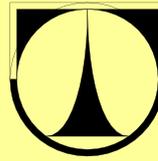




**University of Applied Sciences
Zittau/Görlitz**
Theodor-Körner-Allee 16
D-02763 Zittau



**Technical University of
Liberec**
Studentska 2, 461 17 Liberec 1
Czech Republic

**Faculty of
Electrical Engineering
and Informatics**

**Faculty of
Mechatronics and Interdisciplinary
Engineering Studies**

Experiment Instructions

August 2018

TCsB Temperature Control of a solid Body by Controller SIPART DR20

Dr.-Ing. Steffen Gärtner
Doc. Ing. Osvald Modrlák



Faculty of Electrical Engineering and Informatics
Process Automation / Measurement Engineering



Contents

1	Goal of the experiment	3
2	Basic principles of feedback control	3
2.1	Analysis of feedback control.....	3
2.1.1	Basic structure of control.....	3
2.1.2	The structure of feedback control	4
2.2	Model of control.....	5
2.2.1	Open-loop control.....	5
2.2.2	Closed-loop control	6
2.3	Controller	7
2.4	Root-Locus Design Method	8
2.4.1	Structure of controller by using the Root-Locus Design Method for tuning.....	8
2.4.2	Basic properties of Root Loci.....	8
2.4.3	Time- domain specifications	9
2.4.4	Program packet "sisotool"	11
2.4.5	Generally rules for controller zeros setting	12
3	Set-up of the experiment	16
3.1	Structure of experiments	16
3.2	Controlled plant	17
3.3	Controller SIPART DR 20.....	18
3.4	Data acquisition and control	19
4	Operating the equipment and software used in the experiment	20
4.1	Controller SIPART DR20.....	20
4.1.1	Parameterization	20
4.1.2	Structuring.....	21
4.1.3	Controller output functions of controllers are:	22
4.2	Data acquisition and control	22
5	Conducting the experiment.....	24
5.1	Task of the experiment.....	24
5.2	Conducting of measuring and recording on the SITRAIN DR 20	24
5.3	Instruction to the identification measurement	24
5.4	Modelling.....	27
5.4.1	Modelling of the controlled system	27
5.4.2	Theoretical analysis of the control loop	27
5.4.3	Studies on the model of the control loop.....	27
5.5	Studies on the real feedback system (control loop)	27
6	Analysis of the experiment	29
7	Literature.....	29

1 Goal of the experiment

- Analysis of an electric heating component
- Identification and modelling using the MATLAB modelling language
- Verification
- Controller parameter optimisation
- Handling of an industrial control unit

2 Basic principles of feedback control

2.1 Analysis of feedback control

2.1.1 Basic structure of control

In general two generic control configurations are used: open-loop and closed-loop systems.

Open-loop systems

The generic open-loop system is shown in figure 2-1. The “**Input transducer**” converts the form of input w^* to that used by the controller. The “**Controller**” drives the “**Process**” or “**Plant**” with the controller output u . The input w^* is sometimes called the “**reference**” while the output of the plant y_s can be called “**controlled variable**”. The “**disturbance**” d is an undesirable input to the plant.

Open-loop systems do not correct for disturbances and are simply commanded by the reference signal.

