USER'S MANUAL
for temperature programmer 3123

- **Temperature programmer with 8 programs**
- **Control: P, PI, PID, ON / OFF**
- **Output functions: heating or cooling, alarm**
- **2 inputs**
- **3 outputs**
- **Communication**
- **Can send setpoint temperature to 1011 controllers**

3123

1. TECHNICAL DATA

Main characteristics		
	Power supply	90 ÷ 250 Vac; 40 ÷ 400 Hz; 4VA max
	Number of inputs	2
	Number of outputs	3
	Display	Two, 4+5 digits, 7-segment LED, 9mm, green; LED indication of currently active program segment
	Operating conditions	T: 0 ÷ 50 ^{0}C; RH: 5 ÷ 90%
	Storage	T: - 40 ÷ 85 ^{0}C; RH: 5 ÷ 90%
	Dimensions (W x H x D) (mm)	96 x 96 x 145
	Mounting hole (W x H) (mm)	91 x 91
	Weight	560g

Inputs		
Thermocouple	Type	J, K, L, R, S, B
	Cold junction compensation (CJC)	Internal or 0 ^{0}C, 25 ^{0}C, 40 ^{0}C, 50 ^{0}C (reference)
RTD	Type	Pt - 100, 3 - wire
	Cable resistance	max 10Ω per wire
Linear input	Type	Linear current or voltage
	Range	0 ÷ 20mA (for current input); 0 ÷ 1V or 0 ÷ 10V (for voltage input)
Input filter	Filtration factor	1 ÷ 128

Outputs		
Relay	Characteristics	3 - pin; 8A / 250 Vac, uninterrupted load 3A max
	Usage	Heating, cooling or alarm
Logical (optional)	Characteristics	max 20mA, 18 Vdc; non isolated
	Usage	Heating, cooling or alarm

Measurement (accuracy)		
	Measurement frequency	8Hz (125mS)
	Measurement resolution	2μV for range 10 ÷ 60mV; 0.8μA for range 0 ÷ 20mA; 50μV for range 0 ÷ 1V; 500μV for range 0 ÷ 10V
Measurement error	Linearization error	≤ 0.1%
	Thermocouple CJC error	< 1 ^{0}C for ambient change in range 0 ÷ 50 ^{0}C
	Total error	< 0.5% ± 1 digit

Control function		
Programmer	Number of programs	8 programs, each with 4 segment; Several programs can be connected in a sequence
Control	Type of control	ON / OFF, P, PI, PID
Alarm	Type	Independent (absolute) high and low limit; High and low deviation alarm
	Operation mode	"latched" and “non-latched", NO / NC alarm signaling

Communication		
Digital	Communication standard	EIA 485
	Communication protocol	EI - BISYNCH

Programmer 3123 is microprocessor operating device intended for sophisticated temperature or other process control according to given program. 3123 generates setpoint as an array of horizontal segments in time adjustable by duration and steepness. Device memory can hold 8 programs with 4 such segments. There is a possibility for multiple run of one program, sequent connection of several programs etc... As an option, special contacts for program flow control can be installed on the back side of device.

Programmer supports P, PI, PID and ON/OFF type of control. Selected type is same for both outputs if both inputs are active.

This device has 2 inputs for connection of various temperature sensors: thermocouple and RTD or standard current / voltage inputs. If needed, second input can be turned off. When active, both inputs must have same type of input sensors attached. Linearization of sensor characteristics, compensation of cable resistance for 3-wire RTDs and cold junction compensation for thermocouples provide high measurement accuracy.

Programmer 3123 has a modular structure which allows built in of 2 output modules - relay and logical - on each of 3 outputs. First two outputs are for control, each for its own control loop, while third one ha only alarm function operating for both control loops.

Two security levels are embedded: using access code and special access grant procedure for easy parameter access and maximal protection of accidental change.

On customer request, assembly for communication according to EIA 485 (RS-485) standard can be installed which enables connection of controller with PC or some other microcomputer system.

1.1. Ordering code

Ordering code for new device gives manufacturer precise data of desired characteristics and functions.

Ordering code has a following form:

TYPE - X - XX - XXX - XXXX - XXXXX

X - Inputs (probe type, same for both control loops)

XX - measurement range

XXX - Output type: output 1 / output 2 / output 3

XXXX - communication (optional)

XXXXX - rear program contact (RPC - optional)

Example:

3123 - K - 0 ÷ 1200 ^{0}C - relay / relay / relay

or

3123 - Pt-100 - 0 ÷ 200.0 ^{0}C - logical / logical / relay - EIA 485

or

3123 - J - 0 ÷ 400 ^{0}C - relay / relay/ relay - RPC

NOTE: 3123 is delivered with predefined inputs, but customer can alter it at any time.

2. INSTALLATION

Dimensions of the controller and mounting hole data are given in the characteristics. Device is fixed with Π type profile to front plate of the power electric switch board.
Upper row of the clamps on the back side of device are for probes (inputs) and other informational cables connection. Bottom row is for energy cables for power supply and outputs connection.

2.1. Power supply connection

Device is powered by 90 ÷ 240 Vac, 50 Hz, via clamps 23 and 24. Clamps 22 and 23 are internally short-circuited.
Device will power up immediately after connected to power supply.

2.2. Outputs connection

3123 programmer has 3 outputs which can be relay or logical.
For relay type, output can be NO or NC. NC should be used only for signaling. maximal uninterrupted load is 3A. Fuse is necessary.
In case of logical output, signal is set to high level when output is active. Output is useful for SSR (solid state relay) drive. Logical outputs are built in instead of relays. They are not isolated from the probe inputs.
Outputs 1 and 2 are used for process control, and output 3 only as alarm.
Connection of outputs modules is shown on picture 2.1.

2.3. Inputs connection

3123 support various inputs: thermocouples (certain types - see technical data), 3 wire Pt-100 RTD, and standard current signals 0-20mA or voltage signals: 0-1V or 0-10V. When both inputs are used, sensors must be of same kind. Connection is shown on the picture 2.1.
When thermocouples are connected at great distance, appropriate compensation cable must be used.

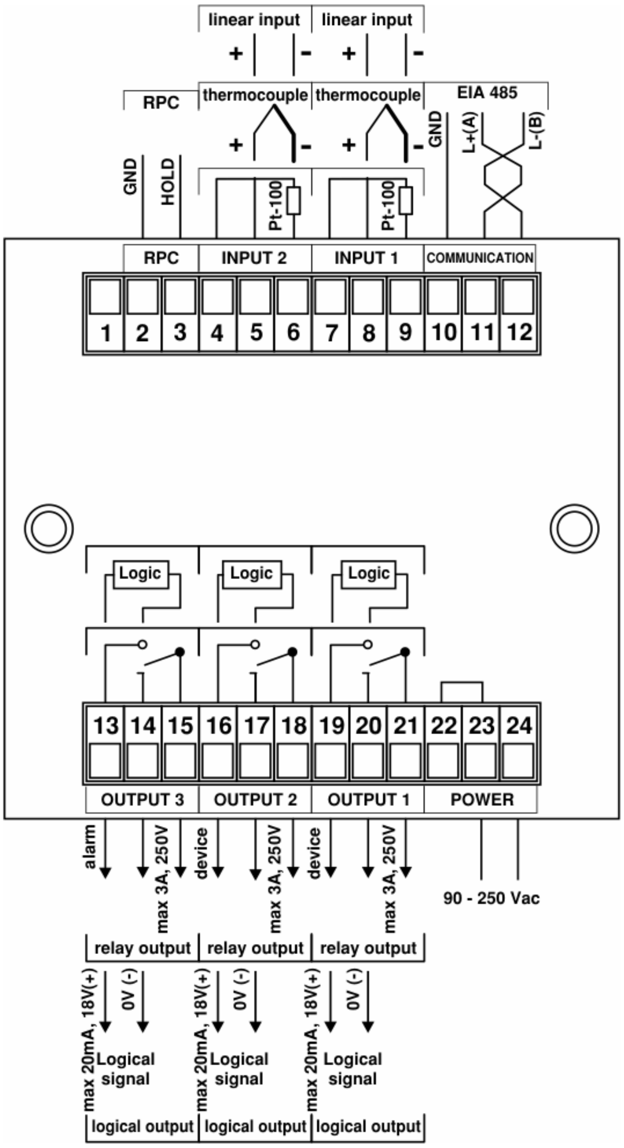
To set up device for specific type of probe it is required to set certain parameters to appropriate value and adjust dip switches SW1 and SW2 accordingly (see chapter 6.3.1).
Device is delivered already set up for specific probe, but customer can change this in any time if necessary.

2.4. Rear program connector (RPC) connection

Upper row of connectors on the rear side of the programmer has 2 contacts: 2 (GND) and 3 (HOLD) which are used for remote program flow control. This is optional feature built in only on user request. Depending on the state between these 2 contacts, programmer can set RUN or HOLD state. Non-voltage contacts of switches, relays, optocaplers etc can be connected.
More details are available in chapter 4.4.2 of this manual.

2.5. Communication connection

Programmer 3123 can communicate with other microprocessor systems. This is optional feature available at customer's request upon ordering).
2 wire, twisted pair, shielded cable should be used. Maximal length allowed for this cable is 1200m. Impendency for this type of cables is 120Ω. Pull down resistors equal to cable impendency must be placed on cable ends to reduce influence of signal reflection at the cable end. Cable shield should be connected to ground of PC computer or some other device.



Picture 2.1: **Rear side connection layout**

3. OPERATION

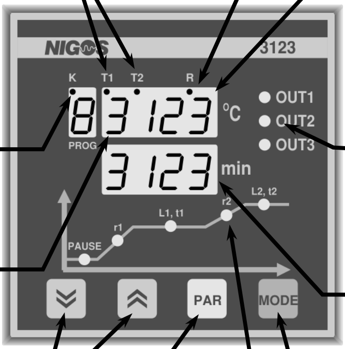
LED dots **T1** and **T2** indicate control loop for which the temperature is displayed:
- T1 is lit - measured temperature on probe 1
- T2 is lit - measured temperature on probe 2
- both dots turned off - program setpoint

LED dot **K** - when lit indicates that communication with PC computer

DISPLAY PROG shows the number of currently selected program

Buttons **DOWN** and **UP** are used for decrease / increase of the value of selected parameter

Button **PAR**
- parameter selection
- alarm deactivation if that option is active



LED dot **R** indicates programmer's state:
- blinking - normal operation according to program
- lit - temporary stopped program
- turned off - programmer is inactive

UPPER DISPLAY shows:
- measured value
- parameter symbols
- symbol Snbr - when error in input signal is found (probe malfunction)

LED diodes **OUT1, OUT2, OUT3** signals activity of outputs 1, 2 and 3

LOWER DISPLAY shows:
- remaining time till active segment ends
- selected parameter value
- programmer state
- alarm and system error messages

Button **MODE**
change the type of display on upper LED: measured temperature on probe 1, probe 2 or program temperature

SEGMENT LEDs mark current phase of the program, i.e. currently active segment during setup

3.1. Standard display mode

Upon power connection, installed software version is displayed. Upper display displays message UEr and lower one displays software version. This message stays displayed for a few seconds and has only informational purpose (can be used during consultation with manufacturer in regards to some problem).

If the system is connected properly, there is no problem found no button is pressed and programmer is inactive (not running a program), display PROG shows ordinal of currently selected program, upper display shows some temperature value and corresponding LED dot is lit: T1 - for probe 1, T2 - for probe 2, none - program setpoint (**note:** use button MODE to change source) and lower display shows message idLE indicating that programmer is inactive. Other indication might only include lit diode OUT 3 which indicates alarm situation (only if occurred). Programmer returns automatically to this "standard inactive display mode" few seconds after any button is pressed.

When in active mode, display is slightly different. Lower display shows remaining time (in minutes) for completion of currently running program segment. Also, segment LEDs will indicate program progress. LED of currently active segment will flash, while previous (indicating already finished segments) LEDs will lit constantly.

3.2. Device parameters

3.2.1. Selecting and adjusting parameter values

Parameter selection is made by pressing the button PAR. Parameter symbols are displayed on upper display, and its value on lower. Use buttons ⏬ and ⏫ to change the value shown on lower display. After one parameter is set, press the button PAR to go to another parameter. If display is reseted due to longer period of inactivity, press button PAR again to select desired parameter and continue adjustments. All parameter adjustments are saved in device memory automatically few seconds after last button is pressed and they remain saved even after disconnection of the device from power line.

All activity over communication line are similar to ones performed by pressing the buttons, so it will not be discussed in details.

3.2.2. Code protected parameters (CodE)

Note: Before proceeding with any parameter change, please read this manual carefully!!

In order to protect parameters from accidental change or un-authorized access, certain system parameters are in a list protected with user code. In order to gain access to these parameters, following procedure is required:

Press the button PAR and select the parameter CodE. Symbol is shown on upper display, initial value zero (0) on the lower.

- Use buttons ⏬ and ⏫ to adjust the value to 3 123. This is default factory-set access code.
- Press the button PAR.

