current output.


```
                                          +
                                  ⎯⎡/⎤⎯⎯ ⊘ 5

         ┌─────────────┐
         │ Current     │         ⎔  Dzener,  51 V
         │ regulation  │
         │ circuit     │
         └─────────────┘
                                      ⎯⎯ ⊘
                       5020258P
```
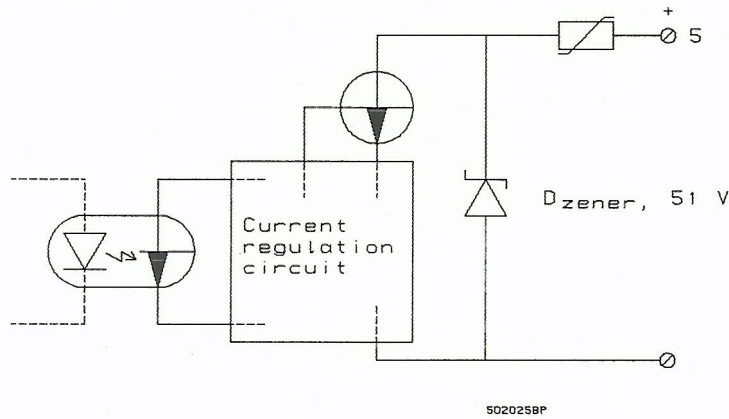
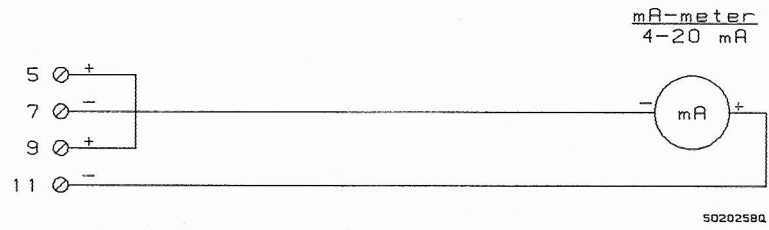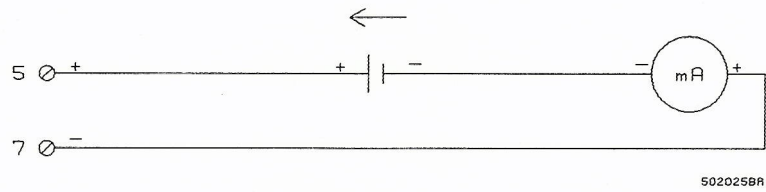Fig. 3.1.c.: Connection of current output with supply from the internal voltage output of the module.

mA-meter
4-20 mA

5 ⊘ +
7 ⊘ −
9 ⊘ +
11 ⊘ −

− mA +

5020258Q

Fig. 3.1.d.: Connection of current output with external supply.

5 ⊘ +   + | | −   − mA +
7 ⊘ −

5020258R

4.   **PID Regulator.**

The PD 1611 module is equipped with an internal PID regulator (channel 8), which can be used for various control purposes.

The PID regulator can be used both to control a current signal (4- 20,mA) or a cyclic signal (ie.a digital signal with a set frequency but variable on-time). This is done through other channels in the module.

The regulator is set with the P, I and D parameters and a code, which defines the input signals to the regulator.

It is possible to define three different signals to the regulator, 1,2 and setpoint.

The input signals can either come from the input channels of the module, or be input via display unit or P-net.

The signals 1 and 2 can be added, subtracted, multiplied, or divided with each other. Thus it is possible to set the module, for example, for proportional regulation between 2 measured values.

Using the display unit or the P-net, the regulator can be set to "Manual".

Block diagram of PID regulator:

502025AA
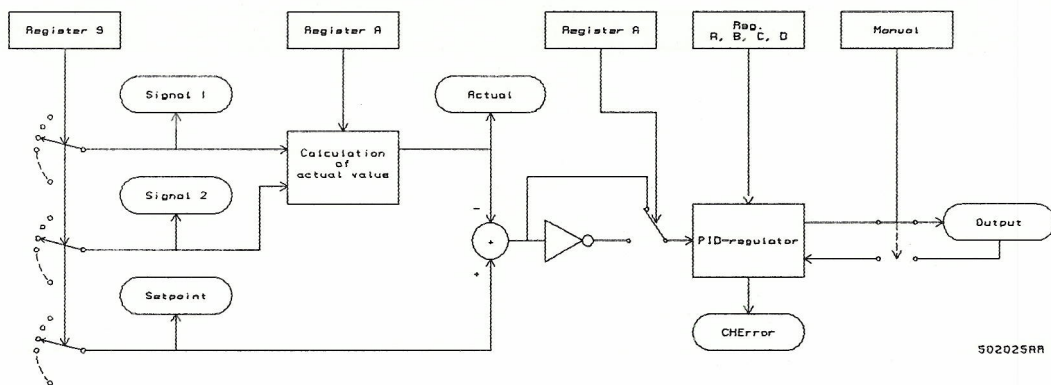
Registers for PID Regulator (channel 6)

Reg.no.	Register name	Write	Storage medium	Number of bytes	Read out
0	Output		R	4	Decimal
1	Signal1 *	X	BR	4	Decimal
2	Signal2 *	X	BR	4	Decimal
3	Actual		R	4	Decimal
4					
5					
6					
7	Setpoint *	X	BR	4	Decimal
8	FagReg8	X	R	1	Binary
9	Code9	X	E	4	Hexadec.
A	CodeA	X	E	4	Hexadec.
B	Xp	X	E	4	Decimal
C	Ti	X	E	4	Decimal
D	Td	X	E	4	Decimal
E					
F	CHError		R	1	Hexadec.

* Indirect Registers.

Reg. 1: Signal1 *

The selected input value for signal 1 can be seen in this register.

Reg. 2: Signal2 *

The selected input value for signal 2 can be seen in this register.

Reg. 3: Actual

The result of the calculation using the two input values (signal 1 and signal 2) is stored in this register.

Reg. 7: Setpoint *

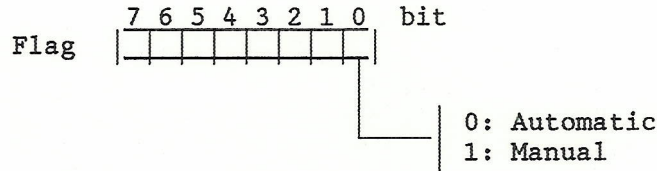The setpoint, which is used for PID regulator control, can be read from this register.

Reg. 0: Output

The output signal from the PID-regulator can be read from this register as a number between 0 and 100%. If the output should exceed 100%, then the output register will statisize at 100%. Also the module will generate an error code (see register F). This also occurs if the output signal drops below 0%.
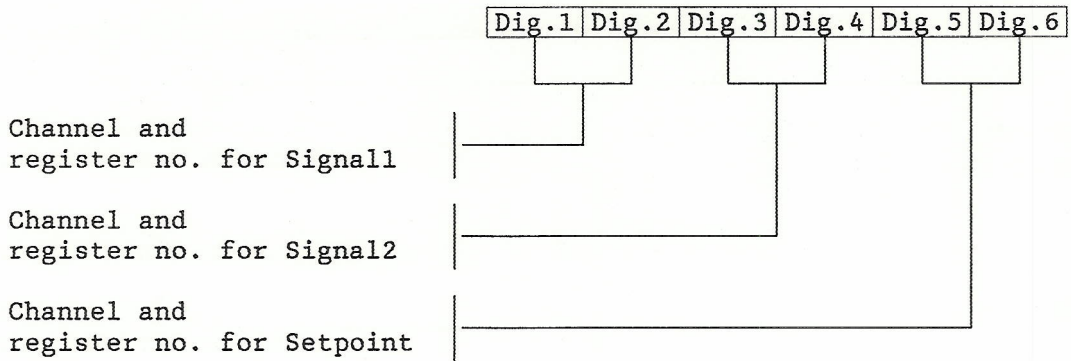
Reg. 8: FlagReg8

This flag register can be used for switching between automatic & manual control. That is done by writing a 0 or 1 into the register, as shown below:

```
           7 6 5 4 3 2 1 0   bit
Flag      |_|_|_|_|_|_|_|_|
                         |
                         |____|  0: Automatic
                              |  1: Manual
```
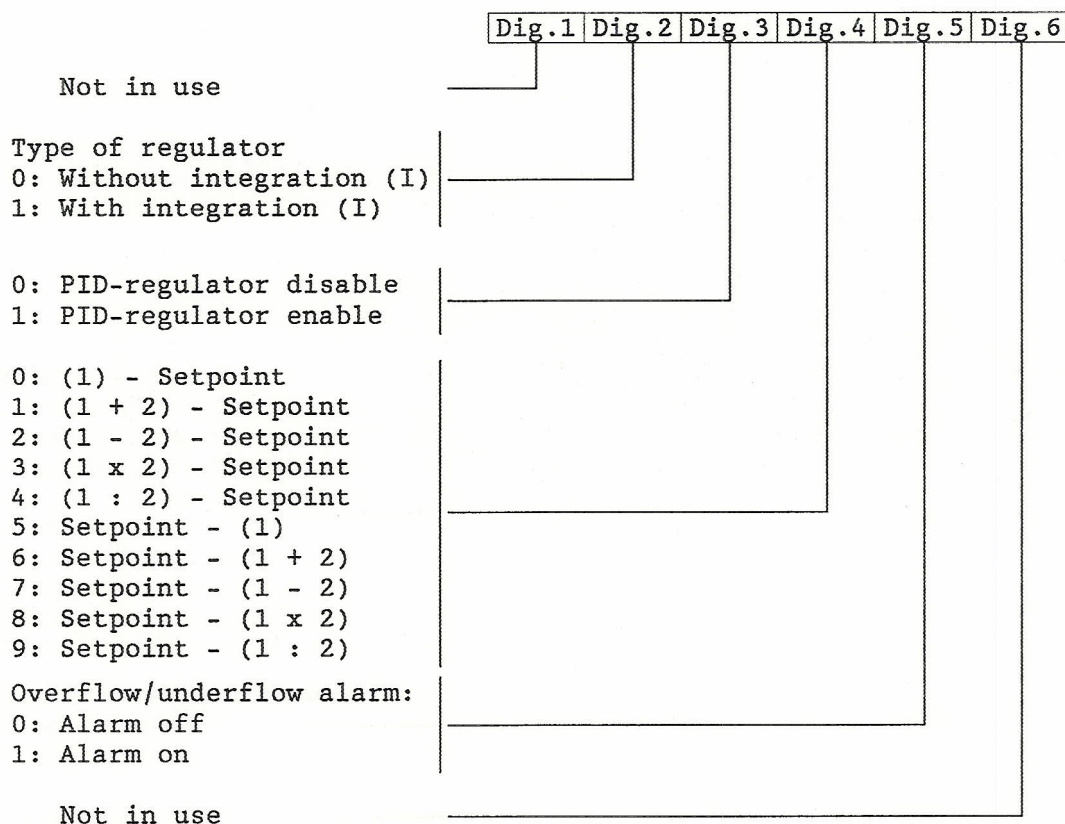
When the regulator is in "Manual" it is possible to preset an output value, using the display unit or P-net. When the regulator is reset to automatic, the control will continue from the set output signal. This facility is called "Bump-less transfer" and can be used, for example, for avoiding overshoots at startup when using a very slow control-loop.

The "Manual" facility can also be used for a quick zero-setting of the output signal, for example in an alarm situation.

Register 9: Code 9

```
                              |Dig.1|Dig.2|Dig.3|Dig.4|Dig.5|Dig.6|
                                 |_____|       |_____|     |_____|
Channel and                         |             |           |
register no. for Signal1   |_____|             |           |

Channel and                  |                    |           |
register no. for Signal2   |_____|           |

Channel and                  |                                |
register no. for Setpoint  |_____|
```

Register A: Code A

```
                              ┌Dig.1│Dig.2│Dig.3│Dig.4│Dig.5│Dig.6┐
          Not in use          ─────┘     │     │     │     │     │

Type of regulator                        │     │     │     │     │
0: Without integration (I) ──────────────┘     │     │     │     │
1: With integration (I)                        │     │     │     │

0: PID-regulator disable                       │     │     │     │
1: PID-regulator enable    ────────────────────┘     │     │     │

0: (1) - Setpoint                                    │     │     │
1: (1 + 2) - Setpoint                                │     │     │
2: (1 - 2) - Setpoint                                │     │     │
3: (1 x 2) - Setpoint                                │     │     │
4: (1 : 2) - Setpoint      ──────────────────────────┘     │     │
5: Setpoint - (1)                                          │     │
6: Setpoint - (1 + 2)                                      │     │
7: Setpoint - (1 - 2)                                      │     │
8: Setpoint - (1 x 2)                                      │     │
9: Setpoint - (1 : 2)                                      │     │

Overflow/underflow alarm:                                  │     │
0: Alarm off               ────────────────────────────────┘     │
1: Alarm on                                                      │

          Not in use       ──────────────────────────────────────┘
```

Using digit 2 of the code register (Reg.A) the regulator's integration time (I) can be disabled.