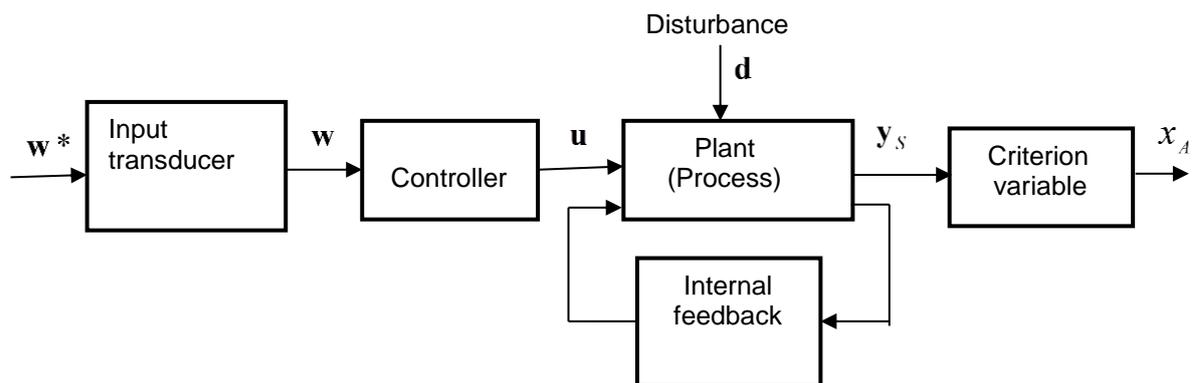


Fig. 2-1: Open loop system

Closed-loop systems

The generic architecture of a closed-loop system is shown in figure 2-2. The input transducer converts the form of the input w^* to the form w of the controller known as “**desired value**”. The summing junction algebraically subtracts the feedback signal y , which arrives via the **feedback path**, from the desired value/set point signal w . The resulting signal e is generally called “**error**”.

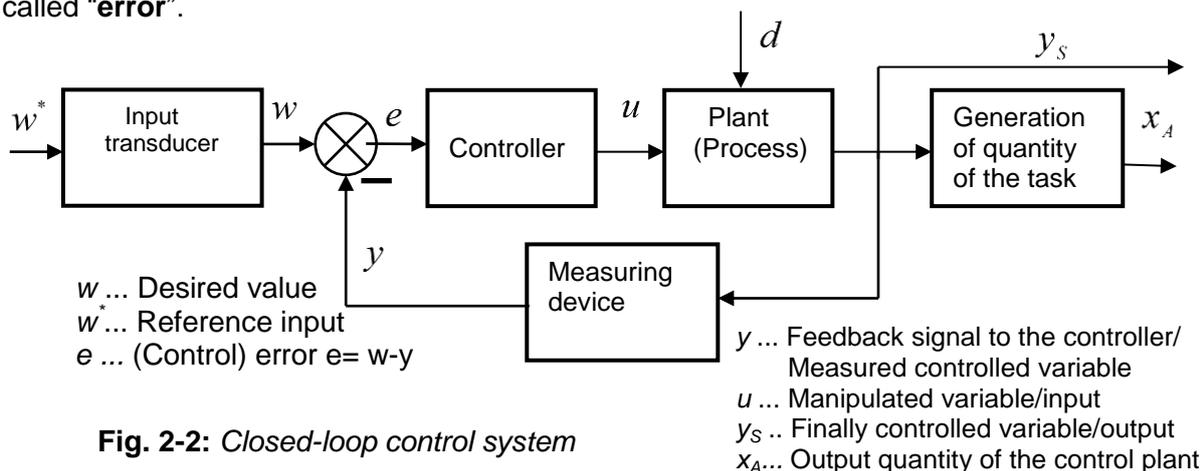


Fig. 2-2: Closed-loop control system

The controller generates a signal $u(t)$ known as “**manipulated signal**”, which is the input to plant (process).

The **closed-loop system** compensates for disturbances by measuring the output response $y(t)$, feeding that measurement back through a feedback path, and comparing that response to that input $w(t)$ at the summing junction. If there is any difference between the two responses, the system drives the plant, via error (control error) “ e ”, to make a correction. If there is no difference, the system does not drive the plant, since the plant response is already the desired response.

2.1.2 The structure of feedback control

A block scheme of a technological process is shown in figure 2-3. We can denote as extended/generalized plant a part of the scheme that consists of plant (process) and measuring devices (see figure 2-4). We recall that a **plant** contains amplifiers with transducers and the technological process with actuator.