After correct code entering, access to code protected parameters is possible without the need to enter code again for as long as the programmer is connected to power line. New code will be required after next powering up.
Code value 3 123 is default value set in the factory during production. It can be changed following the procedure described in chapter 5.3.
List of code protected parameters contain parameters which describe process characteristics and directly influence quality of control. Also, there are parameters which define alarm types and alarm limits.
All parameters are set to their default values, but in certain cases, their adjustments is necessary according to specific process control requirements. On customer request, even default values can be set during production to desired values.
Values of most parameters can be changed freely. However, there are certain parameters in the list which can only be viewed, and not changed because these are essential for system operation, and are therefore especially protected using access level protection - chapter 5.4. Their presence in the list is required in order to gain information about the system setting.

3.3. Error report

3123 programmer has an ability to recognize certain errors which may occur in device itself or on certain parts of control system, and displays appropriate message.

Occurrence of symbol Snbr on upper display and symbols Snb. 1 and/or Snb.2 on lower display means that the device found that signal received from appropriate probe has irregular value.
Main causes of this state can be:

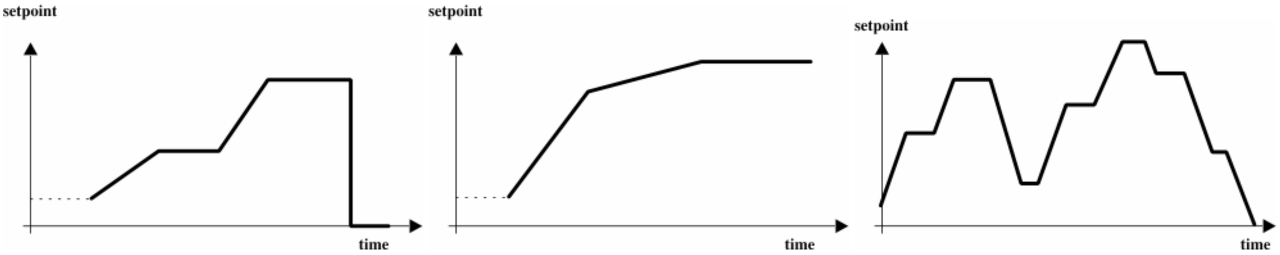
- connection break between controller and the probe (or transmitter)
- improper input connection
- discrepancy between probe type defined by parameter Sond and actual probe attached
- probe / transmitter malfunction
- programmer hardware/software error

Simultaneously with error reporting, programmer will enter into appropriate operation mode with malfunction probe which will be describe in chapter 6.3.8.

In case that bottom display shows symbols CS.Er, in.Er or E2.Er it is a warning that there is an error in programmer function. In this case, turn off the device and contact manufacturer.

4. OPERATING PRINCIPLES

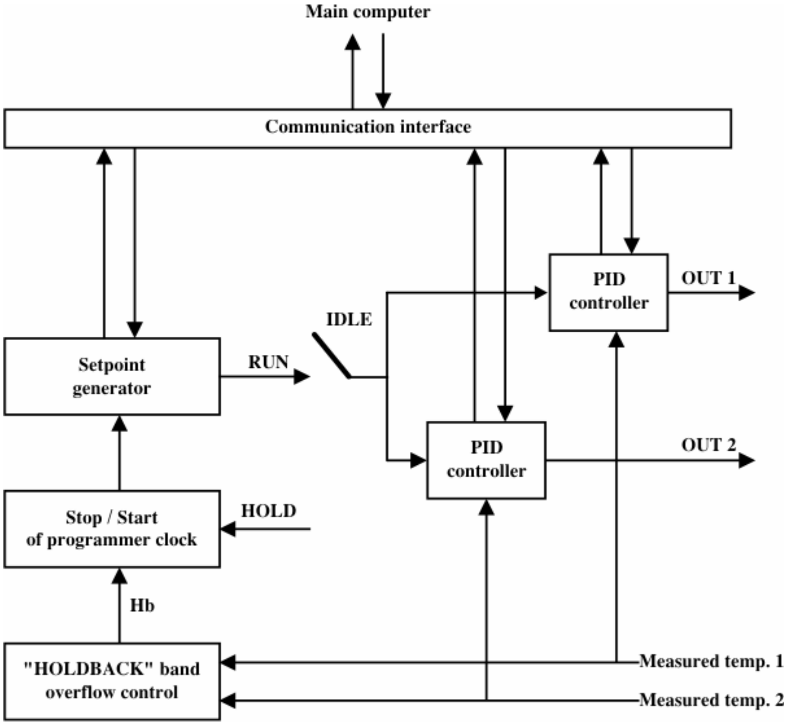
Temperature programmer 3123 enables control of complex technological processes which require several changes of temperature set points during process following the strict criteria for each phase of the process. Examples of these processes are shown on picture 4.1.
Since this device support temperature measurement and control in 2 control loops, setpoint generated by the programmer is same for both loops, while actual control is separate using separate outputs. Further text will only consider one control loop with remark that same apply to selection of 2 control loops. In order to enable second loop, parameter Sn2 must enable usage of second loop too. More details are given in chapter 6.3.3.



Picture 4.1. **Program examples**

4.1. Programmer structure

Part of temperature programmer 3123 is PID controller which together make a unique system where programmer generates setpoint according to prepared program and PID controller makes sure that temperature follows it as best as possible. Device also provide automatic deviation control and holdback control. Picture 4.2 shows programmer's block diagram.



Picture 4.2. **Conceptual block diagram of a programmer**

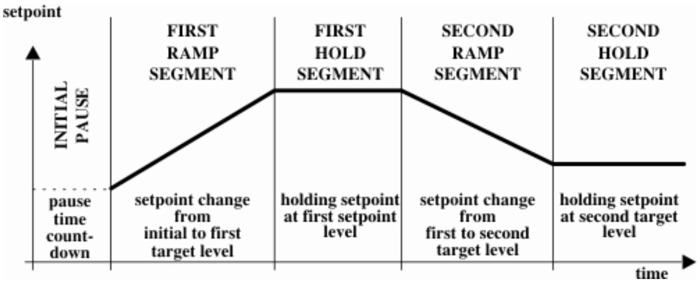
4.2. Program segments

Programmer can store up to 8 different programs, where each program has 4 consecutive segments. Segments always have following format: first ramp segment, first hold segment, second ramp segment, second hold segment.

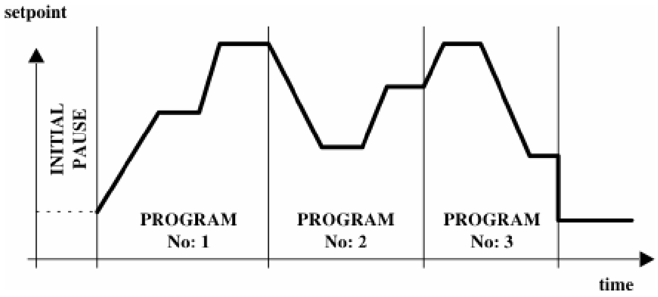
For each ramp segment, programmer increase or decrease setpoint at a linear ramping rate till target level is reached. Ramping rate is expressed in units/minutes (most common: °C/min). During segment execution, programmer calculates remaining time to completion and displays it on bottom display (true only if there is no unpredicted hold-ups). When certain holdback occurs, setpoint change according to ramp rate as well as time countdown are automatically stopped until conditions for normal progress of current program segment are met. Possible causes for hold-ups are given in chapters 4.3.3., 4.3.4 and 4.4.2.

Hold segment maintain setpoint unchanged for defined time set by user. Setpoint (target level) achieved in last ramp segment becomes setpoint for next hold segment. It is maintained same until defined time, in minutes without hold back time included, expire.

Program start delay - this options induce initial pause as a special program segment which is executed only once before program start. Immediately after program start, this segment is activated and user can adjust time of pause in minutes. If nothing is entered, program starts normally few seconds after.


setpoint
INITIAL PAUSE
FIRST RAMP SEGMENT
FIRST HOLD SEGMENT
SECOND RAMP SEGMENT
SECOND HOLD SEGMENT
pause time count-down
setpoint change from initial to first target level
holding setpoint at first setpoint level
setpoint change from first to second target level
holding setpoint at second target level
time

Picture 4.3: **Program segments**



Picture 4.4: **Example of 3 programs connected in sequence**

There are several way how programs can run. Program may run as complete sequence of defined segments, certain segments can be turned off, or several programs can connect in sequence thus resulting in most various combination.
Picture 4.3 shows segments in single program, while picture 4.4 shows example of multiple program connection in one sequence.

4.3. Programmer states

Programmer can enter 3 states during operation either automatically, or on user request if required. Possible states are:

- **IDLE** - programmer inactive, no program active
- **RUN** - programmer active, program running
- **HOLD** - program running, but programmer is temporarily stopped, setpoint is maintained at the value which was defined at the moment programmer entered this state

Additional state is starting PAUSE state - when programmer pauses certain time before program execution starts

Programmer state is defined by the value of the parameter StAt. Values for this parameter can be manually adjusted using buttons, or automatically assigned by programmer logic. Manual setting of this parameter enables user manual control of the programmer operation.

Current programmer state is indicated by activity of LED dot R located on upper display:

- **R** dot turned off - IDLE state or initial pause state
- **R** dot blinking - RUN state
- **R** dot constantly lit - HOLD state

Additional indication of programmer's state is performed by front panel LED behavior. Bottom left LED labeled with PAUSE indicates initial pause is active. Blinking of some other LED indicates that appropriate segment is active. Past segments (already finished) are noticeable because their LEDs are constantly lit. One by one, they lit up indicating program progress. If several programs are linked, upon completion of one program and starting the new one, LEDs are reseted and this cycle repeats.

When program is stopped, all segment LEDs are lit constantly. It is (apart LED dot R) another indication of temporary holdback in progress.

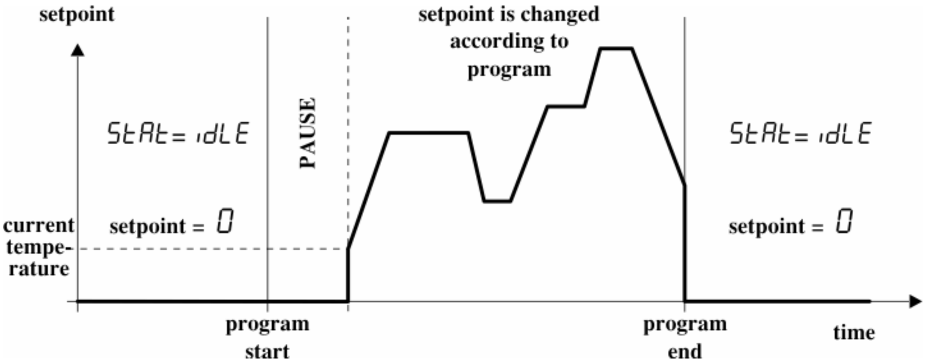
While in RUN or HOLD state, PID controller as a part of unique system still activates / deactivates appropriate outputs to ensure that measurement value (temperature) matches actual setpoint. In IDLE state, PID controller will turn off outputs (first and second) and thus stop process control.

4.3.1. IDLE state

IDLE state defines inactive programmer, parameter StAt has a value idLE and this symbol is constantly shown on bottom display. LED dot R and all segment LEDs are also turned off. First 2 outputs are turned off during this state, while third one used for alarm signaling can be active depending on alarm parameter setting and current system situation.

Programmer enters IDLE state automatically after completion of selected program. It can also be manually set.

It is possible to go from IDLE state to any other listed state by adjusting the value of parameter StAt using front panel buttons. Also, it is possible to force programmer into this state by selecting value idLE for parameter StAt. This is also basic procedure to stop currently active program.



Picture 4.5: **IDLE state at the beginning and at the end of the program**

4.3.2. RUN state

RUN state defines active programmer, i.e. program is running (in progress). Programmer executes all segments from active program, while PID controller ensures that measured temperature adequately follows setpoint changed according to program. LED dot R blinks to indicate RUN state, and segment diodes also indicate current state.

During program run and standard display mode, lower display will show time in minutes remained until completion of current program. If both probes are set to active state, program start will influence both control loops under same conditions (chapter 6.3.3).

When RUN state is activated, value of parameter StAt is set to run. As with other states, it can be automatically or manually set to any other state by setting the value of parameter StAt to appropriate value.

4.3.3. HOLD state

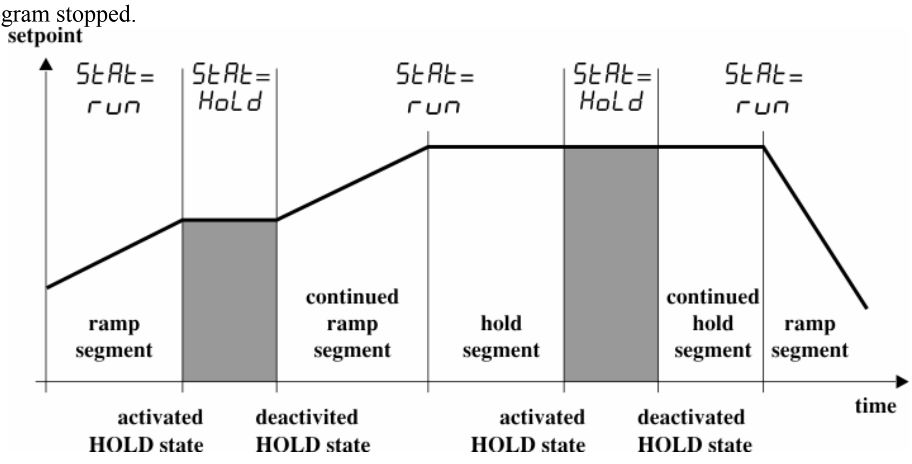
HOLD state defines a situation where program progress is stopped at some segment, while PID controller still work trying to maintain temperature at level achieved in the moment program stopped.