Digit 3 is used for enabling/disabling the PID regulator. If the regulator is not required, then digit 3 must be set to a 0, to avoid possible errors in the channel.

Digit 4 is used to select how the regulator should calculate the output value (as described above).

Digit 5 is used to select whether the overflow/underflow alarm is enabled/disabled (see Reg. F).

Reg. B: Xp

Xp is the PID regulator's proportional band. The proportional band for a regulator is the change required in the input signal to give a change from 0 to 100% in the output signal (without I and D). Xp is defined in the same units, as the input signal to the regulator.

Reg. C: Ti

Ti is the integration time constant for the regulator, which is the time it takes for the I-component of the regulator to give the same change in the output signal as that made by the P- component, following a permanent change of the input signal.

The I-effect in the regulator is minimised by setting Ti to a very high value. Ti is defined in seconds.

Reg. D: Td

Td is the regulator's differentiation time. The differentiation time for a regulator, is the time a constantly rising input signal must take to rise from 0-100% (equivalent to Xp) in order to give a constant output signal of - 100% (without P and I).

As the signal from the D-component in the regulator is negative when the input signal is rising, the component will act as a "brake" on the output signal.

The D-component in the regulator can be totally disabled by setting Td to 0. Td is defined in seconds.

If problems arise in setting the three parameters P, I and D, please refer to technical literature on the subject.

Reg. F: CHError

The CHError register indicates whether there are any current errors on the channel, or if there is a general error on the module, which could result in the channel being faulty.

```
                    128  64  32  16   8   4   2   1  Dec. ┐
                     80  40  20  10   8   4   2   1  Hex. ├─Read out
                   ┌───┬───┬───┬───┬───┬───┬───┬───┐      ┘
                   │ 7 │ 6 │ 5 │ 4 │ 3 │ 2 │ 1 │ 0 │ bit
                   └───┴───┴───┴───┴───┴───┴───┴───┘

General error   ─────────┘

Overflow        ─────────────┘

Underflow       ─────────────────┘

Error on Ch.            │
for Signal1             │─────────────────────┘

Error on Ch.            │
for Signal2             │─────────────────────────┘

Error on Ch.            │
for Setpoint            │─────────────────────────────┘
```
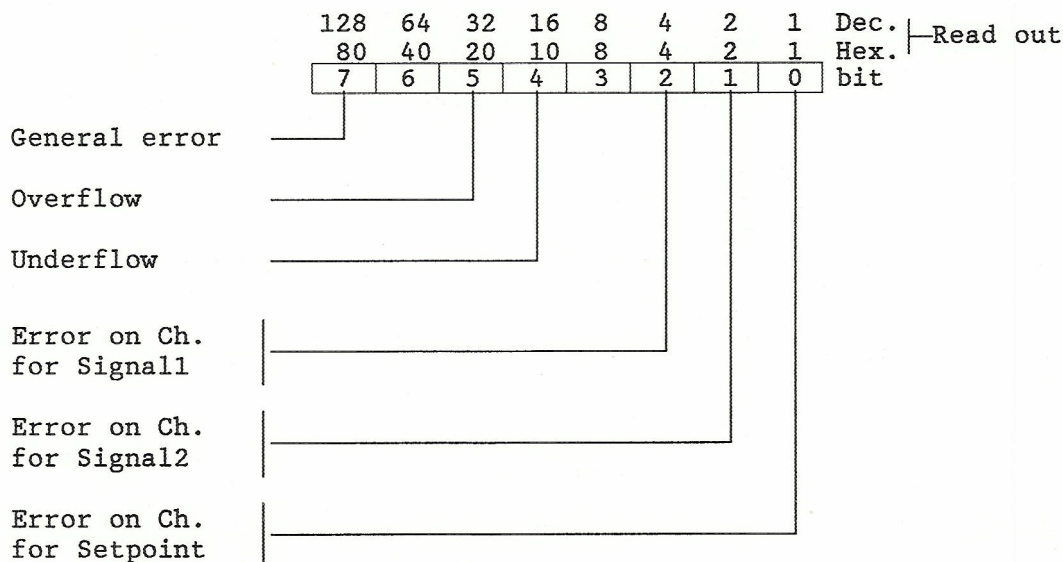
If bit 7 (general error) is set then the other bits are insignificant, as the general error can lead to other errors on the channels (see Service Channel in section 8).

Bit 5 (overflow) is set when a result is above 100%.

Bit 4 (underflow) is set if a result falls below 0%.

Bit 0-2 indicates whether there is an error on the channel from where Setpoint, signal 2 and signal 1 respectively, have been taken.

4.1. Calculation of Actual Value.

As previously mentioned, Signal 1 and Signal 2 can be added, subtracted, multiplied or divided with each other. It is therefore possible to choose various forms of control. The result of the chosen calculation will be stored in the Actual register (reg.3).
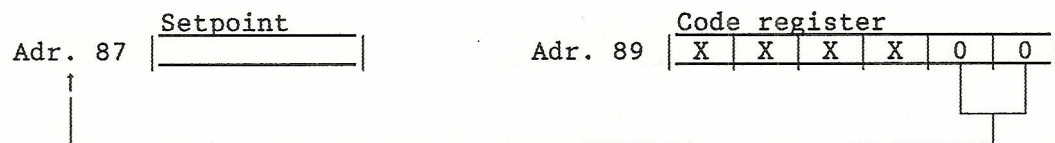
As an alternative, one can store signal 1 directly in the Actual register.

To select the input signals to the PID regulator (signal 1 and signal 2), the code register must be programmed as shown in the example below.

```
            Temperature                         PID-regulator (Code9)
Adr. 10 |_____|                    | 1 | 0 | 2 | 0 | X | X |
        |                                        |___|___|   |___|
        |_____ Value for Signal1 _____|           |
                                                              |
            Current (4-20mA)                                  |
Adr. 20 |_____|                                    |
        |                                                     |
        |_____ Value for Signal2 _____|
```
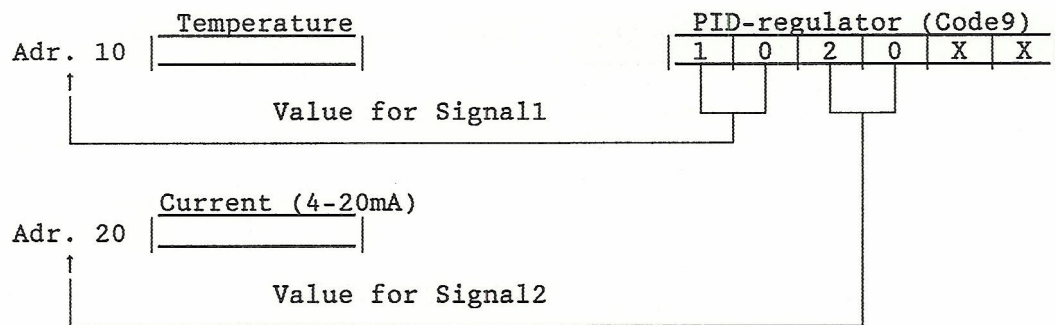
The regulator utilises values from the Actual register and a selected value in the Setpoint register.

The address from which the Setpoint collects data, is selected through the code register (reg.9) as shown in the example below.

```
            Setpoint                            Code register
Adr. 87 |_____|         Adr. 89   | X | X | X | X | 0 | 0 |
        |                                                      |___|
        |_____|
```

Note: When pointing at address 00 by means of a code register, it is possible to write directly in the indirect register, i.e. writing "0" in channel number and register number for Setpoint (digit 5 and 6), makes it possible to write directly in the Setpoint register (Reg. 7).

When a difference occurs between the Actual value and Setpoint, it can be corrected by making the output value rise or fall. This choice is made using an inverter, selected by a code register change.

Using digit 4 in the code register (reg.A), signal 1 and signal 2 can be added, subtracted, multiplied or divided, and defines whether inversion should be used.

```
                                      Code register
                          Adr. 6A | 1 | 2 | 3 | 4 | 5 | 6 | digit

0: (Signal1) - Setpoint
1: (Signal1 + Signal2) - Setpoint
2: (Signal1 - Signal2) - Setpoint    **
3: (Signal1 x Signal2) - Setpoint
4: (Signal1 : Signal2) - Setpoint


5: Setpoint - (Signal1)
6: Setpoint - (Signal1 + Signal2)
7: Setpoint - (Signal1 - Signal2)    ***
8: Setpoint - (Signal1 x Signal2)
9: Setpoint - (Signal1 : Signal2)
```

The result of the expression in the brackets will be stored in the Actual register.

If the Actual value is greater than the Setpoint value, the following is true:-

** These 5 functions will result in a rising output value (e.g. cooling), i.e. the invertor is disconnected.

*** These 5 functions will result in a falling output value (eg.Reheating), i.e. the inverter is connected.

4.2. Example of simple regulation.

This type of regulation can, for example, be used in processes where one wishes to regulate the temperature in a container.

In addition to the PID-regulator, one uses the current output and one of the two analog inputs, programmed to measure temperature.

The current output controls a valve which, by means of a steam pipe, regulates the temperature in the container.