HOLD state is user activated by setting the parameter StAt to value HOLd or using RPC contacts if they are built in (optional). Detailed description of program control is available in chapter 4.4. During HOLD state, LED dot R on upper display and appropriate segment LEDs are constantly lit.

This state will induce time extension for as long as this state lasted.

Again, HOLD state is deactivated by setting different value for parameter StAt or using rear panel connectors (RPC). After HOLD state deactivation, programmer will continue operation from the point where it was stopped.

Picture 4.6 shows an example of the program, which enters HOLD state during operation.



Picture 4.6: **Programmer HOLD state**

4.3.4. HOLDBACK state

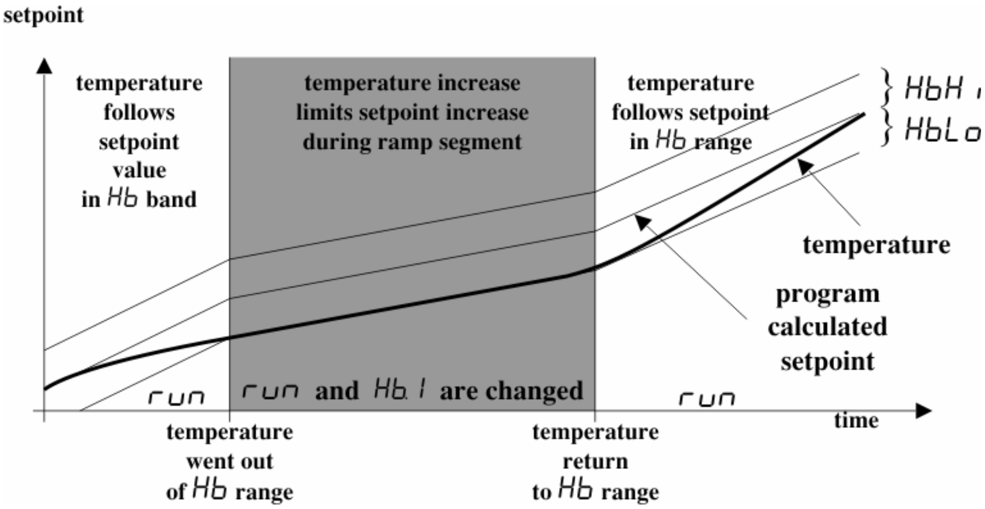
HOLDBACK state is special type of HOLD state. Programmer behave same as in HOLD state, with only difference that user cannot set this state, but only programmer itself.

If during program run *difference* between currently measured temperature and programmer setpoint becomes higher than value of parameters HbHi **- holdback high band, i.e.** HbLo **- holdback low band** (see chapter 4.6), programmer automatically sets HOLDBACK state. This procedure gives actual temperature a chance to "catch up" with programmer setpoint and controls overshoot or undershoot.

Similar to HOLD state, setpoint that was calculated at the moment programmer entered HOLDBACK state remains unchanged, and PID controller tries to return measured temperature into allowed band. Clock is stopped, which enables technological process to finish. This means that actual segment duration will be extended for a period of HOLDBACK duration. When actual temperature returns within allowed band, programmer will automatically deactivate HOLDBACK state, and activate RUN state, which enables normal program completion.

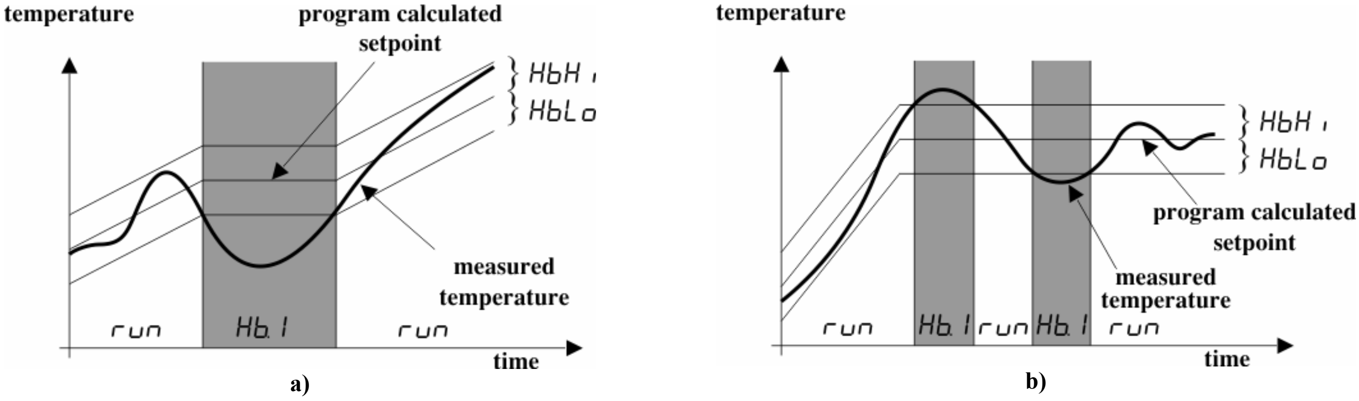
During HOLDBACK state, parameter StAt receives value Hb, and LED dot R on upper display is constantly lit. Appropriate segment diode on front panel is also lit. Lower display will flash symbols Hb.1, Hb.2 or both depending in which control loop holdback condition occurred.

Effect HOLDBACK state has on a program is shown in picture 4.7 and 4.8. Picture 4.7 shows HOLDBACK state, which occurs when system response limits setpoint increase. Programmer switches between RUN and HOLDBACK state until temperature "catch up" with setpoint.


setpoint
temperature follows setpoint value in Hb band
temperature increase limits setpoint increase during ramp segment
temperature follows setpoint in Hb range
HbHi
HbLo
temperature
program calculated setpoint
run
run and Hb.1 are changed
run
time
temperature went out of Hb range
temperature return to Hb range

Picture 4.7: **Example of HOLDBACK state**

Picture 4.8 represent HOLDBACK state occurrence during ramp segment (a) and due to system oscillation during hold segment (b).



Picture 4.8: **Examples of HOLDBACK state**

4.4. Programmer control

Programmer control, i.e. change of state by user can be done in 3 different ways:

- using front panel buttons
- using contacts 2 and 3 on the rear side of device (rear panel connectors - RPC)
- using PC command (if communication option is installed)

All above ways of control are equal and programmer will execute last given command. Only thing impossible is to set programmer into RUN state while contacts 2 and 3 on rear panel of device are open, no matter if buttons, or PC communication is used. When these contacts are open, and RUN state initiated, controller will enter HOLD state instead.
Device is delivered with factory connected contacts 2 and 3, and their usage is optional (must be requested during ordering).

Programmer can be controlled via PC using installed communication line and appropriate software. This option will not be covered in this manual.

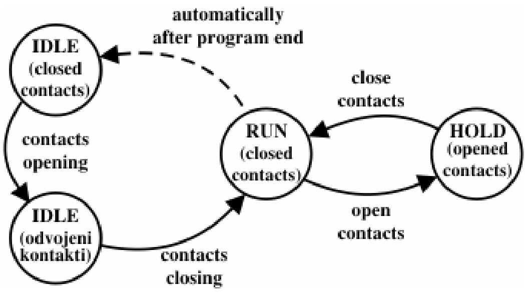
4.4.1. Program control using front panel buttons

User can obtain full control over the programmer using the buttons on the front panel of device. Programmer parameters are accessible in any time and can be accessed using button PAR. Their symbols are shown on upper display and their values adjustable with buttons ⏬ and ⏫ are shown on lower display. This control most commonly imply setting or changing of the value of parameter StAt. Possible values are: idLE, run, HOLd and Hb where the last one can not be set by user - only by programmer. More details are available in chapter 4.3.

4.4.2. Program control using rear panel contacts (RPC)

Contacts 2 and 3 in upper row on the back side of device are used for remote control of program flow (programmer operation). This is optional feature and is built in only on customer's request. Closing and opening of these contacts control the behavior of the programmer 3123 and consequently influence program flow.

For regular use, these contacts can be connected to voltage-free contacts of suitable switch, relay, opto-capler, etc which can be installed at a certain distance from temperature programmer itself. This enables certain comfort of use and simplify programmer control especially for control of repeated process.



Picture 4.9: **Programmer state diagram when rear panel contacts are used**

When contact 2 and 3 are closed, programmer can be set to any state using front panel buttons or communication commands, while open contacts set the programmer into HOLD state whenever RUN state is activated.
If contacts are opened while programmer is in RUN state, it will advance into HOLD state. Programmer can return to RUN state only when contacts are closed, otherwise only IDLE state is possible through buttons or communication instruction.
If contacts are opened while programmer is in IDLE state, it can be set only to HOLD state using buttons or communication instructions. Closing of already opened contacts while programmer is in IDLE state will evoke transition to RUN state and start of selected program. This option is especially interesting because it allows repeated start of selected program with short opening and closing of the contacts after programmer finished previous program cycle and entered into IDLE state.

4.5. Program parameters

At the start of parameter list - as always accessible - are parameters which describe program. Among them is a parameter ProG which represent program number and a group of segment parameters for selected program.

Parameters are accessed in standard procedure by pressing the button PAR, and then their symbols are displayed on the upper display and values on lower. Values can be adjusted using buttons ⏬ and ⏫.

4.5.1. Program number

Parameter ProG - program number - indicates ordinal number of the selected program which will be starting program if we select RUN state, i.e. number of currently active program if programmer is already in RUN state. Value of this parameter is displayed on display PROG. Memory can store up to 8 different programs, so this parameter can have values 1 to 8.

All following program parameters which list after parameter ProG (using subsequent presses on button PAR), relate to the program whose number is displayed on PROG display. They are so called segment parameters.

4.5.2. Segment parameters

Segment parameters are also in the group of program parameters and are used to describe setpoint change for selected program. They follow parameter Pro9 in the list and relate to program defined by this parameter. Segment parameters are:

r 1 - first ramp segment is the rate at which the setpoint ramps, from initial state to first target level **(L 1)**. Rate is adjustable in range 0.01 to 99.99 ^{0}C in minute. Initial value is last measured temperature before segment started i.e. last valid temperature setpoint from previous program in case programs are in sequence. Ramp angle depends whether target level is higher or lower than initial value.

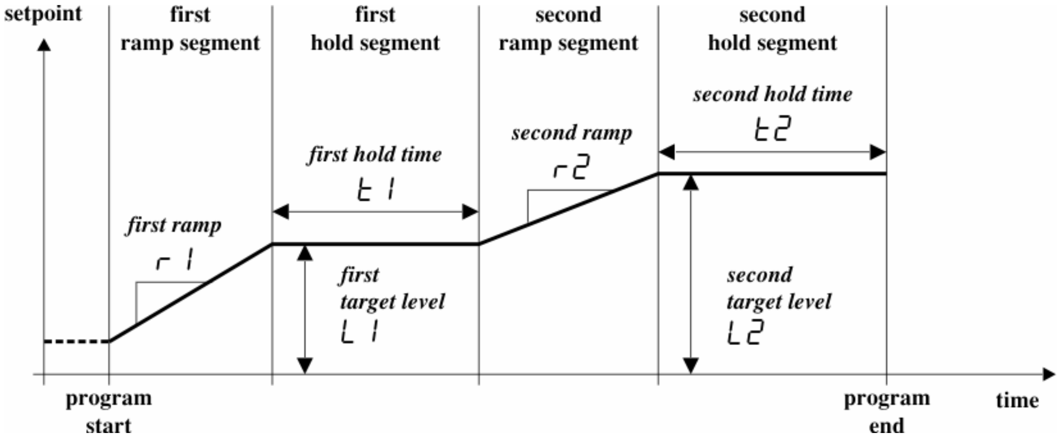
L 1 - first target level is temperature value to which the setpoint ramps when the programmer is set into RUN state. This becomes setpoint during first hold segment. Value of this parameter is adjustable over entire range of input sensor.

t 1 - the period of time the setpoint remains at L 1, after completing first ramp (r 1), and before second ramp (r2) begins. Parameter is adjustable in range 0 to 9999 minutes. If the value is set to 0, this segment is skipped.

r2 - second ramp defines the rate at which the setpoint will ramp from L 1 to L2. Rate logic and parameter range is same as for r 1.

L2 - second target level to which the setpoint ramps after completion of second ramp segment. **Characteristics are same as for L 1** .

t2 - the period of time the setpoint remains at L2 , after completing second ramp (r2), and before program ends, or next program starts if there is consecutive program defined. **Characteristics are same as for t 1** .



Picture 4.10: **Segment parameters within one program**

4.5.3. Programmer behavior at the end of the program

Parameter End defines programmer behavior at the end of the program. If the value of parameter End is set to StoP, upon completion of last segment, programmer will end the process, turn off all outputs and set itself into IDLE state.
If the value of parameter End of specific program is set to some of symbolic values: JmP. 1 , JmP.2 , ... to JmP.8, programmer will, after current program completion continue run of next program, the one at which points the number in above symbolic value.
Transition from one program to next one is without pause, by concatenating first ramp segment of the next program to just finished program.
This option allows forming of complex programs with several segments, which can be very useful for complex process control. Also, it is possible to make "loops" with unlimited number of repetition of certain programs, or combination of previous two option which considerably expands range of possible applications of the programmer 3123.