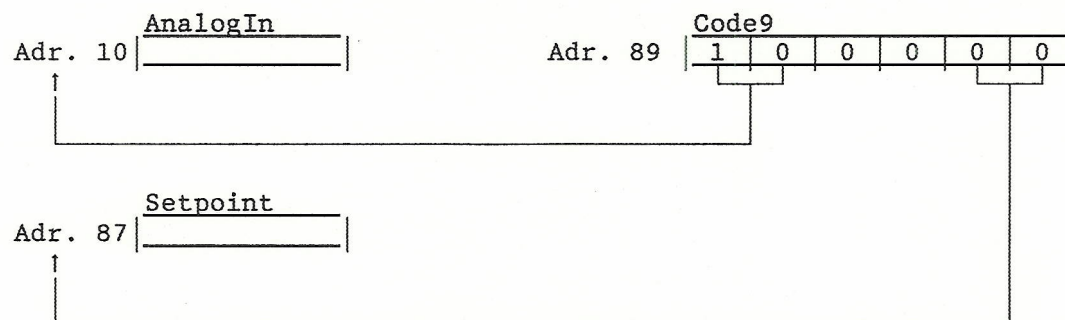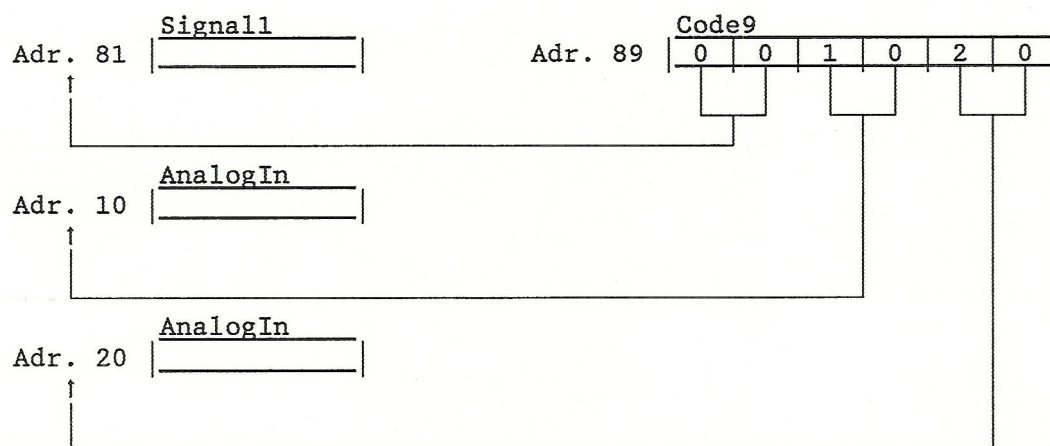The analog input measures the temperature in the container by means of a temperature detector (Pt-100).
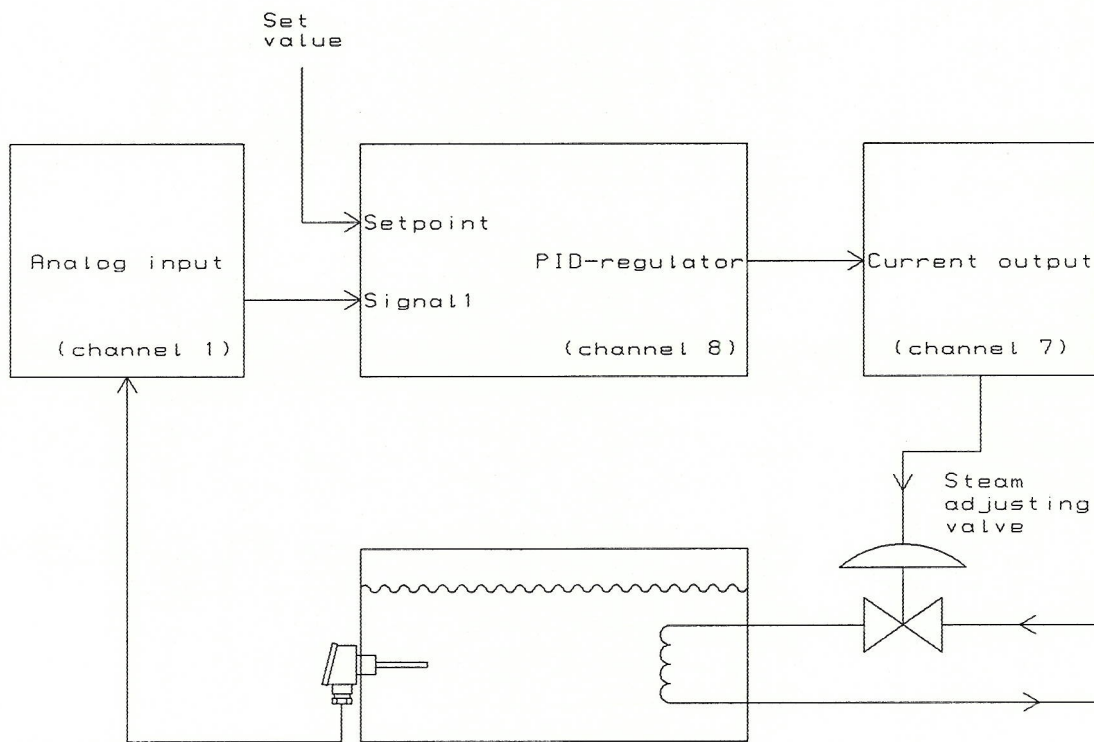

Fig. 4.2.a.

Setting a code register.

The analog input is programmed to measure temperature. The result of the measured temperature is placed in address 10, from where the PID-regulator can fetch its input signal. Apart from the temperature input signal the regulator needs a setpoint-value, which is inserted in the SetPoint register. These two input signals to the regulator are selected in the code register (Reg. 9), which is programmed as follows:

```
        AnalogIn                       Code9
Adr. 10|_____|      Adr. 89 | 1 | 0 | 0 | 0 | 0 | 0 |
   |                              |__|        |__|
   |_____|              |
                                                |
        Setpoint                                |
Adr. 87|_____|                            |
   |                                            |
   |_____|
```

The other code in the PID-regulator channel can, for example, be programmed as follows:

```
    CodeA
| 0 | 1 | 1 | 5 | 1 | 0 |
```

Then the three parameters P, I and D (Tp, Ti and Td) must be inserted.

The output value for the PID-regulator is placed in the output register (Reg. 0), ie, the current output must be programmed to follow this value.

4.3. Example of sequence regulation.

This type of regulation can, for example, be used in processes, where one wishes to make a temperature in a container follow the temperature in another container, perhaps with a set difference.

Apart from the PID-regulator, one uses the current output and the two analog inputs, programmed to measure temperature.

The current output controls a steam valve which regulates the temperature in the container.
The two analog inputs measure the temperature in each container by means of temperature detectors (Pt-100).
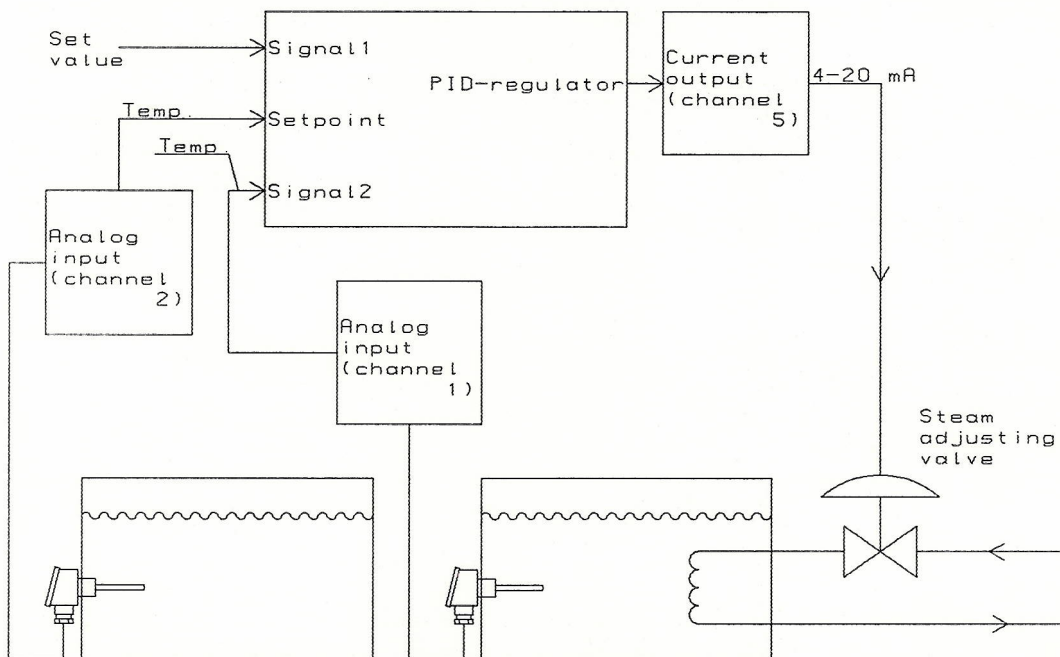
Fig. 4.3.a.

Setting code registers.

The analog inputs are programmed to measure temperature. The results of the temperatures measured will be placed in address 10 and 20, from where the PID-regulator fetch its input signals. Apart from the two temperature input signals, the regulator needs a difference setting which is inserted in the signal 1 register. These three input signals to the regulator are pointed to by the code register (Reg. 9), which is programmed as follows:

```
        Signal1                          Code9
Adr. 81 |_____|            Adr. 89 | 0 | 0 | 1 | 0 | 2 | 0 |
        ↑                                     |       |       |
        |_____|       |       |
                                                      |       |
        AnalogIn                                      |       |
Adr. 10 |_____|                                    |       |
        ↑                                             |       |
        |_____|       |
                                                              |
        AnalogIn                                              |
Adr. 20 |_____|                                            |
        ↑                                                     |
        |_____|
```

The other code in the PID-regulator channel can, for example, be programmed as follows:

```
                          CodeA
                Adr. 6A | 0 | 1 | 1 | 6 | 1 | 0 |
```
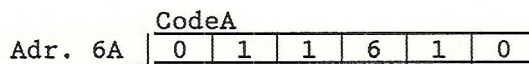
Then the three parameters P, I and D (Tp, Ti and Td) must be inserted.

The output value for the PID-regulator is placed in the output register (Reg. 0), ie, the current output must be programmed to follow this value.

4.4. Example of pressure difference regulation.

This type of regulation can, for example, be used in processes, where, one wishes to control pressure by means of the difference between two pressures.

In addition to the PID-regulator, one uses the current output and the two analog inputs, programmed to measure current (4-20 mA).

The current output controls a valve which directly controls the pressure in the pipe.

The two analog inputs measure current (4-20 mA), which represents the pressure in two different pipes.
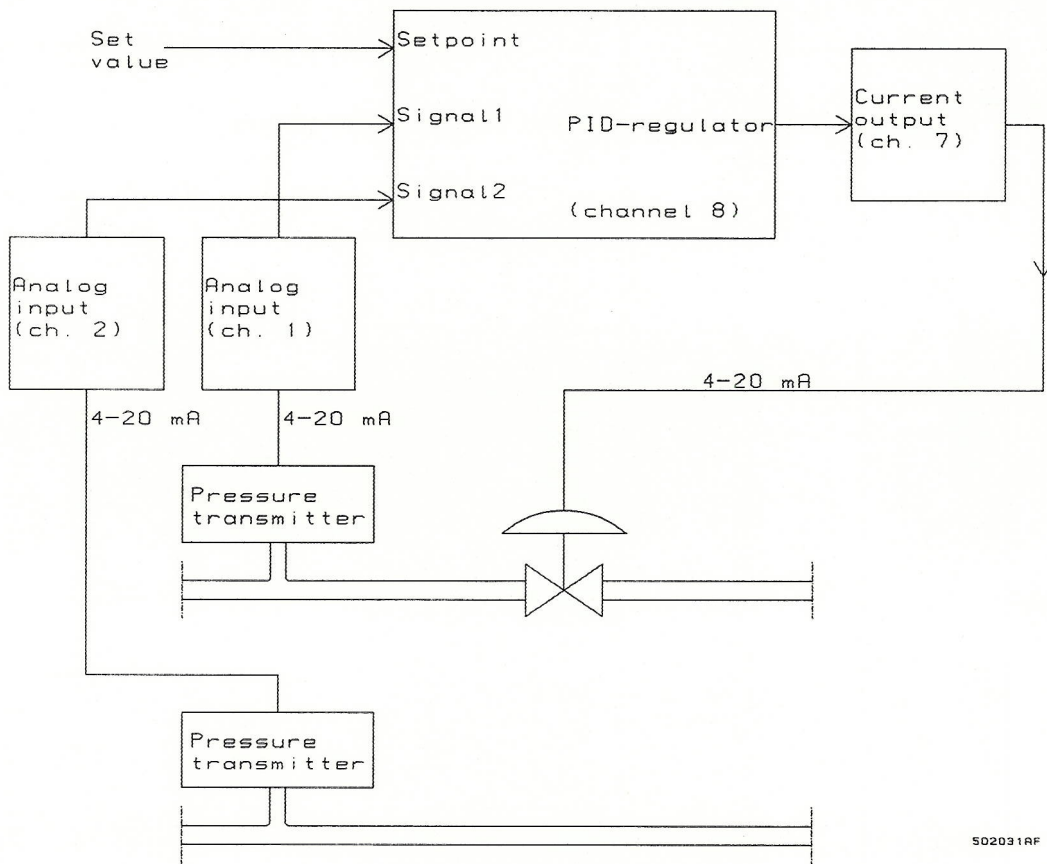
Fig. 4.4.a.

Setting code registers.

The analog inputs are programmed to measure current (4-20 mA). The result of the measured currents will be placed in address 10 and 20, from where the PID-regulator can fetch its input signals. In addition to the two current signals, the regulator needs a setpoint value which is inserted in the SetPoint register. These three input signals to the regulator are pointed to by a code register (Reg. 9), programmed as follows:
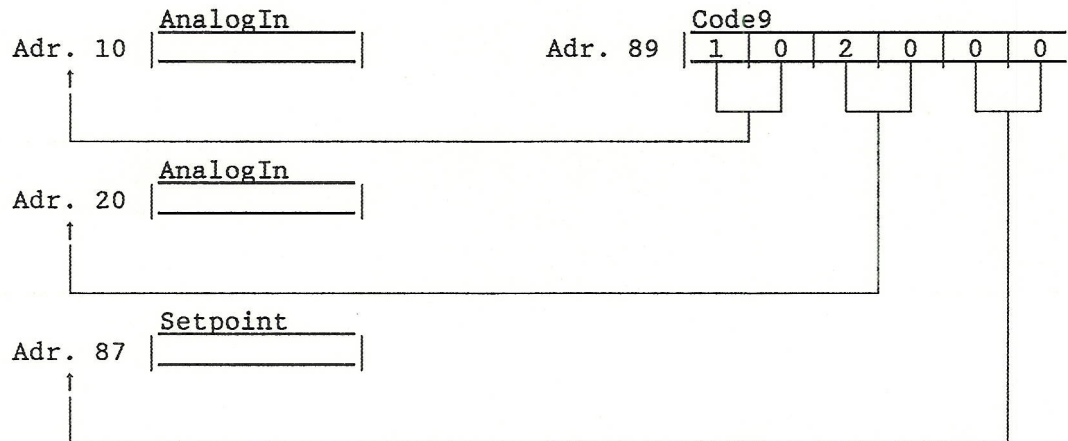
```
         AnalogIn                          Code9
Adr. 10 |_____|         Adr. 89 | 1 | 0 | 2 | 0 | 0 | 0 |
   ↑_____|_____|_____|

         AnalogIn
Adr. 20 |_____|
   ↑_____|

         Setpoint
Adr. 87 |_____|
   ↑_____|
```

The other code on the PID-regulator channel can, for example, be programmed as follows:

```
                  CodeA
Adr. 6A | 0 | 1 | 1 | 7 | 1 | 0 |
```

Then the three parameters P, I and D (Tp, Ti and Td) must be inserted.

The output value for the PID-regulator will be placed in the output register (Reg. 0), ie, the current output must be programmed to follow this value.

4.5. Example of scaling by means of PID regulator.

If it is required to scale a measured value, it can be done with the PID regulator. For example, to multiply a frequency reading with a set scaling value.
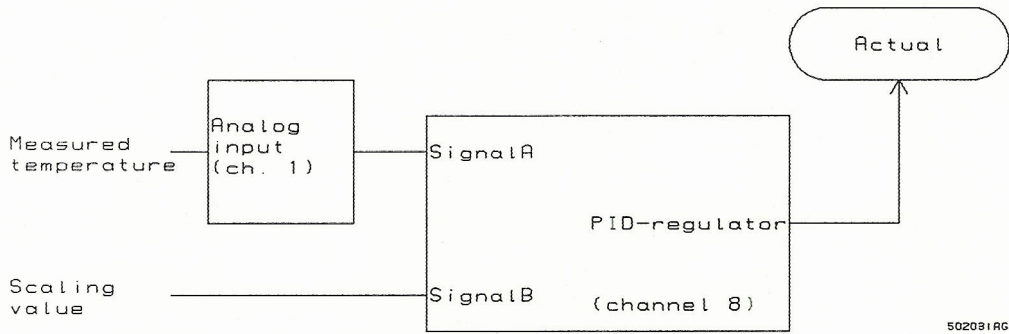
Fig. 4.5.a.

Setting Code Register

The analog input (channel 1) is programmed to measure temperature. The result of the measured temperature will be placed in address 10, from where the PID regulator can fetch one of its input signals. Apart from the temperature input signal, the regulator needs a scaling value, which is inserted in the Signal2 register. These two input signals to the regulator are pointed to by a code register (Reg. 9), which is programmed as follows:
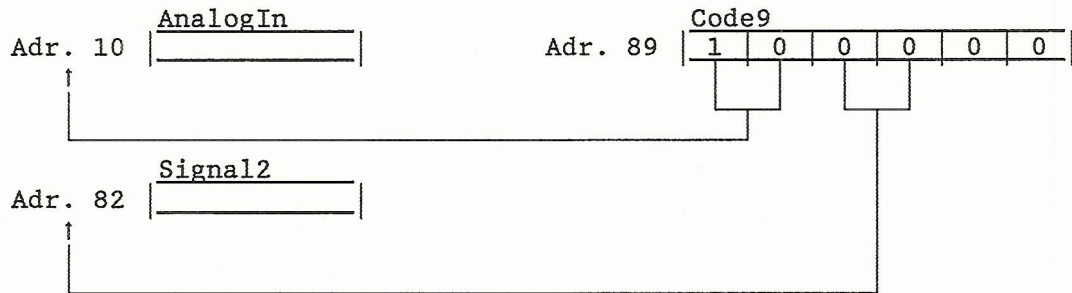
The other code register on the PID-regulator can be programmed as follows:

CodeA

0	1	1	3	0	0

The final value after scaling can be read in the Actual register (reg. 3).

4.6. Example of proportional regulation.

This type of control can be used for example, in processes whereby one wishes to control a flow by means of the ratio between two flows.

In addition to the PID-regulator, the modules current output is used.

The measured results from the two flow transmitters are transferred by means of computer via the P-net (this data could also be transferred, for example, by means of a current signal).

The current output controls a valve which determines the flow through the pipe.
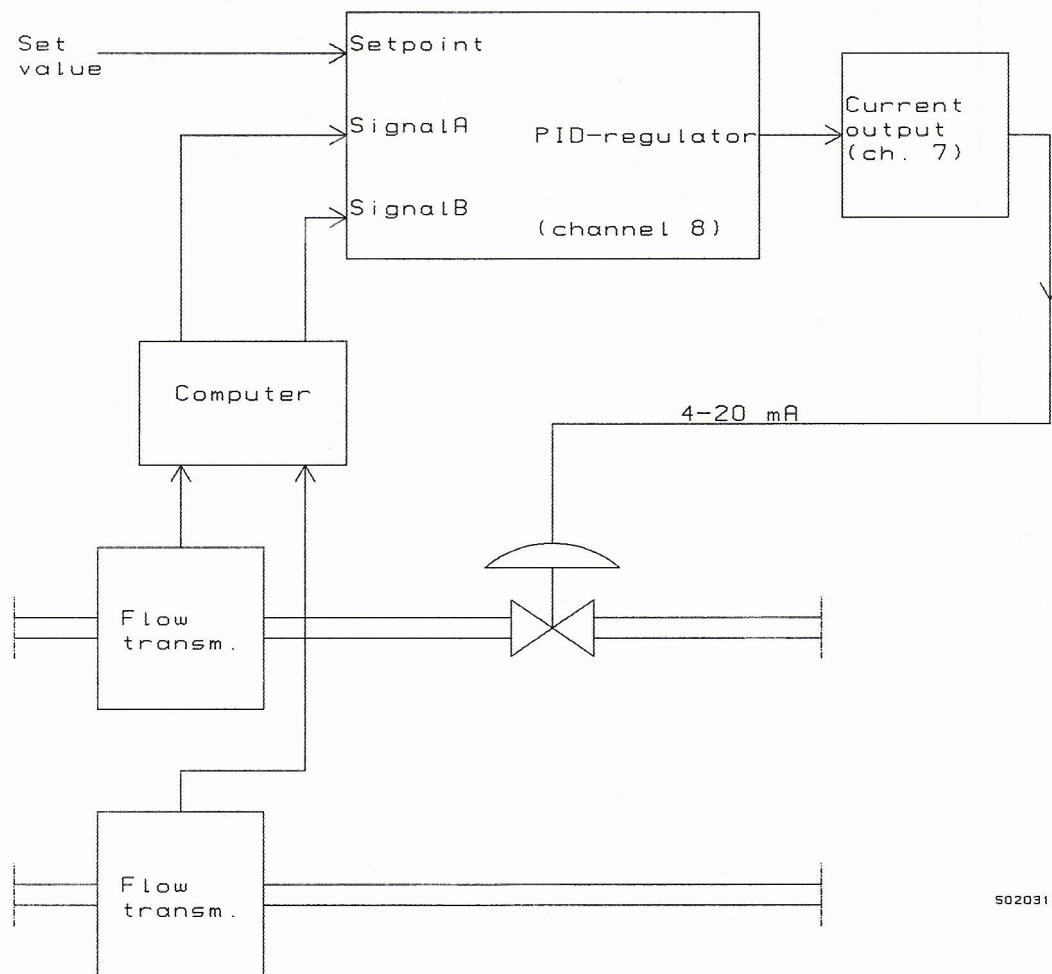
Fig. 4.6.a.

Setting the Module

The two measured results from the flow transmitters are placed by the computer, each in its own register. In addition, the regulator needs a fixed setpoint value, which is inserted in the SetPoint register. These three input signals to the regulator, are pointed to in a code register (Reg. 9), which is programmed to do the following:
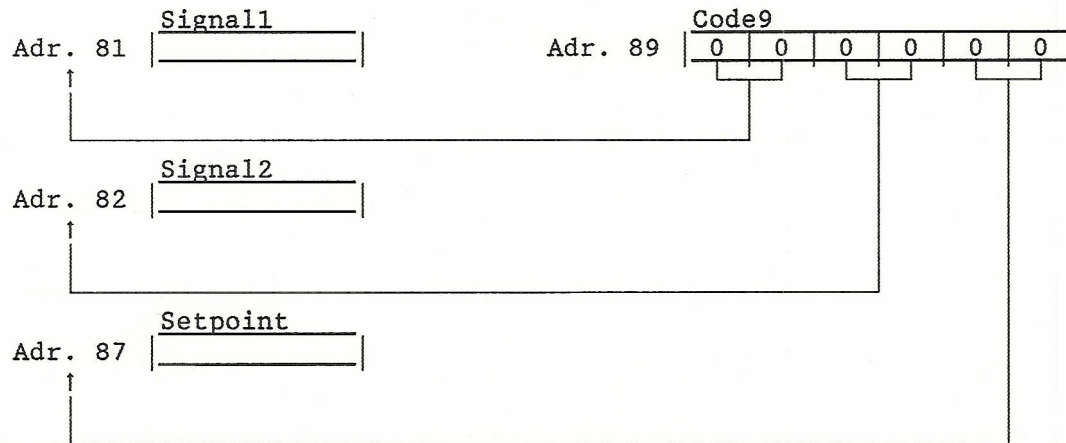
The other code on the PID-regulator channel can be programmed as follows:

```
CodeA
| 0 | 1 | 1 | 9 | 1 | 0 |
```

The three parameters, P, I and D (Tp, Ti and Td) can now be set.

The output value from the PID regulator is placed in the output register (reg.0), i.e. the current output must be programmed to follow this value.

5. Limit Switch / Batching Channel.

The module has two limit switch/batching channels (channel 9 and A). The channels can, for example, be used as alarm detectors from two measurement values (Actual1 and Actual7), where both values can be defined by the user. Each of the two channels can control two flags, each with each own corresponding Offset value.
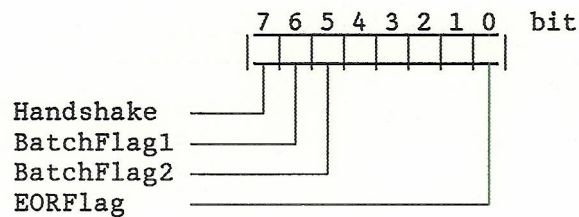
Diagram showing limit switch/batching channel:
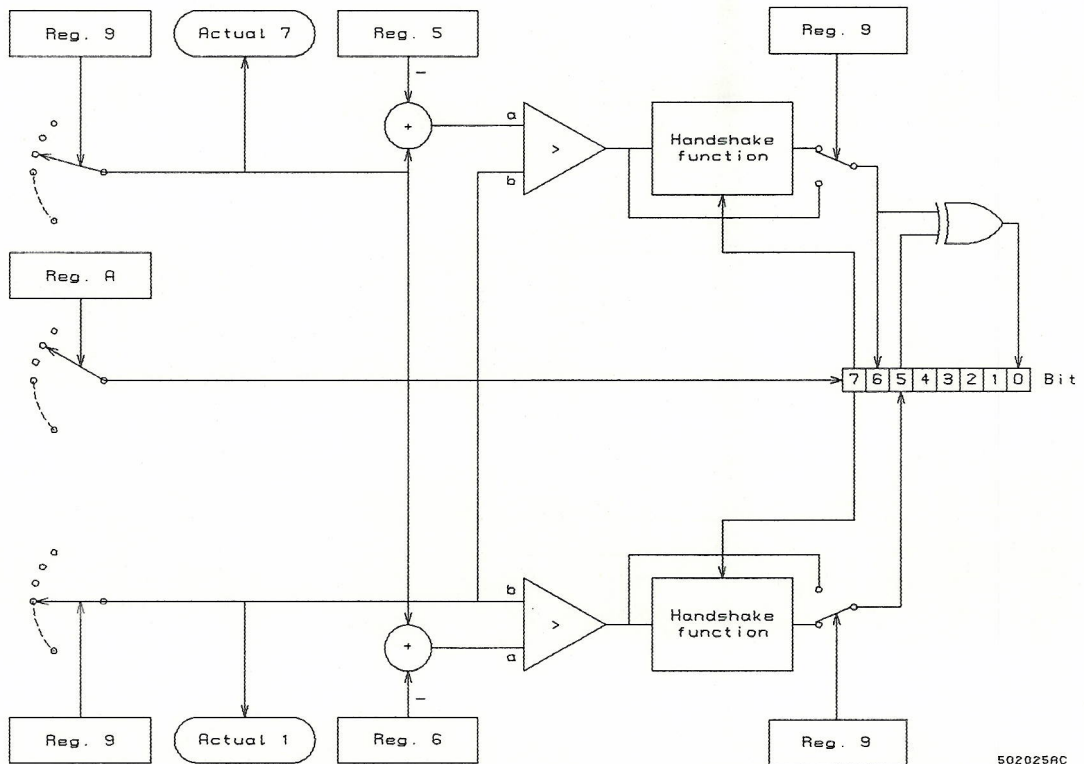
Registers for Limit Switch / Dosage Channel (channel 9-A)

Reg.no.	Register name	Write	Storage medium	Number of bytes	Read out
0	FlagReg	X	R	1	Binary
1	Actual1 *	X	BR	4	Decimal
2					
3					
4					
5	ActOffset1	X	BR	4	Decimal
6	ActOffset2	X	BR	4	Decimal
7	Actual7 *	X	BR	4	Decimal
8					
9	Code9	X	E	4	Hexadec.
A	CodeA	X	E	4	Hexadec.
B					
C					
D					
E					
F	CHError		R	1	Hexadec.