4.6. Holdback band

Parameters HbHi and HbLo are used to define values of high and low holdback band limit. If these values are exceeded in any control loop (either on probe 1 or probe 2), programmer will enter HOLDBACK state and temporary hold program until difference between setpoint and measured temperature return within allowed band.

Holdback band parameters are in the code protected list (chapters 3.2.2 and 5.1), so it is required to enter access code to gain parameter access. Their value is adjustable in range 1 to 9999 ^{0}C in integer display, i.e. 0.1 to 999.9 ^{0}C in decimal point display mode. Each of these two parameters can be turned off by setting its value to OFF. In this case, parameter becomes inactive and there is no holdback in the part of band specific parameter relates to.

4.7. Program setup

Before program setup procedure is started, it is required to establish all phases program will go through and required parameters for each phase, sequence of programs, required conditions, etc... Based on these data, establish the layout of program which can fulfill all these requests.

New program must be entered while programmer is in the IDLE state. Otherwise, programmer will block the change of parameters of currently running program. So, when making the program first check if programmer is in IDLE state, and if not change the value of parameter StAt to idLE and wait few seconds for programmer to return to standard display mode (this means parameter change is memorized). Program setup procedure is as follows:

- Press the button PAR to select parameter Prog. Lower display shows ordinal number of the program. If necessary, use buttons ⏬ and ⏫ to adjust program number.
- Press button PAR to select parameter r 1 - first ramp. Press buttons ⏬ and ⏫ to adjust rate of temperature change. Value in ^{0}C per minute is displayed on lower display.
- Press button PAR again to select parameter L 1 - first target level. Adjust first target temperature.
- Press button PAR again to select parameter t 1 - first hold time. Press buttons ⏬ and ⏫ to adjust it.
- Press button PAR again to select parameter r2 - second ramp. Press buttons ⏬ and ⏫ to adjust it.
- Press button PAR again to select parameter L2 - second target level. Press buttons ⏬ and ⏫ to adjust it.
- Press button PAR again to select parameter t2 - second hold time. Press buttons ⏬ and ⏫ to adjust it.
- Press button PAR again to select parameter End, which determine programmer behavior at the eng of last segment in program. Use buttons ⏬ and ⏫ to select one of following symbols on lower display:
  - StoP - after last segment is finished, programmer will stop all activity and switch to IDLE state,
  - JmP. 1, JmP.2 , ... or JmP.8 - after program is ended, programmer will continue with next program which is selected here.

After last entry is made using procedure above, it is required to wait few seconds for programmer to return to standard display. This is setup procedure for one program. Repeat this procedure for any other program (up to 8 can be stored in memory).

This ends new programs setup procedure. If during data entering programmer switch to standard display mode due to long period of inactivity, and setup procedure is not finished, simply press button PAR until desired parameter shows, and continue program setup.

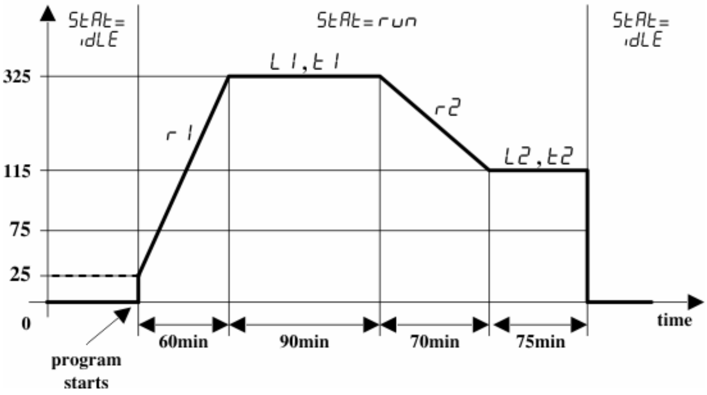
Same approach can be used to alter existent programs - only choose appropriate parameter and change its value.

After program parameters are stored, it is required to setup values for holdback band parameters HbHi and HbLo (high and low holdback band). These parameters are code protected, so use procedure from chapter 3.2.2. to unlock the programmer, and then proceed parameter adjustment using standard procedure.
Note that value of these two parameters is same for all programs, so their value should be checked each time different program is activated.

EXAMPLE of standalone program:

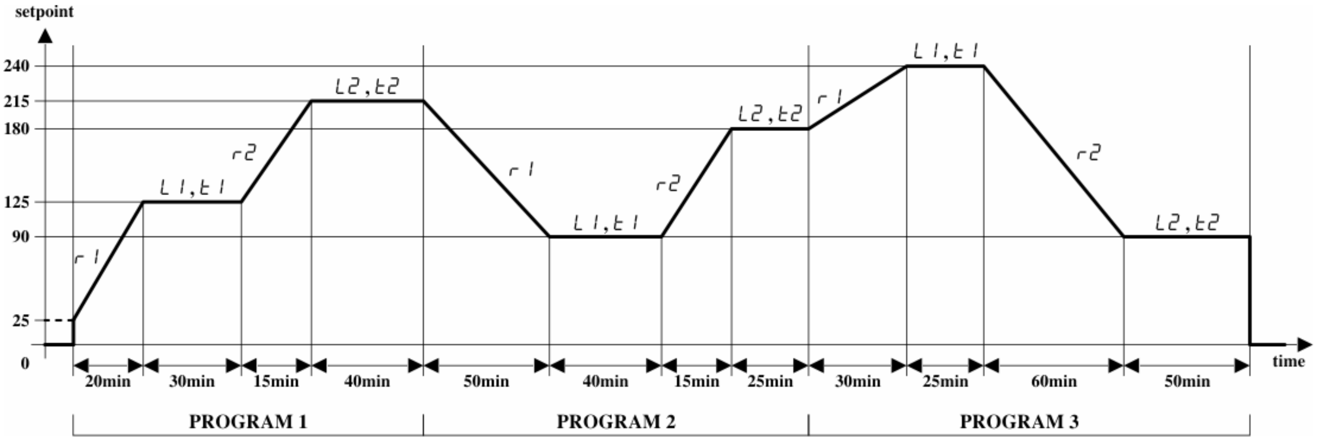
Measured temperature in the moment program is started is 25 °C. Other parameters are as follows:

r 1	=	5.00
L 1	=	325
t 1	=	90
r2	=	3.00
L2	=	115
t2	=	75
End	=	StoP



EXAMPLE of 3 concatenated programs execution:
Measured temperature in the moment program is started is 25 °C. Other parameters are as follows:

PROGRAM 1 parameters:	**PROGRAM 2 parameters**:	**PROGRAM 3 parameters**:
Pro9 = 1	Pro9 = 2	Pro9 = 3
r1 = 5.00	r1 = 2.50	r1 = 2.00
L1 = 125	L1 = 90	L1 = 240
t1 = 30	t1 = 40	t1 = 25
r2 = 6.00	r2 = 6.00	r2 = 2.50
L2 = 215	L2 = 180	L2 = 90
t2 = 40	t2 = 25	t2 = 50
End = JmP.2	End = JmP.3	End = StoP



4.8. Program start and operation

4.8.1. Program start

Check functionality of entire system before starting the program.
There are 3 ways to start the program: using front panel buttons, using rear panel connectors (RPC) or using instructions carried via communication line.

To start the program using front panel buttons use following procedure:
- Check current programmer state - programmer must be in IDLE state (lower display shows symbol idLE).
- Check the number of currently selected program - number is shown on PROG display. If adjustment is needed, set the parameter Pro9 to desired number and wait few seconds for programmer to return to standard display mode (parameter change is saved).
- While programmer is in standard display mode, press button ︽. Lower display will show symbol run - indicating that program is activated and device is preparing itself for RUN state.
- Wait few seconds for programmer to start operation. During preparation phase (symbol run is still displayed on lower display), it is still possible to change a decision and cancel preparation phase by pressing the button ︾ to return to IDLE state (lower display will show idLE). Preparation phase can be shorted with short press on the button PAR.

After program is started and preparation phase finished, initial pause is activated. Duration of this phase can be parameter adjusted, and when it is finished, programmer will advance into RUN state. Short press on button PAR shortens initial pause and starts program immediately. Program is now started and LED dot R and segment LEDs start to indicate further progress.

Second option to start the program is using the rear panel connectors (RPC). It is similar to first option with major difference that RUN state must be initiated directly - assigning value run to parameter StAt in usual way. The rest of the procedure is same as in first case. RPC gives an option to activate program from a distance at any time if system is correct and RPC are closed. Off course, all program parameters must be already set to desired values. There are 2 possible cases when these contacts are used:
- If contacts were CLOSED prior to program start, their opening and closing will start the program. This option is suitable when same program is frequently repeated. This way, starting of the program is much easier and faster compared to using front panel buttons.
- If contacts were OPENED, closing them will start the program. This works for both new program start and to continue earlier started program which had to be temporary stopped by opening of RPC.

Use button MODE to select which data will be shown on upper display: program setpoint, or measured temperature on input 1 or 2 (shown by LED dots T1 and T2).

4.8.2. Initial pause setup

After program is activated and preparing phase is over, programmer 3123 will start initial pause, i.e. it will delay start of program for few more seconds giving the operator enough time to set actual delay time for program start. This pause is determined by the value of parameter PAUS which is initially set to 0 (shown on lower display), as well as blinking of the LED PAUSE. While this display is active, operator can use buttons ▼ and ▲ to set up actual delay time before program starts. After set up is finished, leave the programmer untouched and it will start count down and automatically start the program when pause time pass.

During countdown, user can adjust new values for remaining time using buttons ▼ and ▲. If the value is set to zero and button PAR is pressed, programmer will immediately end pause time and start the program.
If after initial pause operator did not react and set actual pause (delay) time, programmer will ignore it and start the program immediately.

4.8.3. Program stop

Temperature programmer 3123 will automatically stop the operation when last segment of selected program is finished and set the IDLE state. In case earlier ending of the program is required, operator must use buttons, or communication instructions. Rear panel contacts do not allow permanent stop of the programmer, only temporary.

Program stopping can be provoked by selecting parameter StAt using button PAR and setting its value to idLE. After few seconds, programmer will stop operation and set to IDLE state. Speed up of this procedure is possible with short press on the button PAR after parameter StAt is set to idLE.

4.8.4. Program hold

When temporary stop of running program is required (to set programmer into HOLD state), operator can choose any of 3 available routines: using button commands, using RPC or through communication instructions.

Temporary stopping of the program using buttons is performed when parameter StAt is set to HOLd using buttons PAR, ▼ and ▲. Standard procedure will activate HOLD state after few seconds, but instant activation of HOLD state is possible with short press on button PAR after parameter setting. In this state, programmer timer is stopped and LED dot R and appropriate segment LED are constantly lit. Programmer will stay in this state until parameter StAt is assigned value run for program proceeding, or value idLE for program termination. While in HOLD state, PID controller will continue maintaining temperature at setpoint achieved in the moment when HOLD state is activated.
HOLD state can be initiated with opening of rear panel contacts (RPC) during RUN state. If RPC are opened while programmer is in IDLE state, there is no immediate setting of HOLD state, but if operator tries to set the programmer into RUN state, programmer will switch to HOLD instead because RPC are still opened. When HOLD state is set using buttons, it can not be changed using RPC.

4.8.5. Parameter change during program execution

While program is active, it is not possible to change program parameters. They must be adjusted properly while programmer is in IDLE state before the program is started. However, it is possible to view parameter values for currently active program and some code protected parameters which regulate control of the process can even be changed.

4.8.6. Power failure

During program run, all relevant data regarding programmer state, such as current program, segment, remaining time etc are saved in device's memory. In case power failure occur, programmer will have all relevant data available once the power is restored. Parameter PūrF will influence programmer action upon power supply is restored.
Parameter PūrF has 3 possible values:

- StoP - upon power supply restoring, programmer will stop the program and set active IDLE state
- rSEt - upon power supply restoring, programmer will start interrupted program again no matter which phase (segment) it was in
- cont - upon power supply restoring, programmer will continue program from the interruption point, but taking current measured temperature in count. Also, if there is big difference between setpoint and measured temperature due to long period without power, programmer will reverse the program up to the point where measured temperature match setpoint most, but taking care of parameter values in the specific restoring segment.

Parameter PūrF is in code protected list and can be changed independently, no matter which is current state of the programmer.

5. PARAMETER TABLES, PROTECTION LEVELS AND ACCESS RIGHTS

Temperature programmer 3123 has 2 levels of parameter protection:

- **operator level (code protected parameters)**
- **configuration level**

Operator protection level is formed in order to protect certain parameters from accidental change and unauthorized access. These parameters mostly influence quality of control and should be sometimes accessed and adjusted. Access procedure is described in chapter 3.2.2). Additional protection on this level is access right limitation. Access right determine which parameter will be show, adjustable or hidden. Access rights are assigned to each parameter on configuration level.

Configuration level provide highest protection level and request special access procedure. Once accessed, configuration level enables access to all parameters - including those vital to proper system function. Procedures for access right assigning and access code change are only possible on this level.