*   Indirect register

<u>Reg. 0: FlagReg</u>

The flag register contains Batch Flag 1 and Batch flag 2, which are controlled automatically. These two flags works as the output state for the channel, ie, a digital output can follow one of these flags. An Exclusive Or function on the two Batch-flags is automatically carried out, and is placed in bit 0. Furthermore a Handshake bit exists, which is used to start batching.

```
            7 6 5 4 3 2 1 0  bit
           | | | | | | | | |

Handshake  ────────┘ │ │       │
BatchFlag1 ──────────┘ │       │
BatchFlag2 ────────────┘       │
EORFlag    ────────────────────┘
```

Reg. 1: Actual1. *

In this register, the Actual1 value, pointed to by means of the code, can be read.

If the code register points to address 00, then the register can be written to.

Reg. 5: ActOffset1.

The offset value for controlling Batch-Flag1 is inserted in this register.

Reg. 6: ActOffset2.

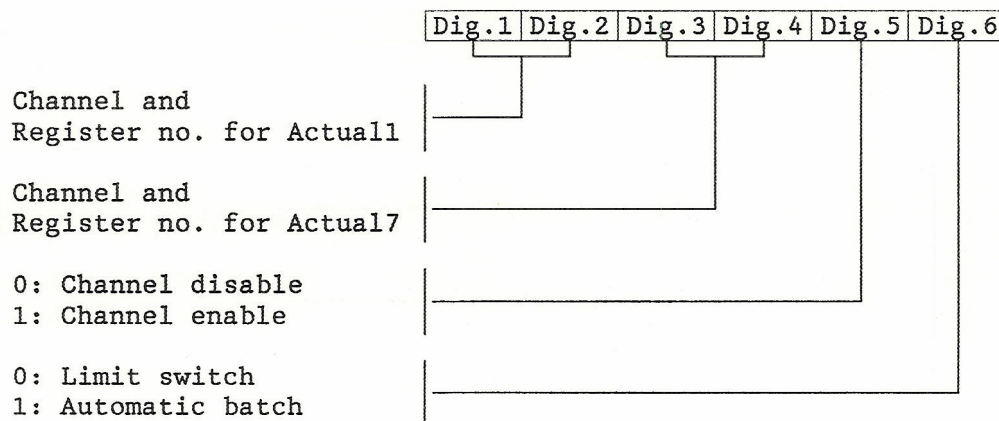The offset value for controlling Batch-Flag2 is inserted in this register.

Reg. 7: Actual7. *

The Actual7 value chosen by means of the code (Reg. 9) can be read in this register.

If the code register points to address 00, one can write into the register.

Reg. 9:Code9.

Code register for limit switch/batch channels (Reg. 9).

```
                              |Dig.1|Dig.2|Dig.3|Dig.4|Dig.5|Dig.6|

        Channel and            |_____|
        Register no. for Actual1|

        Channel and            |_____|
        Register no. for Actual7|

        0: Channel disable     |_____|
        1: Channel enable      |

        0: Limit switch        |_____|
        1: Automatic batch     |
```

An indirect register must not be pointed to by means of the first four digits in the code as this could lead to errors being generated.

The channel can be inhibited by use of digit 5. If it is not required to use the channel, a 0 must be inserted in digit 5 in order to avoid errors being generated on the channel.

Digit 6 determines whether the channel will operate as a switch of for batching purposes.

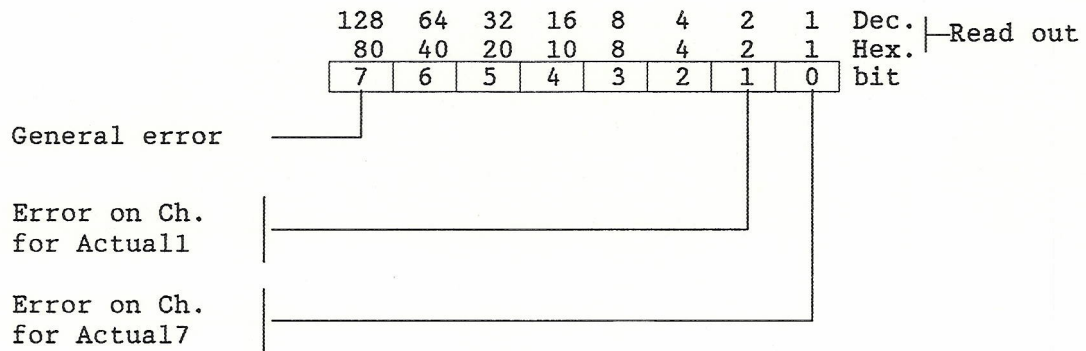If address 00 is pointed to by digits 1-4, then the corresponding indirect register can be directly written to.

Reg. A: CodeA.

```
                              ┌─────┬─────┬─────┬─────┬─────┬─────┐
                              │Dig.1│Dig.2│Dig.3│Dig.4│Dig.5│Dig.6│
                              └─────┴─────┴─────┴──┬──┴─────┴──┬──┘
                                                  └───────────┘
        Handshake input            ┌──────────────────────────┐
        Bit no., channel no.,      │
        Register no.               │
```
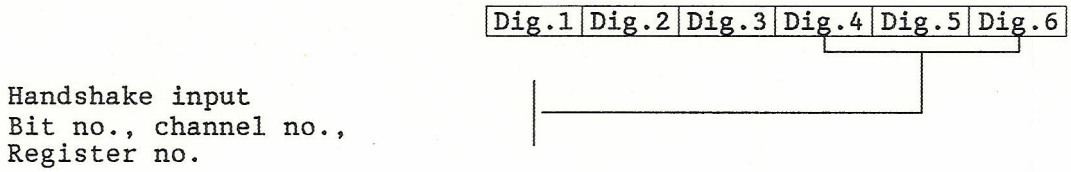
If it is required to control the Handshake flag by means of another bit, for example an input flag on a digital input, then this can be done by pointing to the selected bit, ie, bit No, channel No. and register No must be known. If this function is not desired, a 0 must be inserted in digit 4, 5 and 6, then the handshake flag can be controlled directly in FlagReg (Reg. 0).

Reg. F: CHError.

The CHError register shows whether there is an error on the channel concerned, or whether there is an error on the module as a whole, which might lead to the channel being defective.

```
                   128  64  32  16   8   4   2   1  Dec. ┐
                    80  40  20  10   8   4   2   1  Hex. ├─Read out
                   ┌───┬───┬───┬───┬───┬───┬───┬───┐     │
                   │ 7 │ 6 │ 5 │ 4 │ 3 │ 2 │ 1 │ 0 │ bit ┘
                   └─┬─┴───┴───┴───┴───┴───┴─┬─┴─┬─┘
  General error ─────┘                       │   │
                                             │   │
  Error on Ch.     ┌─────────────────────────┘   │
  for Actual1      │                              │
                                                  │
  Error on Ch.     ┌──────────────────────────────┘
  for Actual7      │
```

If bit 7 (general error) is set, then the other bits are insignificant, as the general error can be caused by error codes on the other channels (see Service channel).

Bit 1 is set if an error occurs on the channel from which the Actual1 value is fetched.

Bit 0 is set if an error occurs on the channel from which the Actual7 value is fetched.

5.1. Limit Switch, function.

The channel can be used as a limit switch to monitor a fluctuating signal (eg, a temperature measured on one of the two analog input channels).
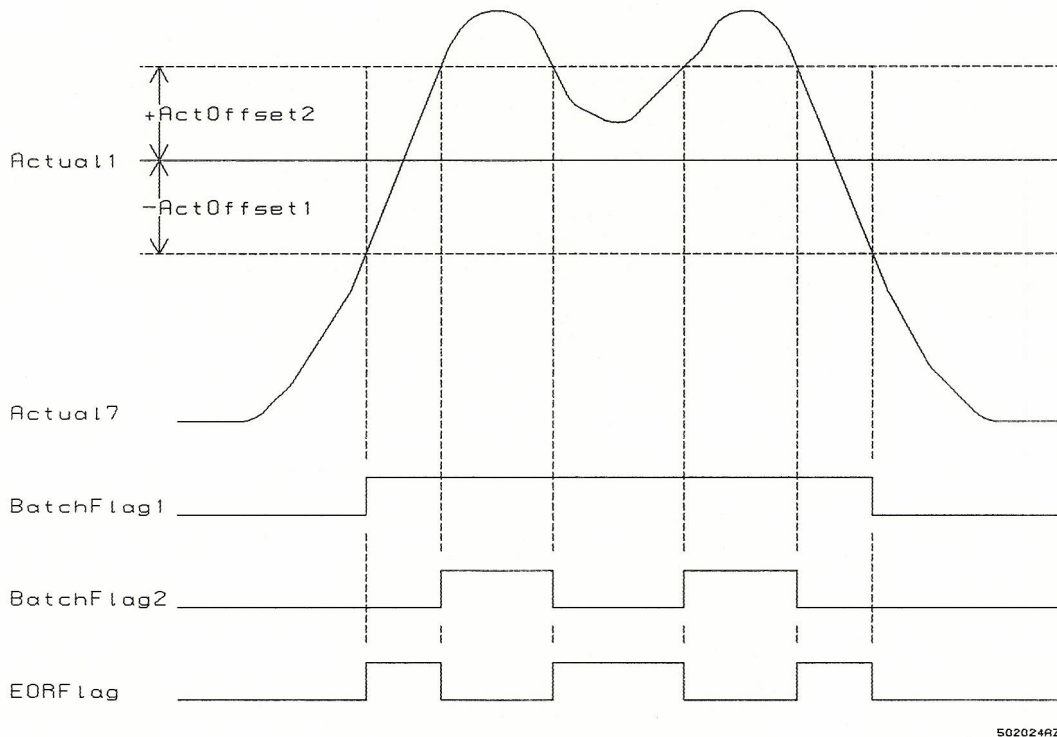
5.1.1. Limit Switch, example.

Fig. 5.1.1.a. shows an example of how this limit switch could be used.

By means of the code register (Reg. 9) the Actual7 value is given as the fluctuating input signal. Actual1 is defined as a fixed setpoint; this is done by pointing to address 00, and writing directly in the Actual1 register. Furthermore a tolerance around the setpoint value can be set by means of the two offset values, because ActOffset1 is made negative and ActOffset2 positive. Thus each of the two batch flags corresponds to an offset value.

BatchFlag1 will be OFF until Actual7 exceeds Actual1 + ActOffset1. Likewise, BatchFlag2 will be OFF until Actual7 exceeds Actual1 + ActOffset2. The two offset values are thus used to determine how a time must elapse from when the Actual7 value reaches the setpoint (Actual1); so by making one offset value negative, one can determine how long a time must elapse from when the BatchFlag changes, until the Actual7 value reaches the setpoint (Actual1).

If one wishes to test whether the Actual7 value lies within or outside the limits, one can use the EORFlag. The EORFlag is set when the Actual7 value lies within the limits.

Fig. 5.1.1.a.

5.1.2. Limit Switch with positive offset values, example.

One can also select to make both offset values positive, as shown in fig 24.
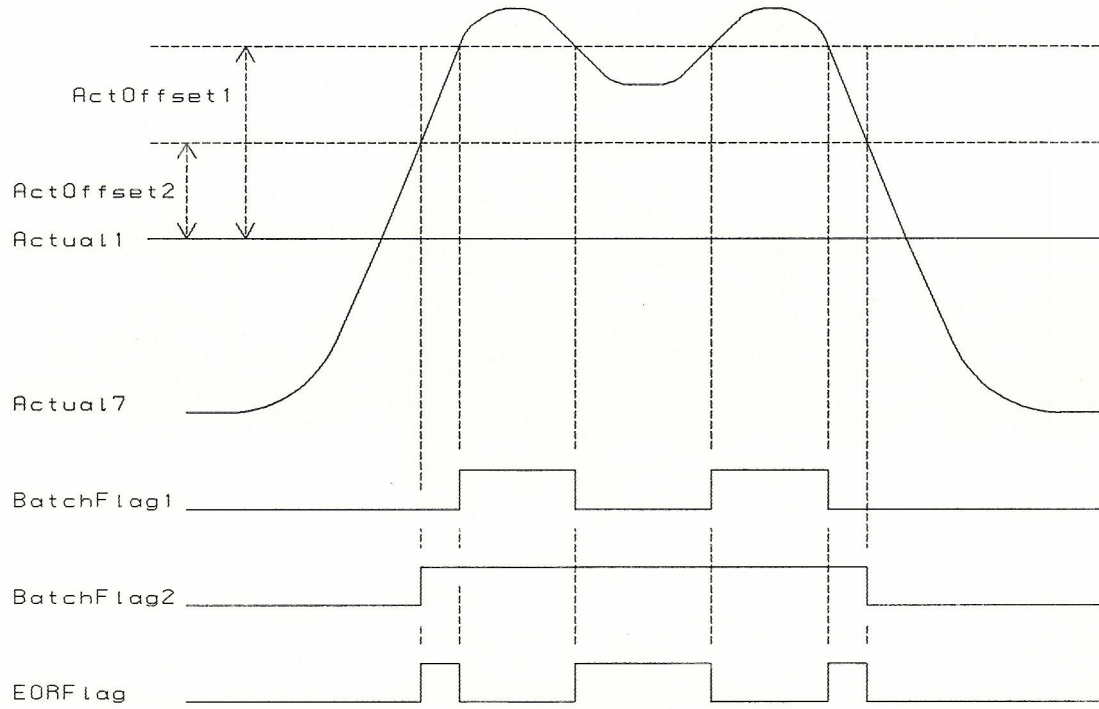
Fig. 5.1.2.a.

5.2. Batch run.

This channel can also be used for batching purposes. The batch runs until the measured value (Actual1) equals the desired number (Actual7). A batch channel can control two flags (in Reg. 0) from the same measured value and the same setpoint, but with two different setpoint Offset values (ActOffset1 and 2) - one for each flag. These offset values are used for example, when batching with two speeds or when dosing with a slow value. When the number still to come equals the offset value, the batch will be stopped.

The batch is set up by supplying the channel with its input signals (Actual1, Actual7, ActOffset1 and ActOffset2), then the batch can be started by setting the Handshake flag. This flag will be cleared automatically, and only then will the two batch flags be stable, ie, they can be read via the P-net.

When the two batchflags have been cleared, they cannot be set again unless the batch is restarted by setting the Handshake flag.

BatchFlag1 will be ON until Actual1 exceeds Actual7 - ActOffset1. Batchflag2 will be ON until Actual1 exceeds Actual7 - ActOffset2. The two offset values must therefore be positive numbers.

Below is shown an example of a batching system for dosing.
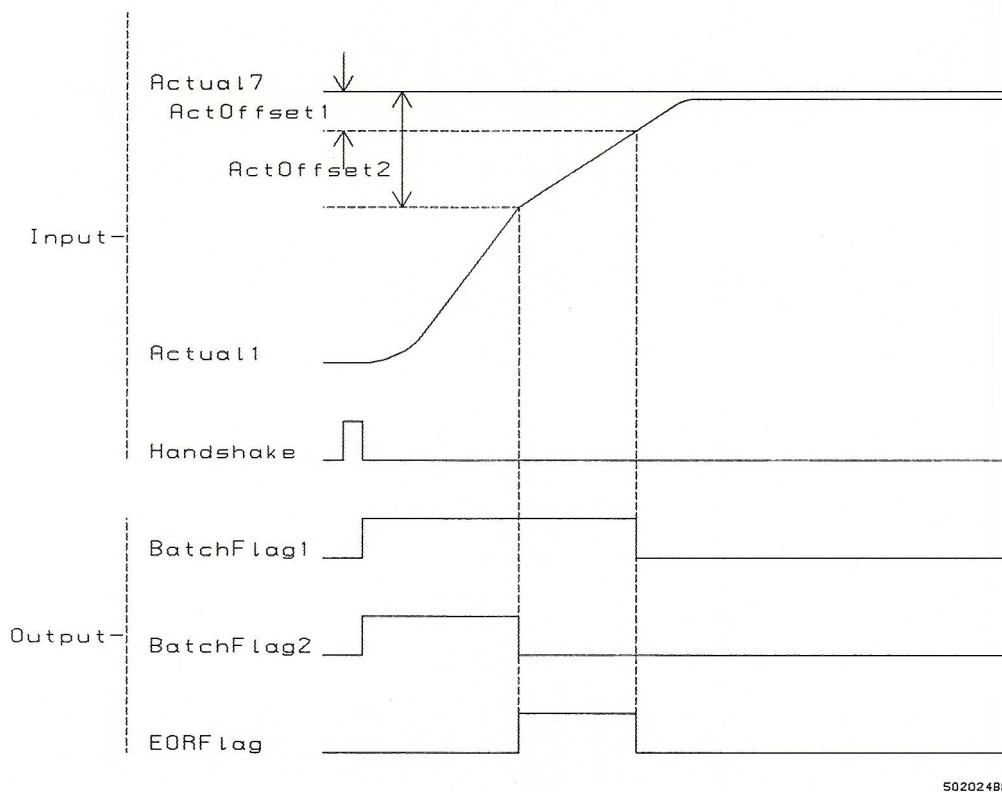
Fig. 5.2.a.

Setting the code register.

In order to define the Actual1 and Actual7 values, the desired values must be pointed to by means of a code register, as shown in the example below.

The Actual1 value can, for example, be a temperature, measured by means of an analog input (channel 1), and Actual7 can be a fixed defined value.

```
                Temperature                            Code9  | 1 | 0 | 0 | 0 | 1 | 1 |
Adr. 10  |_____|
       ↑
       |_____|


                Setpoint
Adr. 97  |_____|
       ↑
       |_____|
```
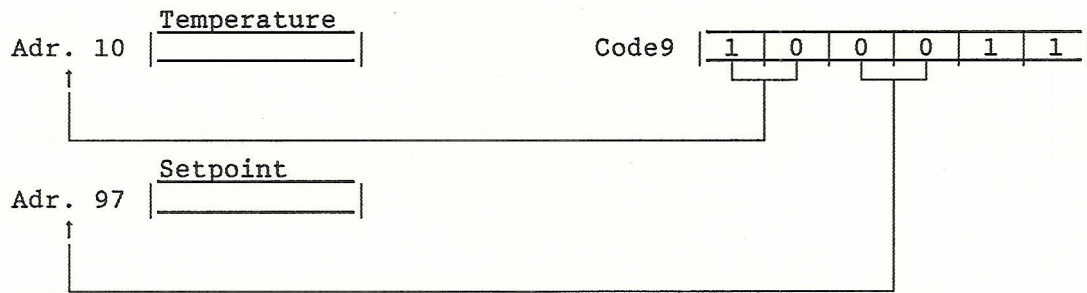
6.   Service Channel (Channel 0).

The service channel contains registers which are relevant to all other channels. The service channel contains data about the module type and version No. of the program, both types of data being important to the P-net operation. The service channel monitors the microprocessor, collects error codes from other channels, and also controls the monitor facility called a watch-dog.

In the case of external errors, (for example a break in the communication cable, or a loss of supply voltage in the control computer) this would result in the module not being called by the P-net, and the Watch-dog will automatically inhibit all outputs, and cause the current output to decrease to 4 mA.

The user is able to decide how much time there should be between call-ups to the module via P-net. If this time is exceeded the watch-dog facility comes into operation as described above. At the same time, an error will be registered in the general error code register and in the error code registers for the other channels.

Registers in the Service Channel.

Reg.no.	Register name	Write	Storage medium	Number of bytes	Read out
0					
1	DeviceType		P	2	Decimal
2	PrgVers		P	2	Decimal
3	Error3	X	R	1	Hexadec.
4					
5					
6					
7	WDTimer	X	R	2	Decimal
8	WDPreset	X	E	2	Decimal
9	Code9	X	E	4	Hexadec.
A					
B					
C					
D					
E					
F	ErrorF		R	1	Hexadec.

Reg. 1: Device Types

This register contains the module type number, e.g. 1611.

Reg. 2: PrgVers

This register contains the program version number, e.g. 8601

Reg. 3: Error3

This register contains the highest error code detected since the last error code register reading. Even if the error was transitory, the code will be retained until the register is interrogated. The error code meanings are the same for register 3 and register F.

By inserting an FF in the register, the module will carry out a Reset.

Reg. 7: WDTimer

When the module is communicated with via P-net, the timer is automatically set to the value from the preset register (reg.8). If the timer reaches zero before it is set again, the watch-dog flag will be reset, causing the outputs to switch OFF and the current output to revert to 4 mA.

Reg. 8: WDPreset Register

The maximum time allowed between two call-ups to the module, without the watch-dog being activated, is set in this register.

Reg. 9: Code 9

This register enables the watch-dog facility to be selected.

The register contains 6 digits.

Digits 1-5 are not used.

Digit 6 defines whether the watch-dog is selected or not.

Dig.1	Dig.2	Dig.3	Dig.4	Dig.5	Dig.6

0: Watchdog selected
   (watchdog flag
   controlled by WDTimer)
1: Watchdog off
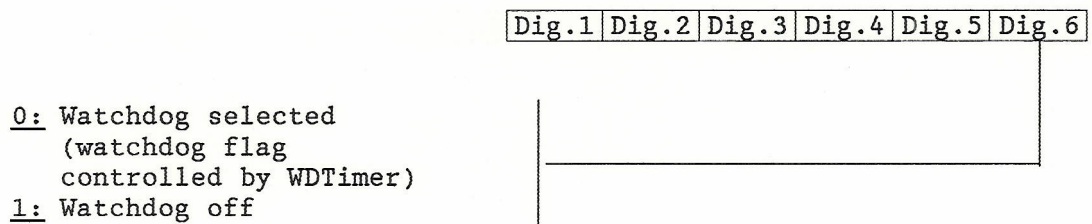
Register F: ErrorF

When faults occur, this error code register will contain the last error code while the fault exists, but if the fault disappears the code will not be retained in the register (unlike register 3).

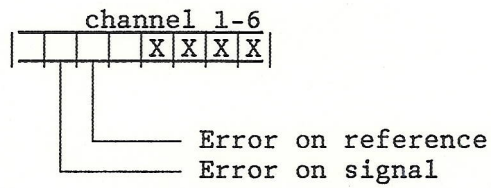The error code register contains 8 bits which need to be converted to a (hexadecimal) number between 0 and 93.