Configuration level is accessible depending on the position of special switch inside the device. While this switch is closed, only operator level is accessible. Once this switch is opened, configuration level access is possible. This operation should be performed only by trained personnel. Configuration level access procedure is described in chapter 5.2.

Regardless of current access level, certain parameters determine existence of other parameters in the list. In case some parameter is not shown in the list, it is because for current device setup their showing has no sense.

5.1. Parameter tables

Following tables show all parameters available.

Table 5.1. **General parameters**

PARAMETER LABEL / DESCRIPTION		**RANGE OF PARAMETER VALUES**	**DEFAULT VALUE**
UEr	Software version	Value shown on lower display can not be changed	
CodE	Access code	-999 to 9999	3123

Table 5.2. **Program parameters** (always accessible)

PARAMETER LABEL / DESCRIPTION		**RANGE OF PARAMETER VALUES**	**DEFAULT VALUE**
StAt	Programmer state	idLE - IDLE state - programmer inactive run - RUN state - program normally running HOLd - HOLD state - program temporary held Hb - HOLDBACK state	
Prog	Program ordinal number	1 to 8	1
r 1	First ramp in selected program	0.01 to 99.99 °C/min	10.00
L 1	First target level in selected program	Measurement range for selected sensor	25 25.0
t 1	First hold time in selected program	0 to 9999 minutes	1
r2	Second ramp in selected program	0.01 to 99.99 °C/min	10.00
L2	Second target level in selected program	Measurement range for selected sensor	25 25.0
t2	Second hold time in selected program	0 to 9999 minutes	1
End	Programmer behavior at the end of selected program	StoP - after program is ended, programmer will end all operation JmP.1 , JmP.2 , ... , JmP.8 - after program is ended, programmer will continue with next program which is selected here	StoP

Table 5.3. **Parameters accessible on operator level (code protected parameters)**

PARAMETER LABEL	DESCRIPTION	RANGE OF PARAMETER VALUES		DEFAULT VALUE
HbLo	Low holdback band (code protected program parameter)	1 to 9999 - integer display 0.1 to 999.9 - decimal point display OFF - turned off		20 20.0
HbH i	High holdback band (code protected program parameter)	1 to 9999 - integer display 0.1 to 999.9 - decimal point display OFF - turned off		OFF
ProP	Proportional band (if Ctrl is set to OnOF it represent hysteresis)	1 to 9999 - integer display 0.1 to 999.9 - decimal point display		10 10.0
int.t	Integral time constant (not shown when Ctrl is set to OnOF)	OFF - integral calculation turned off 1 second to 9999 seconds		360
dErt	Derivative time constant (not shown when Ctrl is set to OnOF)	OFF - derivative calculation off 1 second to 999 seconds		5
H c.t	Heat cycle time (not shown when Ctrl is set to OnOF)	1 second to 250 seconds		60
H A0	Type of alarm on output 3 for high limit of independent alarm	OFF - alarm off LAt - latched alarm nLAt - non-latched alarm		OFF
L A0	Type of alarm on output 3 for low limit of independent alarm	OFF - alarm off LAt - latched alarm nLAt - non-latched alarm		OFF
d A0	Type of alarm on output 3 for both limits of deviation alarm (high and low)	OFF - alarm off LAt - latched alarm nLAt - non-latched alarm		nLAt
H iAL	High limit of independent alarm on output 3	LoAL - to max measurement range for selected sensor		1200
LoAL	Low limit of independent alarm on output 3	Min measurement range for selected sensor to H iAL		-200
dHAL	High limit of deviation alarm on output 3	1 to 9999 - integer display 0.1 to 999.9 - decimal point display		50 50.0
dLAL	Low limit of deviation alarm on output 3	1 to 9999 - integer display 0.1 to 999.9 - decimal point display		2000 999.9
Sond	Sensor type (requires appropriate setting of DIP SWITCHes **SW1** and **SW2**)	FE J- type J (Fe-CuNi) n iCr - type K (NiCr-NiAl) FE L - type L (Fe-CuNi) r 13 - type R (Platinum 13% Rhodium / Platinum) S 10 - type S (Platinum 10% Rhodium / Platinum) b 30 - type B (Platinum 30% Rhodium / Platinum) Pt. 1 - Pt-100, decimal point display L in - linear input, integer display .L in -linear input, decimal point display	0 ÷ 750 ^{0}C 0 ÷ 1200 ^{0}C 0 ÷ 750 ^{0}C 300 ÷ 1600 ^{0}C 300 ÷ 1600 ^{0}C 600 ÷ 1700 ^{0}C -99.9 ÷ 599.9 ^{0}C -999 ÷ 9999 -99.9 ÷ 999.9	n iCr
Lt iP	Linear input type (shown only when linear inputs are selected, also requires appropriate setting of DIP SWITCHes **SW1** and **SW2**)	n0.1 - linear voltage input 0 ÷ 1V n0.10 - linear voltage input 0 ÷ 10V S0.20 - linear current input 0 ÷ 20mA		n0.1
Sn2	State of second input sensor	on - active OFF- inactive		OFF
Ctrl	Control type	P id - PID control OnOF - ON/OFF control		P id
PūrF	Programmer behavior in case of power loss and restoring	StoP - program is stopped rSEt - program is restarted from start cont - program continues from interruption point		cont

***Table 5.4.* Parameters available only on configuration level**

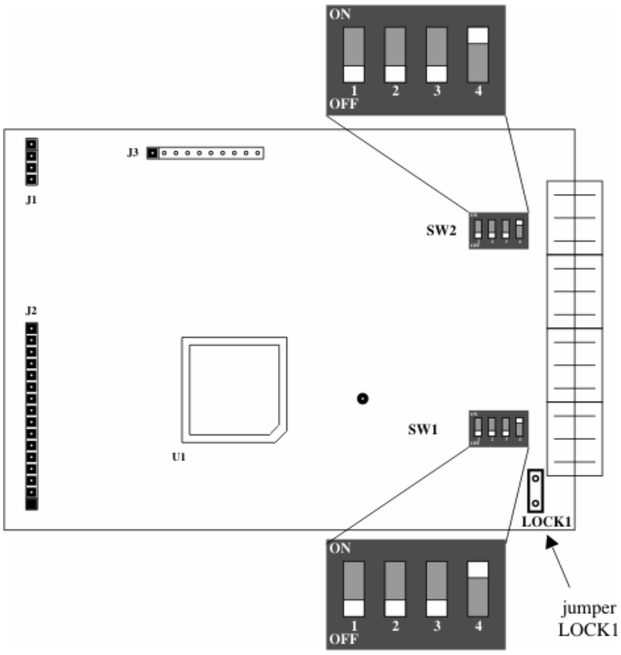
PARAMETER LABEL / DESCRIPTION		**RANGE OF PARAMETER VALUES**	**DEFAULT VALUE**
rEL.3	Defines output 3 as relay output	no - normal open nc - normal closed	nc
Addr	Communication address	0.1 to 3.2	0.1
bAUd	Communication speed	1200 , 2400 , 4800 , 9600 bauds	9600
CJC	CJC reference selection (shown only if some thermocouple is selected for sensor type)	int - internal reference for compensation 0 ^{0}C, 25 ^{0}C, 40 ^{0}C, 50 ^{0}C - outer reference for compensation	int
FiLt	Digital filter on input	1 , 2 , 4 , 8 , 16 , 32 , 64 , 128	4
OFS.1	Calibration offset for first input	-9.99 to 99.99	0.00
OFS.2	Calibration offset for second input	-9.99 to 99.99	0.00
Act	Output activity	d ir - direct output (output level rises when input level rises) rEU - reverse output (output level falls when input level rises) Choose rEU for heating, and d ir for cooling.	rEU
HiSb	High sensor break point for linear sensors	-999 to 9999 - integer display -99.9 to 999.9 - decimal point display	9999 999.9
LoSb	Low sensor break point for linear sensors	-999 to 9999 - integer display -99.9 to 999.9 - decimal point display	-99 -99.9
in_.1	Start value for input signal (only for linear sensors)	0 do 9999	0
rd_.1	Value displayed for input signal in_.1 (only for linear sensors)	-999 to 9999 - integer display -99.9 to 999.9 - decimal point display	0
in_.2	End value for input signal (only for linear sensors)	0 to 9999	9999
rd_.2	Value displayed for input signal in_.2 (only for linear sensors)	-999 to 9999 - integer display -99.9 to 999.9 - decimal point display	1000
ACCS	Access level granting procedure (shown only on configuration level)	HidE - selected parameter is hidden from list (forbidden access) rEAd - read only access to selected parameter ALtr - alterable parameter (free access)	

5.2. Configuration level access

Since this procedure require operations on the hardware, it should not be performed very carefully and only if very needed. Stop all active programs before proceeding with following:

- Turn off power supply, remove all clamps from the back side of device (if necessary label the clamps to ensure proper returning).
- Remove back lid and take PCBs out of device's case.
- Unlock jumper **LOCK1** at lower right side of main PCB (see picture 5.1)
- Return PCBs in the case, close the lid and reconnect clamps on the back side.
- Reconnect power supply.

Access to configuration level is now available. After all required changes on parameters are made, exit configuration level, and proceed with above procedure, only this time, return jumper LOCK1 to locking position (as it was prior to unlocking).



***Picture 5.1:* Main PCB layout with positions of jumper LOCK1 and DIP SWITCH -es SW1 and SW2**

5.3. Access code setup procedure

Changing of access code, which protects the parameters on operator level, is possible only on configuration level. Factory set password 3123 (default value) can be changed using following procedure:

- Enter configuration level (chapter 5.2).
- Now, all parameters are accessible and one of them is parameter CodE - access code. Press button PAR to scroll to this parameter.
- Use buttons ⏬ and ⏫ to set new desired code on lower display
- Wait for programmer to return to standard display mode
- Exit configuration level (chapter 5.2).

This ends access code change procedure. From now on, new password will be required to login to operators level. Make sure to memorize new code.

5.4. Access right limitation setup

Configuration level enables procedure for selection which parameters will be accessible, read only or hidden on operators level. This is access right limitation procedure. For certain selectable parameters access right can be assigned as:

- ALtr - free access - fully accessible and alterable parameter on operator's level,
- rEAd - partially accessible - parameter value is shown on operator's level, but can not be altered,
- HidE - forbidden access - parameter is hidden from list of parameters visible on operator's level - available only on configuration level.

Programmer 3123 is delivered with default setting of access right for each parameter which can be altered using following procedure:

- Enter configuration level (chapter 5.2.).
- Press button PAR to select symbol ACCS on upper display. This enables access right setup procedure.
- Press button ⏫ to select first parameter (symbol is shown on upper display, access right on lower display).
- Press button ⏬ to change access right for selected parameter.
- Press button ⏫ to select next parameter, and repeat above procedure for all parameters which require access right change.
- After completion, leave the programmer untouched to return to standard display (and save all changes into memory).
- Exit configuration level (chapter 5.2).

When access right for specific parameter is chosen, it is essential to pay attention to purpose of this procedure which is protection of certain key parameters for system operation, and limitation of parameters displayed on operator's level for faster and easier access. Operator level list of parameters should not contain parameters which are rarely changed or not changed at all during operation.

6. DEVICE SETUP

Parameters used for device setup are mainly only available when configuration level is enabled. These mainly comply to parameters which define purpose (activity) of outputs and adjustment of input characteristics. It is recommended to leave these parameters on configuration level for better system protection. Access to these parameters is describe in cheaper 5.2.

6.1. Control output activity setup (outputs 1 and 2)

Temperature programmer 3123 can setup activity of its outputs for purposes of heating and cooling. Parameter Act defines this with values:
- dir - for cooling function (direct output - output level rises when input level rises)
- rEU - for heating function (reverse output - output level falls when input level rises)

This parameter is same for both control loops. It can be accessed on configuration level using procedure described in chapter 5.2.

6.2. Alarms on output 3

6.2.1. Alarm types and limits

3rd output on programmer 3123 is used for alarm notification in case some occur during program run.
Alarm situations are mainly caused when temperature on either probe exceed some previously set limit. Limits are adjustable and can be independent (absolute - not relative to current setpoint) or bonded with setpoint. The way how 3rd output repot the alarm is also adjustable.

Independent (absolute) alarm limits are given as unchangeable values which temperature should not exceed, regardless the current setpoint. 2 such limits can be defined:
- **high independent alarm limit**
- **low independent alarm limit**

Both values can be chosen from entire sensor range. Only limitation is that high alarm limit must be set to higher temperature than low alarm limit, and vice versa.

Bonded limits are given as maximal allowed temperature deviations from current setpoint. In case difference between measured temperature and setpoint is higher than one defined by these limits, alarm is reported. Here too we differ 2 limits:
- **high deviation alarm**
- **low deviation alarm**

These 2 limits are set independently and can have any value.

Alarm state is reported by activating output 3 and displaying appropriate message on lower display. Alarm message is altered with standard display. Since it is possible to have several alarms active in the same time, different messages will be displayed for each of them. Possible messages are:
- H A0 - high independent alarm limit exceeded
- L A0 - low independent alarm limit exceeded
- d A0 - deviation alarm limit exceeded

Some operations require to have alarm notification active even after temperature is returned into allowed band, i.e. after condition for alarm notification is not present anymore. According to this, there are 2 different alarms:
- **latched alarm**
- **non-latched alarm**

Latched alarm stays active after alarm condition is not present anymore. To deactivate this type of alarm, button PAR must be pressed. If alarm condition is still present, alarm will not be deactivated. This type of alarm is used when production procedure can not be normally continued after alarm occurrence, and operator must confirm normal conditions for system operation are met.

Unlike this type, **non-latch alarm** is automatically deactivated when temperature returns to allowed band, i.e. alarm condition is not present anymore.

Each alarm type can be set independently so multiple combinations are possible.

Note: Alarm function with temperature programmer 3123 should only be used as a WARNING that certain irregular condition occurred, NOT AS A SAFETY OPTION. For increased system safety, independent protection system must be used.

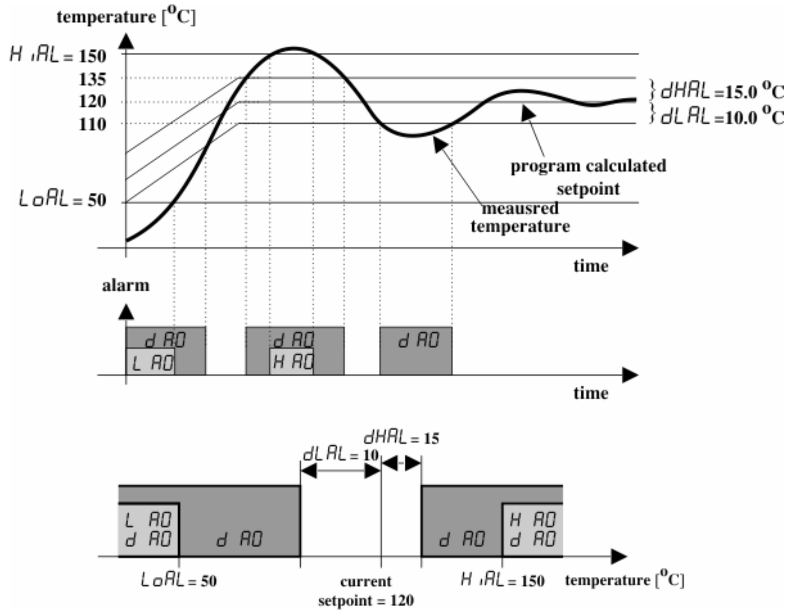
6.2.2. Alarm parameter settings

Code protected parameter list contains alarm parameters which define alarm as independent, deviation, latched and non-latched. These parameters are by default setting alterable, meaning their value can be altered in any moment. Parameters are selected with button PAR, and their value is changed with buttons ⏬ and ⏫.

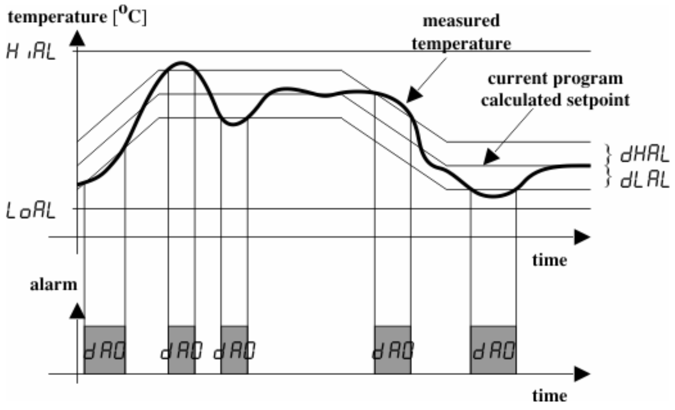
Following parameters define alarm types:
- H AO - high independent alarm
- L AO - low independent alarm
- d AO - deviation alarm

Above parameters can have values:
- LAt - latched alarm
- nLAt - non-latched alarm
- OFF - alarm disabled

Values of following parameters directly define temperature limits where alarm will be activated, in case that alarm type is enabled:
- HiAL - high independent alarm limit
- LoAL - low independent alarm limit
- dHAL - high deviation alarm limit
- dLAL - low deviation alarm limit



Picture 6.1: **Example of simultaneous occurrence of several non-latched alarms**



Picture 6.2: **Example of deviation alarm activation**

6.2.3. Alarm configuration on output 3

Output 3 is used for alarm notification. Temperature programmer 3123 is by default delivered with output 3 manufactured as non-voltage relay output (normal open or normal closed), but on customer request it can be manufactured as logical output. Relay output is best used for driving sound or light system for alarm signaling. Parameter rEL.3 defines relay output as normally open or normally closed. This parameter is available (by default) only on configuration level and is hidden form operator level. To access it, follow procedure described in chapter 5.2. Parameter values are:
- no - normally open - alarm is notified through working relay contact, while in absence if alarm, still contact is constantly on;
- nc - normally closed - alarm is notified through still relay contact, while in absence if alarm, working contact is constantly on

In order to enable alarm reporting, temperature programmer 3123 must be set for this action. It means that alarm limits are set, and appropriate alarm type turned on. Each alarm displays message on displays, activate output 3 and LED OUT 3 starts to blink. This LED is always on when working contact on relay of output 3 is active, i.e. always off when still contact is active.

Factory (default) setting imply alarm notification when still contact is activated. This means that during normal operation, in the absence of alarm, working contact of output 3 and LED OUT 3 are constantly on. This enables to use the output 3 as additional protection in case output 1, output 2 or both switch off and their control signals are sent through output 3. Also for activation of alarm in case temperature programmer 3123 lose power supply in case independent auxiliary power supply is used for alarm itself.

6.3. Inputs setting

6.3.1. Changing the type of input sensor and DIP SWITCH-es setting

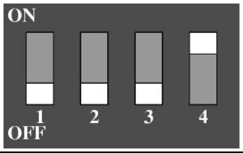
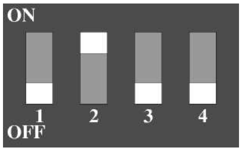
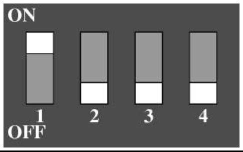
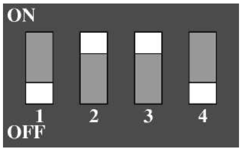
Up to 2 probes with same sensors can be connected to programmer 3123 inputs. Device is delivered already set up for one specific probe type, but user can change that if necessary. Adjustment for certain probe (sensor) is performed on operator level by adjusting the parameter Sond. Values for this parameter are given in table 5.3. When linear input is used, parameter L.t iP which defines type of linear signals must be also set. To avoid accidental change of sensor type, apart from setting proper value for Sond (and if necessary L.t iP), it is also required to set up jumpers on DIP SWITCH-es **SW1** and **SW2** which are located inside the case (see picture 5.1). Jumper layout on these switches must be same and comply with layout given in table 6.1 for specific sensor type.

Improper selection of type of input sensor may cause false measurement which could lead to serious problem during operation!

Input sensor adjustment procedure is as follows:
- Open case and access configuration level (chapter 5.2)
- Adjust jumpers on DIP SWITCH-es **SW1** and **SW2** according to actual sensor type and table 6.1.
- Return PCB in the case, reconnect clamps and power
- Press button PAR to list parameter Sond
- Use buttons ︾ and ︽ to adjust new value for sensor type according to table 5.3.
- If some of linear signals is selected for parameter Sond then parameter L.t iP. must be adjusted accordingly.
- Wait for programmer to return to standard display
- Exit configuration level (chapter 5.2).

We emphasize again that proper operation of 3123 programmer require that input parameters and DIP SWITCH-es jumpers must comply to actual sensor type.

Table 6.1. **DIP SWITCH-es setting**

Thermocouples and Pt - 100 probe	
Voltage input: 0 ÷ 1V	
Current input: 0 ÷ 10V	
Current input: 0 ÷ 20mA	

6.3.2. Linear inputs settings (scaling)

In order to adjust temperature programmer 3123 to actual voltage or current input signal, appropriate parameters available on configuration level must be adjusted.

Beside selecting linear probe and linear signal type (parameters Sond and L.t iP) and adjusting jumpers on DIP SWITCH-es, additional procedure called **linear input scaling** is required. This procedure defines which value will programmer display for which measured input. It is linking known input signals to specific display values. Input signal band is defined with value setting for high and low input limit. Maximal band for specific signal type which programmer can measure is divided into 9999 internal units, where minimal and maximal values are measured during production and can not be changed later. It is required to select appropriate values on inputs in internal units and link them to values which will be displayed on lower display.

To complete this scaling procedure there are 4 parameters which must be set. They are available only on configuration level (table 5.4). Parameter in_.1 is used to select starting value of signal and is given in internal units. Parameter rd_.1 defines value which will be displayed on lower display corresponding to input signal in_.1. Parameter in_.2 defines end input signal value given in internal units and parameter rd_.2 value which will be displayed on lower display corresponding to input signal in_.2.

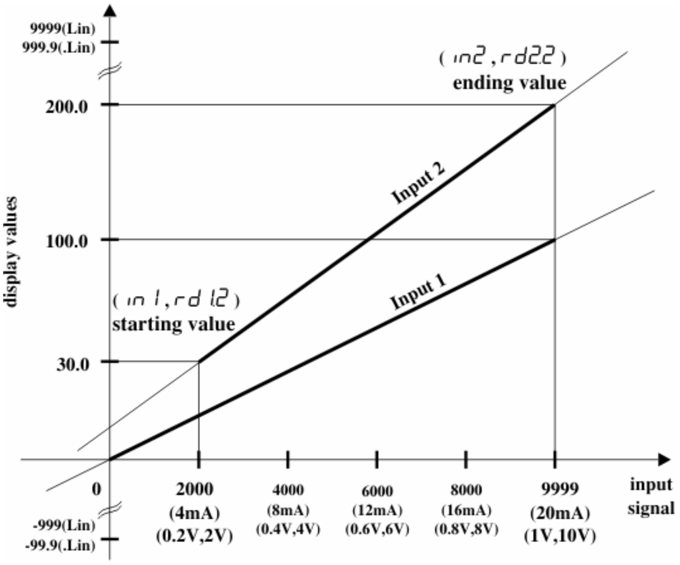
Linear inputs setting (scaling) require sensor type selection procedure execution (chapter 6.3.1), and additional setting of described parameters:
- Enter configuration level (chapter 5.2).
- Release jumper and set DIP SWITCH-es **SW1** and **SW2** into linear input position according to table 6.1.
- Return PCB in the case, reconnect clamps and power supply. Wait for programmer to enter standard display mode.
- Press button PAR to scroll to parameter Sond.
- Use buttons ︾ and ︽ to set parameter Sond to value L in or .L in (for integer or decimal point display), and then set parameter L.t iP to appropriate value for linear input according to table 5.3.
- Set parameter in_.1 to starting value of input signal, and set parameter rd_.1 to starting display (measurement) value. Set parameter in_.2 to ending value of input signal, and set parameter rd_.2 to starting display (measurement) value.
- Wait for programmer to enter standard display mode.
- Exit configuration level (chapter 5.2.).

EXAMPLE 1:
In case we would like to set the programmer to display values in range 0 to 100 for input signal of 0 to 1V, parameters must be set as:

Sond = L in
L.t iP = n0.1
in_.1 = 0
rd_.1 = 0
in_.2 = 9999
rd_.2 = 100

EXAMPLE 2:
In order to set the programmer to display values in range 30.0 to 200.0 for input signal of 4 do 20mA, settings are:

Sond = .L in
L.t iP = S0.20
in_.1 = 2000
rd_.1 = 30.0
in_.2 = 9999
rd_.2 = 200.0



Picture 6.3: **Examples of linear input (scaling) setting**

6.3.3. System with 2 probes (2 control loops)

Temperature programmer 3123 can operate in systems with one or two input probes (i.e. control loops). Default setup consider second probe deactivated, so in case it is used, it must be activated by setting parameter Sn2 to on (second probe is deactivated if value is OFF). Parameter Sn2 is by default located on operator level among other code protected parameters. Connected probes MUST be of same type.

After second probe is enabled, use button MODE to display measured temperature on upper display as well as first one. This temperature is calculated also for program run, holdback, etc...
Although started program is same for both control loops (program setpoint is same for both loops), control is independent in each control loop. Control parameters and program are same, but actual conditions on the probes may require independent control. This imply that approach with 2 probes is recommended only in situations where control loops have only slightly different characteristics. When control loops are very different and there is little correspondence between them, it is better to choose separate programmers for each control loop.

6.3.4. Thermocouples cold junction compensation (CJC)

When thermocouple sensor type is selected, parameter list will show parameter CJC which defines type of compensation at open end of thermocouple or compensating cable. Compensation can be internal (int) when thermocouple or compensating cable open ends are connected to device itself and device itself determines compensation values, or it can have some of fixed values: 0 , 25 , 40 , 50 ^{0}C, which is used when compensation box at selected temperature is used. Default value for this parameter is int.

6.3.5. Offset

Sometimes, measurement correction must be made. Reasons might be different, but most common are:

- **probe zero error correction**: when new probe is installed, there can be a difference in measurement compared to old replaced probe
- **thermal gradient compensation**: appropriate correction can be made if difference of temperature at thermocouple end and point of temperature measurement is known
- **probes pairing**: used when identical reading from 2 probes is required. Difference due to zero error on each probe or actual temperature difference may be compensated using offset.