The error codes can be split into two groups. Numbers below 80 pertain to errors on individual channels, and 80-93 apply to errors on the module as a whole.

```
Bit no.   7 6 5 _ _ _ _
         |0|X|X|_ _ _ _
```

If the most significant bit (bit7) is "0", then an error only applies to a single channel.

```
       Bit no.   7 6 5 4 3 2 1 0
                | | | | | | | | |

Error type  _____|   |_____  Channel no
```

Channel 1 - 6:

```
          channel 1-6
 |  |  |  |X|X|X|X|
          |  |
          |  └──────── Error on reference
          └─────────── Error on signal
```

Channel 7:

```
          channel 7
 |  |  |  |X|X|X|X|
          |  |  |
          |  |  └───── Error on other channel
          |  └──────── Underflow
          └─────────── Overflow
```

Channel 8:

```
          channel 8
 |  |  |  |X|X|X|X|
          |  |  |
          |  |  └───── Error on other channel
          |  └──────── Underflow
          └─────────── Overflow
```

Channel 9 & A: *

```
          channel 9/A
 |  |  |  |X|X|X|X|
          |
          └───────── Error on other channel
```
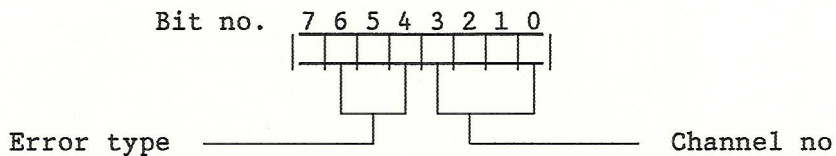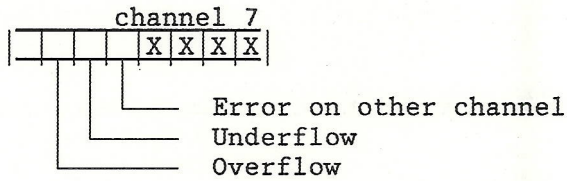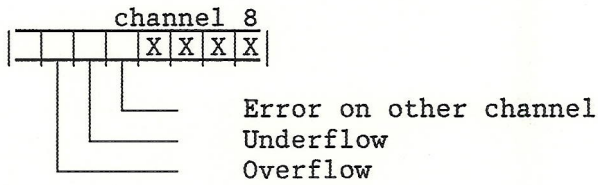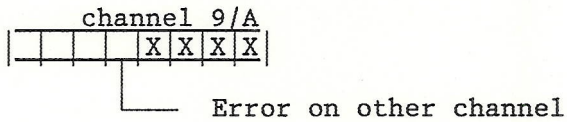
*   cannot be read via display unit PD 230.

If further information on the error is required, register F on the specific channel can be read.

Bit no.   <u>7 6 5</u> _ _ _ _

| 1 | X | X | _ _ _ _

If the most significant bit (bit 7) is "1" then one of the error codes 80-93 applies.

These error codes (read as hexadecimal numbers) have the following meaning:

<u>Error Codes</u>	<u>Meaning</u>
80	Voltage output overloaded
85	WDTimer run out
89	External data-storage fault.
90	Program cycle time fault (microprocessor short of time)
91	Write protected data storage fault.
92	Internal data storage fault.
93	Program storage fault.

7. Display Channel (Channel B).

The display-unit, PD 230, can be directly connected to all of the PROCES-DATA series of modules. Thus it is possible to display or change any register in a module. Moreover, it is possible to both display and modify up to 8 different registers, simply by pressing one of the 8 keys K1 to K8 on the display unit.

However, it is important to note that data, except for a code register is displayed in decimal format. Therefore, if it is required to display a flag register this will be displayed in decimal not in hexadecimal format.
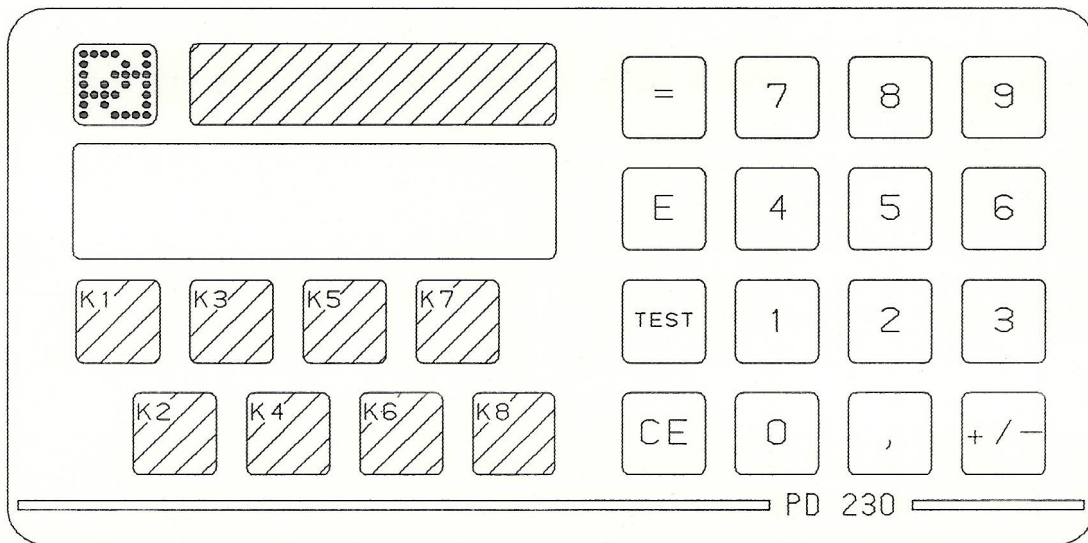
502024AA

Note: The shaded areas illustrate transparent screens, behind which customised text can be placed.

Reg.no.	Register name	Write medium	Storage bytes	Number of out	Read
0					
1	Code1	X	E	4	Hexadec.
2	Code2	X	E	4	Hexadec.
3	Code3	X	E	4	Hexadec.
4	Code4	X	E	4	Hexadec.
5	Code5	X	E	4	Hexadec.
6	Code6	X	E	4	Hexadec.
7	Code7	X	E	4	Hexadec.
8	Code8	X	E	4	Hexadec.
9					
A					
B					
C					
D					
E					
F					

To enable the programming of the PD 1651 module using the display unit, it is possible to modify the code register and other write protected registers (i.e. all registers with an E for storage medium in the register definition). A write-protected register is displayed by keying in E1, followed by the channel and the register number - eg:

Fullscale on channel 1 needs to be modified:

Key	Read out	
|Ē| |1̄|	0	
|0̄| |1̄| |1̄| |1̄|	0111	
|Ξ|	1B	Channel no., Reg. no.
|K1|	100,000	FullScale = 100
|2̄| |0̄| |0̄|	200	FullScale is altered from 100 to 200.
|Ξ|	200,000	FullScale is saved in write protected store

Note: The letters A-F are keyed in as numbers. The display unit will then convert them to letters (hexadecimal notation) for display e.g. 12 corresponds to C.

The display unit has the following characters:

0 :
1 :
2 :
3 :
4 :
5 :
6 :
7 :

8 :
9 :
A :
B :
C : *
D : *
E :
F :

502024BH

* Note these characters, as they are quite unusual.

7.1.   An example of programming a display channel via the PD 230 display unit.

Output in % on channel 7 is to read by pressing display channel 2 key (K2):

Key	Read out	
|E| |1|	0	
	1		1		0		2		1102	
|=|	B2	Channel no., Reg.no.
|K1|	000000	Display code 2

Thereafter the code can be written by altering one digit at a time.

This is done by pressing the display channel key (KX) below the digit to be altered.

|K3|	0	
|7|	7	0 is altered to 7 in digit 1.
|=|	7	2 = channel no. of value read out.
|K1|	700000	Display code 2
|K8|	0	
|3|	3	0 is altered to 3 in digit 6.
|=|	3	3 = 3 digits after decimal point.
|K1|	700003	Display code 2
|E| |1|	B2	
	0		0	
|=|	0	Back to read out mode.
|K2|	56,555	Output in % on channel 2.

Output in % on channel 7 is thus read with 3 digits after the decimal point.

7.2.   Programming the function keys (K1-K8).

To choose up to 8 registers selected by the keys below the screen (K1-K8), the module must be programmed. This is done by using the eight display-code registers in the display channel. There is a code register for each key below the screen (K1-K8). This means that the user can define a specific register for display associated with each channel (1-8).

Note: The code register cannot be displayed or changed by this method.

Reg. 1: Code 1

Dig.1	Dig.2	Dig.3	Dig.4	Dig.5	Dig.6

Channel no.,
Reg. no. for
out/input value
from K1

Number of digits after
decimal point (0-6)

Reg. 2: Code 2

Dig.1	Dig.2	Dig.3	Dig.4	Dig.5	Dig.6
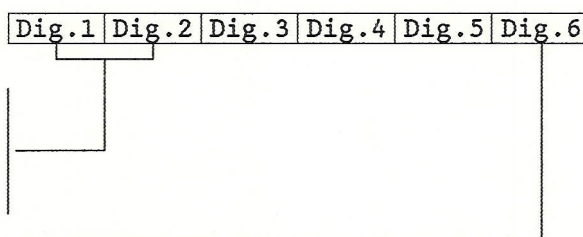
Channel no.,
Reg. no. for
out/input value
from K2

Number of digits after
decimal point (0-6)
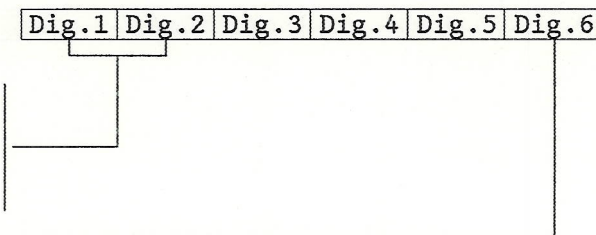
The address of the register required to be displayed or changed is stored in digit 1 and 2.

Digit 6 is used for selecting the number of digits that should follow the decimal point on the displayed value.

Note: If an undefined address is being selected the display unit will show 999999 on the display.

7.3.   Displaying error codes.

The PD 1611 is equipped with an extensive self testing system, which can detect faults caused by the module being wrongly used, or within in the module itself.

When the testing routines detect a fault, the user will be advised of this on the display unit, by means of an "A" for Alarm in the first digit of the display. By pressing the TEST key, the display will show an error code of 2 digits, indicating the nature of the fault (see section 6 "Reg. F: Error F").

The test system is designed so that the alarm is statisized even if the fault disappears. The error code on the display is updated only by pressing the TEST-Key. If several errors occur at once, only the error code with the highest value is retained.

Note: The display can only present error codes on channel 1-8.

7.4.   Connecting the PD 230.

The display unit is connected to the PD 1611 with a single 2 core cable. The display unit is powered through this cable, and exchange of data between PD 1611 and PD 230 also uses this medium. The length of the cable must not exceed 100 m and the cross sectioned area must be at least 0.75 sq.mm.

The cable is attached to the two spade connectors on the back of the display unit and to terminals 1 and 3 on the PD 1611's circuit board.

Terminal 1 should be connected to the spade connector marked "+".

7.5. Assembly drawing, PD 230.

The display unit PD 230 is designed for fixed positioning, eg, on a panel (see diagram).
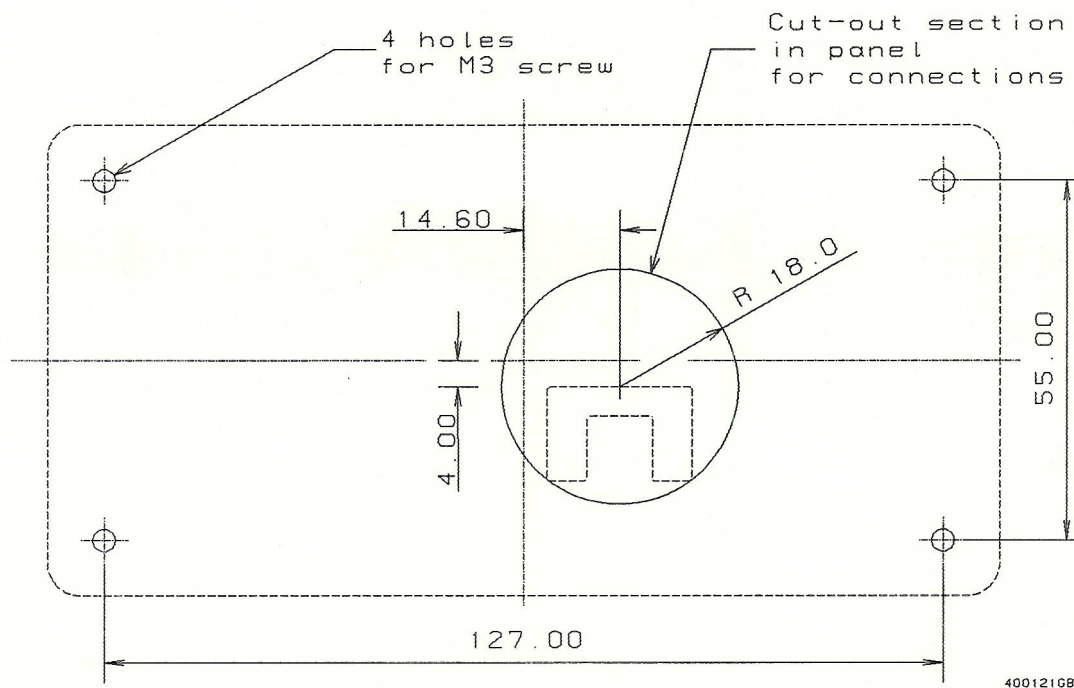
The dimensions of the display unit are:

H x L x W = 8 x 144 x 72 mm.