Values of parameters OFS.1 and OFS.2 are in the programmer added to measured values and these new values are treated as actual values shown on display and calculated for control. Possible values for these parameters are in range -9.99 to 99.99. Default values are: 0.00.

6.3.6. Input signals filtering

Certain disturbance may occur on measurement equipment (probe, transmitter, cables) or programmer itself which may cause instability in measured value (displayed on upper display) and even disturbance in the control process itself. To reduce influence of disturbances on inputs, signal filtering parameter is used. Parameter FiLt can have values: 1 , 2 , 4 , 8 , 16 , 32 , 64 and 128. Higher filter value means lower possibility that some disturbance cause change in temperature reading, but in the same time slows down measurement process, which can be important for process control. Value for filter is best chosen when it eliminates disturbances well, but does not slow down measurement a lot. Default value for this parameter is 4.

6.3.7. Sensor break for linear probes

When certain linear type probes are used, programmer 3123 can define certain limits (band) within which measurement is considered to be correct. In case that signal on certain probe exceed values defined within these limits, upper display will show warning message that irregular input signal value is detected and controller will set special **operation mode with faulty probes** (chapter 6.3.8).
Parameter H iSb sets high sensor break limit, while LoSb sets low sensor break limit. Values are given in °C.

6.3.8. Working with faulty probes

Temperature programmer 3123 has a feature of automatic detection of errors in input signals, and when this happens, programmer report this condition with message Snbr displayed instead of measured temperature on upper display. Simultaneously, symbols Snb. 1 i.e. Snb.2 are flashed on lower display depending on the input number where error is detected. Also, programmer immediately switch to special operating mode which is appropriate for this situation which is:

- immediately switch off appropriate output (output 1 if error is detected on probe 1 and output 2 if error is detected on probe 2).
- set HOLDBACK state and display massage to notify user of current state

This state is active until error is removed, or until program is ended and operator activates IDLE state.

If error is detected while programmer was in IDLE state, only error message is displayed and there is no additional changes in state or operation, with exception that activation of RUN state is not possible.

After signal error is resolved, programmer automatically terminates this operating mode and continues normal operation.

7. CONTROL PARAMETERS AND TYPE OF CONTROL

Main task of temperature programmer 3123 is running temperature processes according to set program, which require calculation of changeable temperature setpoint and providing adequate process control in order to meet temperature requirements.

Well chosen and set control parameters provide good system functioning and contribute to increased quality of final product, efficiency and energy saving. Control parameters are available on operator level and can be accessed after access code is entered. Some parameters may not be visible in the list if due to setting of other parameters it does not make sense for them to be displayed. List of control parameters, range of possible values for each parameter and default settings is given in chapter 5.1.

7.1. Control types

Temperature programmer 3123 offers **2 types of control**:
- **PID control**
- **ON/OFF control**

Type of control is selected by setting parameter Ctrl to one of available values:
- Pid - PID control is selected
- OnOF - ON/OFF control is selected

Press the button PAR until parameter Ctrl is displayed and then use buttons ⏬ and ⏫ to make a selection. Selected control type is same for both control loops.

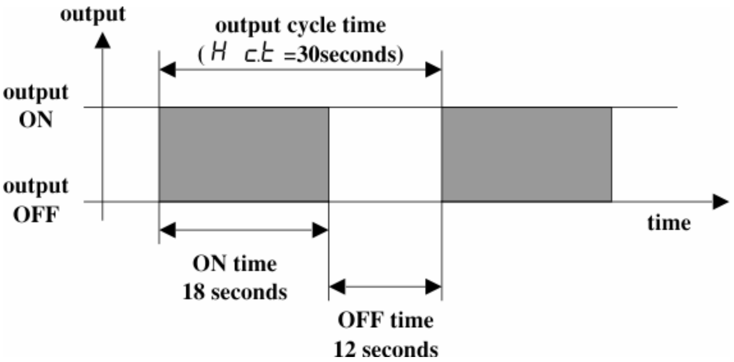
7.2. PID control parameters

7.2.1. Output activity cycle and output level

PID control is performed by activation/deactivation output in certain rhythm. This rhythm is determined by **heat cycle time**. Cycle duration is time between 2 output activation, i.e. sum of time while output is active and inactive (see picture 7.1).

Parameter **heat cycle time** is labeled with symbol H c.t and is available in the code protected list of parameters by default. This parameter can be adjusted in range 1 to 250 seconds and is same for both control loops.

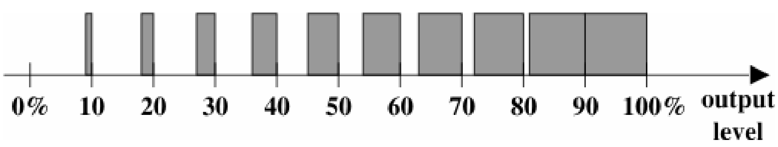
Output level is defined as ***percentage ratio of duration of activity within the cycle and total cycle time for an output***. Value is set in range 0 to 100%. For example, if output level is set to 60% and heat cycle time is 30 seconds, then time of output activity is 18 seconds, and time of inactivity is 12 seconds (as shown on picture 7.1).



Picture 7.1: **Output activity within heat cycle with output level of 60%**

IT is clear that energy induced in the system is directly dependant on output level. The higher the output level is, the more energy will be induced to the system.

When both outputs are activated, output level is constantly calculated for each output independently, because conditions in first and second control loop can differ a lot, so different response from programmer is required on each output.



Picture 7.2: **Output activity depending on output level**

7.2.2. Proportional band

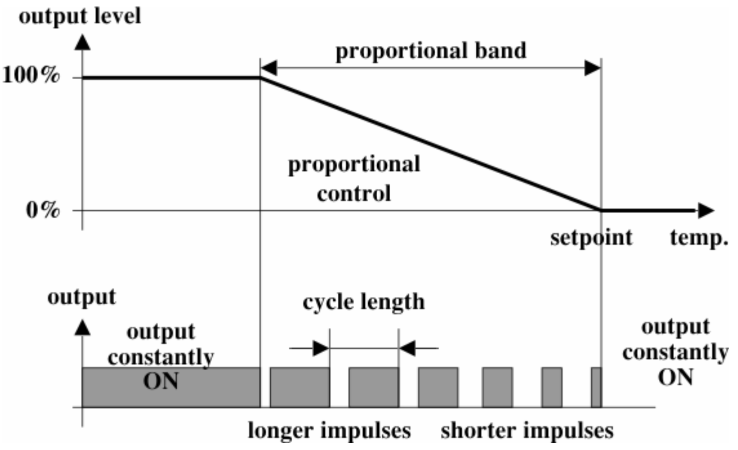
Proportional band is a band within which **proportional control** is performed. To set this band, one limit is on the setpoint, and second is below that setpoint (for heating) and above that setpoint (for cooling). As the setpoint shift during program run, so does this band also.

While measured temperature during program running is within proportional band, programmer will calculate error - difference between setpoint and measured temperature. Based on this error, programmer calculates output level, i.e. ratio of output activity and inactivity, in order to minimize this error. This way, programmer controls amount of energy induced to system. So when the error is great (meaning measured temperature is far from setpoint) - output level will be high, and vice versa - as the error is lowered, so will the output level.

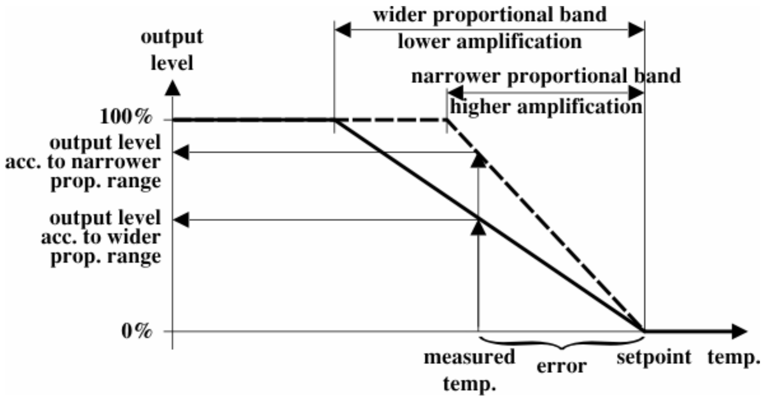
If measured temperature is out of proportional band and setpoint is overshot (setpoint is lower than measured temperature) controller will set the output level to 0% and completely turn off the output, and vice versa, if the temperature is undershoot, it will set output level to 100%. Parameter responsible for adjusting the **proportional band width** is labeled with symbol ProP. Value of this parameter is given in ^{0}C.
Example of relation between proportional band and setpoint is given on picture 7.3.

Proportional band is indirectly proportional to gain which amplifies the error between setpoint and measured value to establish a power level. Narrow proportional band increase system sensibility to errors because of high gain (amplification), and wider proportional band lowers the system sensibility to errors because of low gain (amplification), as shown on picture 7.4, so it is crucial to select proper proportional band for each system. Too wide proportional band can produce significant inertia and settling the temperature away from setpoint. Too narrow proportional band will produce oscillations around setpoint due to high system sensibility to error.

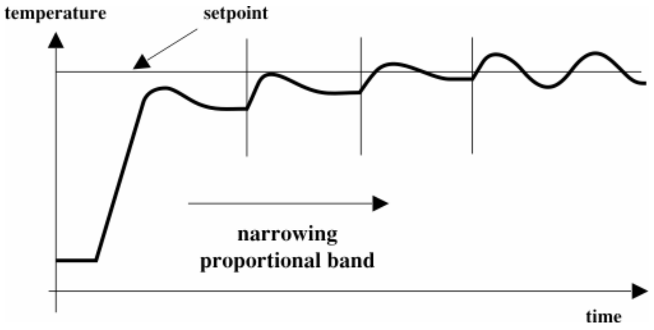
Influence of proportional band to quality of control is shown on example where only proportional control is changed during hold segment (picture 7.5). At first, wide proportional band is set so temperature stabilize at much lower value than setpoint, With gradual narrowing of the proportional band, temperature will come closer to setpoint. If too narrow proportional band is set, then the temperature will start to oscillate around setpoint. So, in conclusion, it is best to choose as narrow proportional band as possible, but in that way that temperature oscillations do not occur.



Picture 7.3: **Relation between proportional band and setpoint**



Picture 7.4: **Influence of proportional band width to amplification**



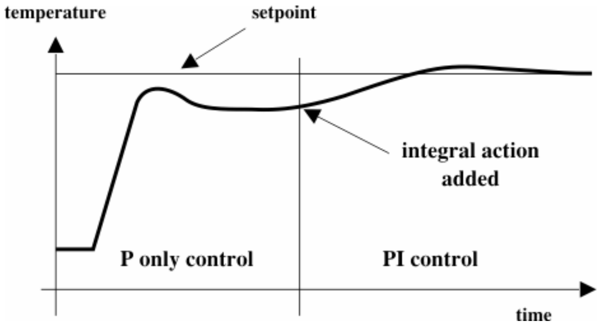
Picture 7.5: **Influence of narrowing the proportional band (P type of control)**

7.2.3. Integral time constant

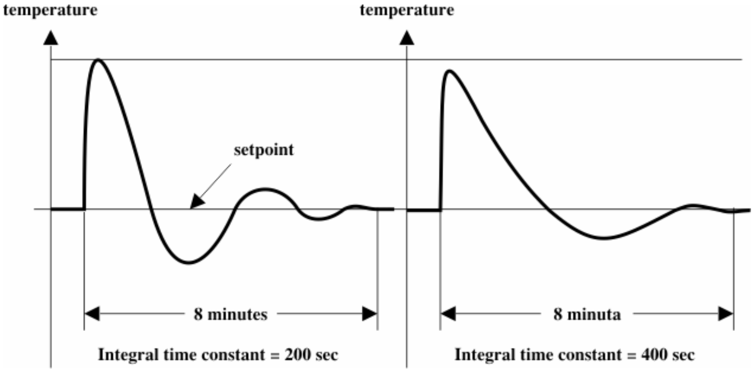
Integral action, or Automatic Reset, is probably the most important factor governing control at setpoint. Parameter **integral time constant** (integral term) introduces integral action into control. This parameter is labeled with symbol int.t, and its value is given in seconds. If integral action is turned off by setting the parameter int.t to value OFF, only proportional control will remain.

Integral term slowly shifts the output level as a result of an error between setpoint and measured value. If the measured value is below setpoint the integral action will gradually increase the output power level in an attempt to correct this error. This action does not allow temperature to settle at the level far below setpoint, which is characteristic for pure proportional control. Picture 7.6 demonstrates the result of introducing integral action. In the beginning, only P control is used. Once the temperature settles at certain level below setpoint, integral action is introduced. After that, temperature slowly rises until it reaches setpoint.

If the integral time constant (term) is set to a fast value the power level could be shifted to quickly thus causing oscillation since the controller is trying to work faster than the load can change. Conversely, an integral time constant which is too long will result in very sluggish control. Picture 7.7 demonstrates influence of widening integral time constant. It is noticeable that lengthening the integral time constant result in slower system response.



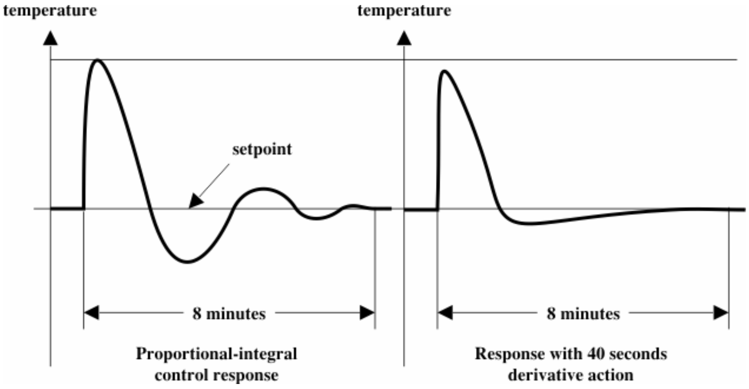
Picture 7.6: **Adding integral action**



Picture 7.7: **Widening integral time constant**

7.2.4. Derivative time constant

Derivative action, or rate provides a sudden shift in output power level as a result of a quick change in measured value. If the measured value drops quickly the derivative term will provide a large change in output level in an attempt to correct the perturbation before it goes too far. This action is most beneficial in recovering from small perturbations.
Derivative action is introduced with adjustment of **derivative time constant** parameter labeled with symbol dErt. Value is adjusted in seconds, and can be turned off by selecting the value OFF.
If the parameter is set to too high value, system will oscillate around setpoint, while too short value does not provide fast enough response to sudden disturbances. Picture 7.8 shows influence of derivative action to sudden changes in system.



Picture 7.8: **Adding derivative action**

Derivative action is often mistakenly associated with overshoot inhibition rather than transient response. In fact, derivative should not be used to curb overshoot on start up because it will worsen system performance during temperature holding segment.

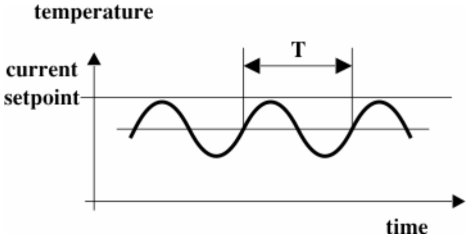
7.2.5. Procedure of parameter settings for PID control

Each system has its own characteristics so it is necessary to adjust control parameters for each system independently in order to achieve best quality of control. There are many procedures for parameter adjustment. Most commonly used is **closed-loop cycling method**. This method is applicable where system allows overshoots during setup. For system which does not allow great temperature oscillations, some other method must be used.
Closed-loop cycling method is performed as follows:

- Make sure system is setup for normal operation (inputs and outputs connected, power supply provided, etc....)
- Prepare test program which has large values for first ramp segment r 1 and first hold time t 1. First target level L 1 should be set to half the setpoint value usually used.
- Select PID control (set parameter Ctrl to value P id).
- Turn off integral and derivative action (set int.t and dErt to value OFF) and both holdback band (HbLo and HbHi).
- Activate test-program.
- Reduce duration of active output cycle as much as system allows it.
- Reduce the value of proportional band (ProP) to lowest possible value. This will force system into oscillation after setpoint is reached.
- Measure the time system requires for one full oscillation - **oscillation period T** - in seconds (if possible find the average of several oscillations for most accurate determination of oscillation period).
- Slowly increase proportional band until system stabilize. Value of proportional bend for which the system stabilize is referred to as **critical gain P** (or point of ultimate sensitivity).
- For values **T** and **P** calculated in this manner, set the PID control parameters according to following table:

Table 7.1: **Closed-loop cycling tuning constants (optimal)**

Type of control	**Proportional band (ProP)**	**Integral time constant (int.t)**	**Derivative time constant (dErt)**
P control	**2 P**		
PI control	**2.2 P**	**0.8 T**	
PID control	**1.7 P**	**0.5T**	**0.12 T**



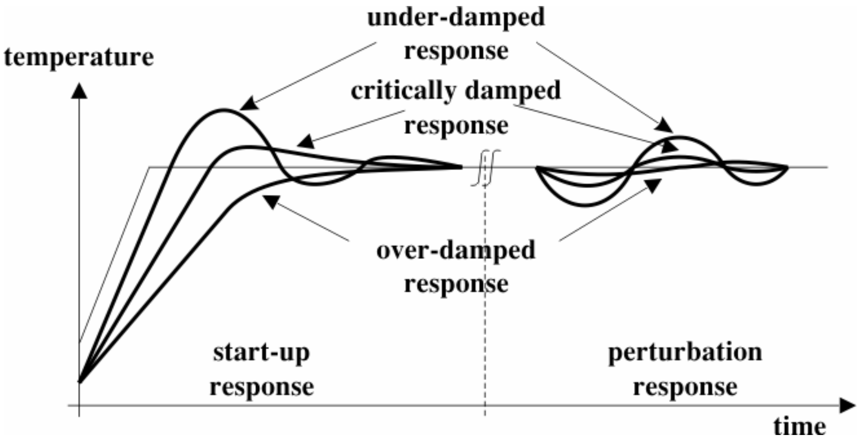
Picture 7.9: **Oscillation period T determination during closed-loop cycling method**

- Stop test-program, adjust program parameters to desired value (do not change PID parameters) and restart test-program for observation of system behavior with newly adjusted PID parameters.

We emphasize that this procedure provide optimal PID parameters for holding segment. Ramp segment might need additional adjustment which depends on type of ramp and system properties. Even PID parameters might be taken only as starting values, and they can finally be adjusted to desired values which will produce:

- setpoint reach with minimal overshoot
- stable temperature holding at setpoint with minimal deviation
- fast response for sudden perturbation caused by outer interference, as well as fast restoration of target level

Typical system responses while reaching target level are shown on the picture 7.10.
Under-damped response generates large overshoot while reaching target level and has significant oscillation same for start up response and perturbation response. In this case, integral time constant and proportional band must be increased to increase damping and reduce oscillations.
Over-damped response does not generate overshoot during start-up response, but the system reaches target level too slowly and has slow response for perturbations. To speed up the system, reduce the values for integral time constant and proportional band.



Picture 7.10: **Typical system response curves**

In cases where system during hold segment shows signs of moderate instability with small oscillations, where they are not produced by outside perturbation, following steps might help:

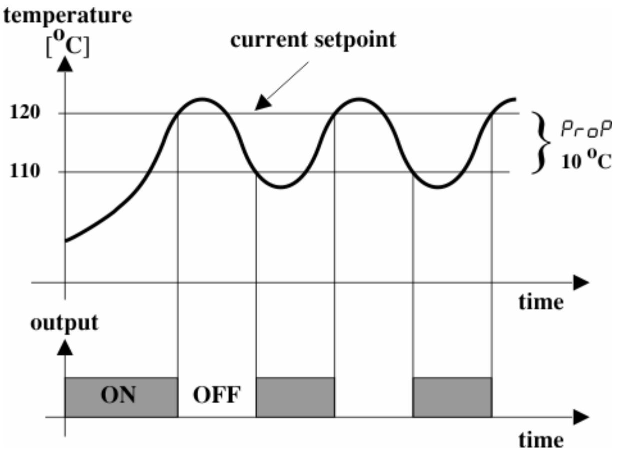
- Compare period of those oscillations with integral time constant int.t. If integral time constant is smaller (in seconds), increase its value to oscillation period.
- If the system still oscillates even with adjusted integral time constant, increase of proportional band (ProP) should be tried.
- Certain systems do not allow usage of derivative action (dErt). Even though this action provide stabilization in many systems, certain systems (such as hot air heating systems) introduce large transport delays, so the use of derivative actions cause instability. If the system can not stabilized with integral action and proportional band parameters increase, try reducing derivative time constant or even deactivate this type of control by setting its value to OFF.

7.3. ON/OFF control parameters

ON/OFF type of control assumes turning on and off appropriate output at defined temperature limits. These limits are relative to setpoint and defined by the parameter **hysteresis**.
ON/OFF type of control is used in systems which do not demand high precision control, but certain deviation from setpoint is allowed. This type of control is also recommended for systems where frequent turning on and off of the outputs is not desired.
To select this type of control, set the value of parameter Ctrl to value OnOF.

7.3.1. Hysteresis used in ON/OFF control

Hysteresis is a difference between the temperatures at which control output is turned on and turned off. Output **turns off** at setpoint, but it is **turned on** again at lower temperature. For heating this value is below setpoint and for cooling it is above setpoint and in both case it is shifted from first limit for **hysteresis value**. During normal program run, setpoint is changed, so these limits are changed with it. Parameter for hysteresis adjustment is labeled with ProP (same as for proportional band) and it given in °C.
Turning the output on/off at limits defined by hysteresis for heating function is shown on picture 7.11.

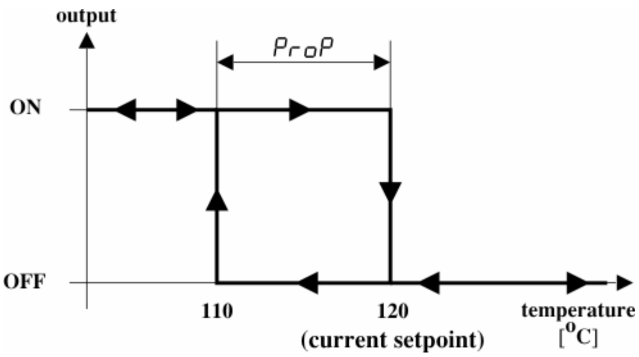


Picture 7.11: **Example of ON/OFF type of control**

Previously described process can be presented also on following way:

Example of ON/OFF control for heating function:

Setpoint = 120 [^{0}C]   ProP = 10 [^{0}C]
ctrL = OnOF   Act = rEU



Example of ON/OFF control for cooling function:

Setpoint = 120 [^{0}C]   ProP = 10 [^{0}C]
Ctr l = OnOF   Act = d ir

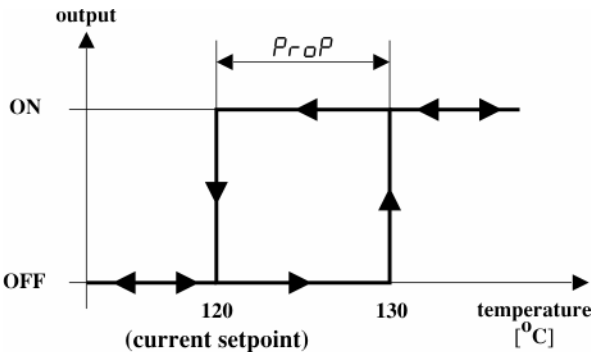


TABLE OF CONTENTS:

1. TECHNICAL DATA 1
1.1. Ordering code 2
2. INSTALLATION 3
2.1. Power supply connection 3
2.2. Outputs connection 3
2.3. Inputs connection 3
2.4. Rear program conector (RPC) connection 3
2.5. Comunnication connection 3
3. OPERATION 4
3.1. Standard display mode 4
3.2. Device parameters 5
3.2.1. Selecting and adjusting parameter values 5
3.2.2. Code protected parameters (CodE) 5
3.3. Error report 5
4. OPERATING PRINCIPLES 6
4.1. Programmer structure 6
4.2. Program segments 7
4.3. Programmer states 7
4.3.1. IDLE state 8
4.3.2. RUN state 8
4.3.3. HOLD state 8
4.3.4. HOLDBACK state 9
4.4. Programmer control 10
4.4.1. Program control using front panel buttons 10
4.4.2. Program control using rear panel contacts (RPC) 10
4.5. Program parameters 10
4.5.1. Program number 10
4.5.2. Segment parameters 11
4.5.3. Programmer behaviour at the end of the program 11
4.6. Holdback band 11
4.7. Program setup 12
4.8. Program start and operation 13
4.8.1. Program start 13
4.8.2. Initial pause setup 14
4.8.3. Program stop 14
4.8.4. Program hold 14
4.8.5. Parameter change during program execution 14
4.8.6. Power failure 14
5. PARAMETER TABLES, PROTECTION LEVELS AND ACCESS RIGHTS 15
5.1. Parameter tables 15
5.2. Configuration level access 17
5.3. Access code setup procedure 18
5.4. Access right limitation setup 18
6. DEVICE SETUP 19
6.1. Control output activity setup (outputs 1 and 2) 19
6.2. Alarms on output 3 19
6.2.1. Alarm types and limits 19
6.2.2. Alarm parameter settings 20
6.2.3. Alarm configuration on output 3 20
6.3. Inputs setting 21
6.3.1. Changing the type of input sensor and DIP SWITCH-es setting 21
6.3.2. Linear inputs settings (scaling) 21
6.3.3. System with 2 probes (2 control loops) 22
6.3.4. Thermocouples cold junction compensation (CJC) 22
6.3.5. Offset 22
6.3.6. Input signals filtering 22
6.3.7. Sensor brak for linear probes 23
6.3.8. Working with faulty probes 23
7. CONTROL PARAMETERS AND TYPE OF CONTROL 24
7.1. Control types 24
7.2. PID control parameters 24
7.2.1. Output activity cycle and output level 24
7.2.2. Proportional band 25
7.2.3. Integral time constant 25
7.2.4. Derivative time constant 26
7.2.5. Procedure of paramater settings for PID control 26
7.3. ON/OFF control parameters 28
7.3.1. Hysteresis used in ON/OFF control 28