The unit is watertight and tolerates ambient temperatures between -10 °C and +50 °C.

8. Voltage Output.

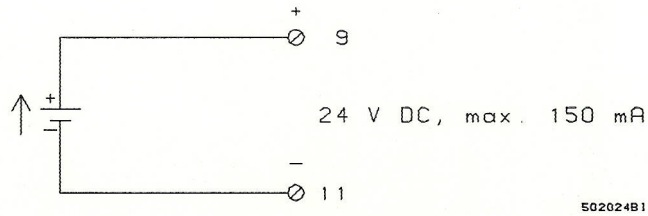The PD 1611 module is equipped with a voltage output which provides 24 V DC, at 150 mA maximum.

Fig. 8.a.: Current output.

The voltage output can be used to supply external DC-equipment.

The voltage output is protected against overloading by a zenerdiode and a current limiting resistor.

Should an overload occur it will result in an error being stored in the error code register.

Following an overload the load should be removed before the output can be used again.
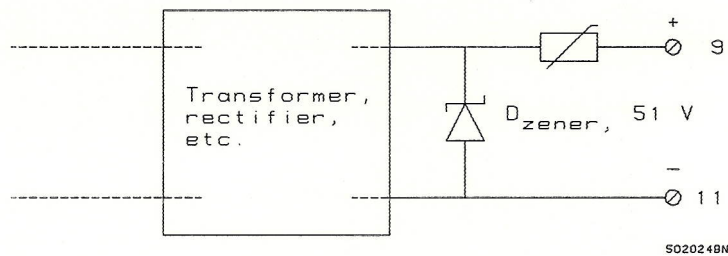
Fig. 8.b.: Voltage output.

9. Construction, Mechanical.

The PD 1611 module consists of two circuit boards housed in an aluminum case. The case measures W x H x D = 138.6 x 120 x 42 mm.
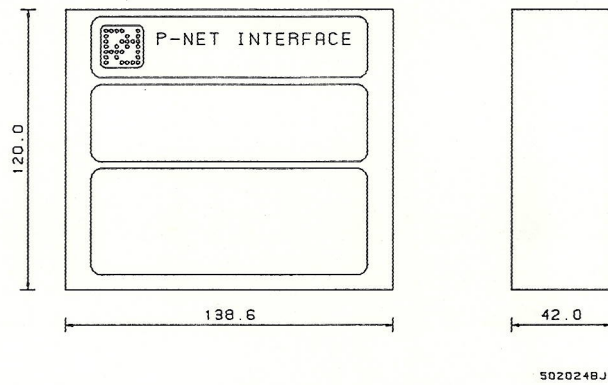

Fig. 9.a.: Aluminum Case for PD 1611.

The module is intended for mounting on a base circuit board, PD 1081 (Fig. 9.b.) where the interface terminals and the switch "Program Enable" are placed. PD 1080 base circuit board is the same for all the modules in PD's module series and measures W x H = 142 x 150 mm.
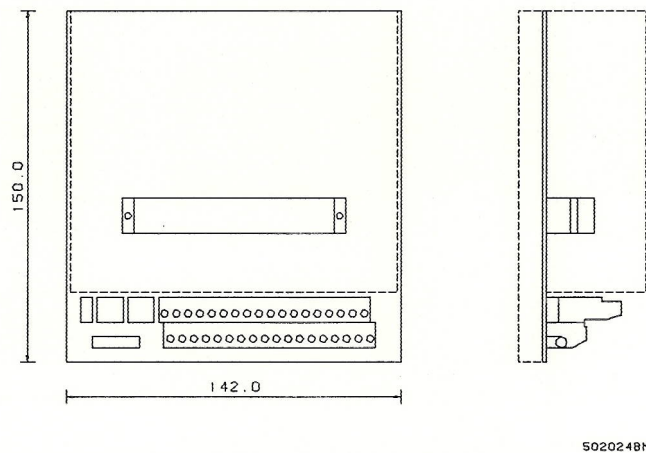

Fig. 9.b.: Base circuit board PD 1081.

## 9.1. Connections.

The connection between the module and the base circuit board is performed by plug and socket. The module can thus be changed without the wiring having to be disconnected.

Apart from the terminals and the program enable switch, the base circuit board contains 2 rotary switches, which are used for setting the address of the module for the P-Net. The number applies to the base circuit board, and therefore need not be set again if a module is changed.

The module, with base circuit board, is designed for installing in either a sealed box for wall mounting or for incorporation in a cabinet with other equipment.

502030AI

10.   Specifications.

	Min.	Typical	Max.	Units
Power supply	20,4	24	27,6	V AC/DC
Power consumption	2,9		7,8	W
Current input:				
Deviation			±0,3	%
Resolution			0,01	%
Repeatability			0,02	%
Impedance			100	Ω
Voltage input (0-500 mV):				
Deviation			±0,3	%
Resolution			0,01	%
Repeatability			0,02	%
Impedance	1			MΩ
Temperature input with Pt-100 detector (DIN 43760)				
Deviation			±0,9	°C
Resolution			0,05	°C
Repeatability			0,1	°C
Temperature range	-30		400	°C
Loss of effect in temp. detector			0,1	mW
Voltage output				
Output voltage at 0 mA	24	26	28	V DC
Output voltage at 100 mA	21,5	23,5	25	V DC
Output current			150	mA
Ambient temperature	0		50	°C

11. Noise limiter.

In environments with a high level of noise it is recommended to insert a hum eliminator before the supply transformer.
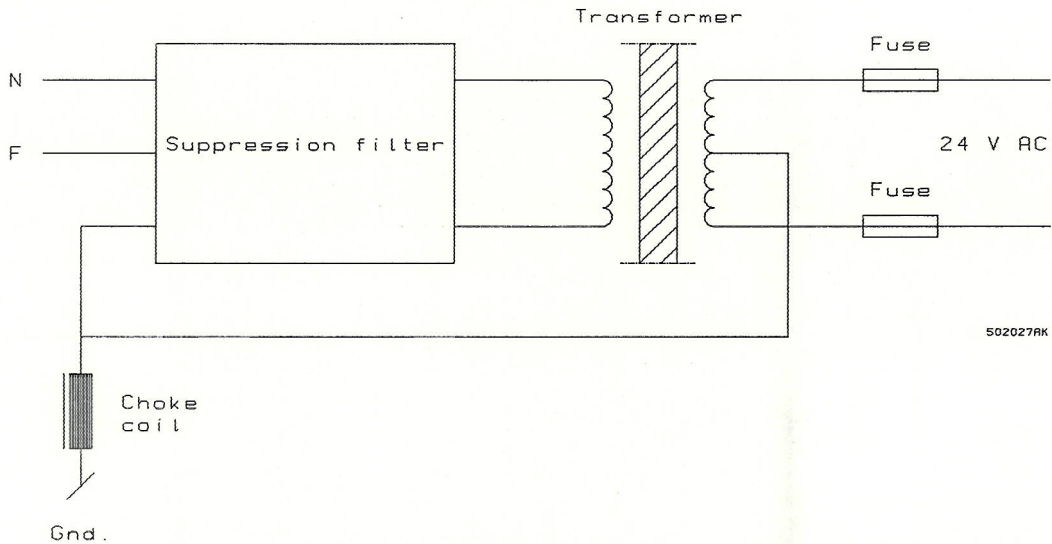
The hum eliminator could be a Schaffner FN332-1/05.

Also a hum eliminator with built-in impedance coil may be used.
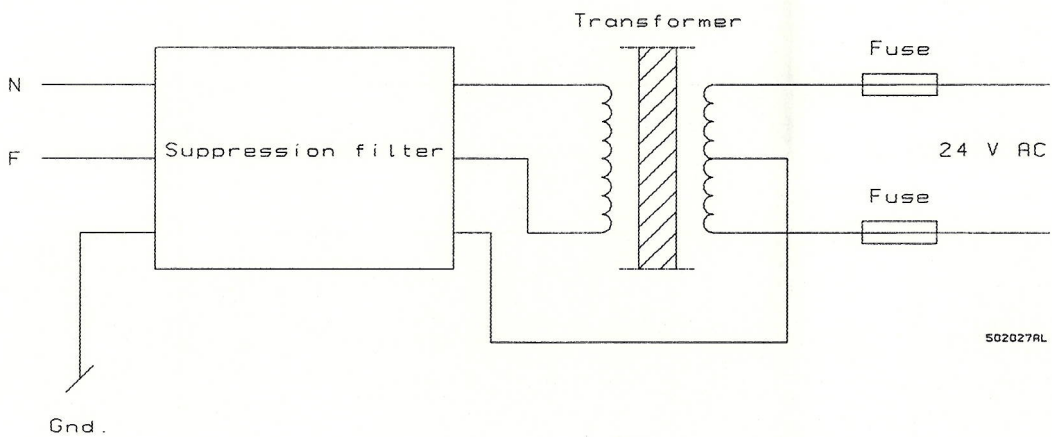
The hum eliminator could be a Schaffner FN343-1/05.

Because of the centre tap on the transformer, two fuses are necessary. If the + or - terminals are short circuited a large current will be produced, because the voltage is ±12 V relative to ground.

It is recommended to use a two-chamber transformer with the primary winding in one chamber and the secondary winding in the other chamber.

12. Survey of registers in the PD 1611 module.

Reg. no.	Service channel 0	Analog input 1-6	Current output 7	PID-regulator 8	Dosing channel 9-A	Display channel B
0		AnalogIn	AnalogOut	Output	FlagReg	
1	DeviceType			Signal1	Actual1	Code1
2	PrgVers			Signal2		Code2
3	Error3			Actual		Code3
4						Code4
5					ActOffset1	Code5
6					ActOffset2	Code6
7	WDTimer		SetPoint	SetPoint	Actual7	Code7
8	WDPreset			FlagReg8		Code8
9	Code9	Code9	Code9	Code9	Code9	
A				CodeA	CodeA	
B		FullScale	FullScale	Xp		
C		ZeroPoint	ZeroPoint	Ti		
D				Td		
E						
F	ErrorF	CHError	CHError	CHError	CHError	

13. Survey of standard identifiers in the PD 1611 module for use in Process-Pascal.

Name	Register no.	Number of bytes	Storage medium	Channel no
ActOffset1	5	4	BR	9-A
ACtOffset2	6	4	BR	9-A
Actual	3	4	R	8
Actual1	1	4	BR	9-A
Actual7	7	4	BR	9-A
AnalogIn	0	4	R	1-6
AnalogOut	0	4	R	7
CHError	F	1	R	1-A
Code1	1	4	E	B
Code2	2	4	E	B
Code3	3	4	E	B
Code4	4	4	E	B
Code5	5	4	E	B
Code6	6	4	E	B
Code7	7	4	E	B
Code8	8	4	E	B
Code9	9	4	E	0-A
CodeA	A	4	E	0-A
DeviceType	1	2	P	0
Error3	3	1	R	0
ErrorF	F	1	R	0
FlagReg	0	1	R	9-A
FlagReg8	8	1	R	8
FullScale	B	4	E	1-6
Output	0	4	R	8
PrgVers	2	2	P	0
SetPoint	7	4	BR	7-8
Signal1	1	4	BR	8
Signal2	2	4	BR	8
Td	D	4	E	8
Ti	C	4	E	8
WDPreset	8	2	E	0
WDTimer	7	2	R	0
Xp	B	4	E	8
ZeroPoint	C	4	E	1-6

13.1. Survey of standard identifiers (Bit).

Name	Register no.	Bit no.	Channel no.
BatchFlag1	0	5	9-A
BatchFlag2	0	6	9-A
EORFlag	0	0	9-A
Handshake	0	7	9-A
Manual	8	0	8

Contents, alphabetic