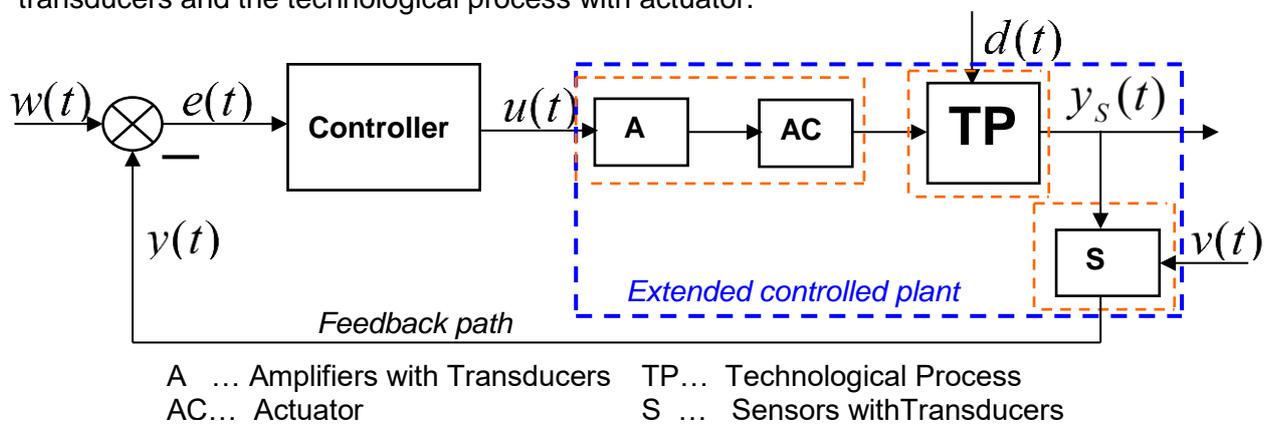


Fig. 2-3: Scheme of a controlled plant/process with measuring devices

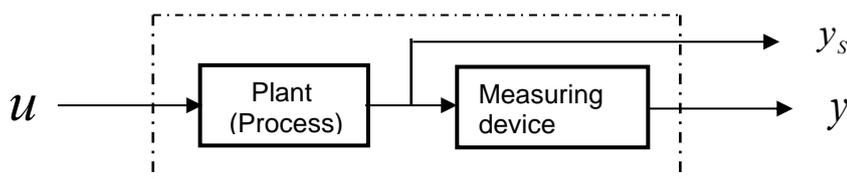


Fig. 2-4: Extended/generalized controlled plant

The measuring device embodies sensors with transducers or transmitters. The dynamic of the extended plant is generally approximated by three transfer functions $F_V(s)$, $F_S(s)$ and $H(s)$. The transfer function $F_V(s)$ approximates the dynamic of amplifiers, transmitters and actuator, the transfer function $F_P(s)$ approximates the dynamic of the process and the transfer function $H(s)$ approximates the dynamic of measuring devices (see the figure 2-5).

Transfer function of the extended controlled plant $F_U(s)$

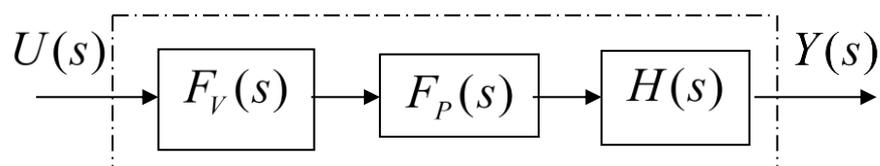


Fig. 2-5: Transfer function of the extended controlled plant

Finally, the transfer function **notation** is often simplified by lumping all instrumentation and process dynamic into one term F_U . This is equivalent to the following expression

$$F_U(s) = \frac{Y(s)}{U(s)} = F_V(s) \cdot F_S(s) \cdot H(s) = F(s) = \frac{B(s)}{A(s)} \quad (2-1)$$

The relationship between the disturbance $d(t)$ and the measured **controlled** variable $Y=Y_D$ can be approximated by the transfer function $F_D(s)$

$$F_D(s) = Y_D(s)/D(s) = \frac{C(s)}{A(s)} \quad (2-2)$$

(C(s) is polynomial)

Under these conditions the model structure of an extended controlled plant with the modeled disturbance d is shown in the figure 2-6.

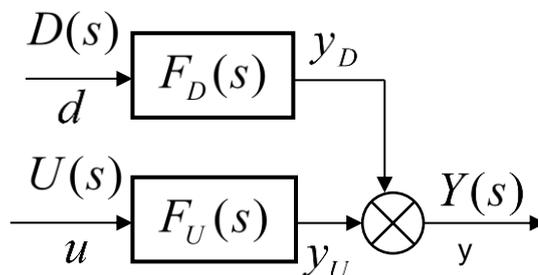


Fig. 2-6: Model structure of an extended plant with modeled disturbance

The output of the model is $y(t) = y_U(t) + y_D(t)$.

The output of the system for joint action of manipulated variable and of the disturbance is equal to

$$Y(s) = F_U(s) \cdot U(s) + F_D(s) \cdot D(s) = \frac{B(s)}{A(s)} U(s) + \frac{C(s)}{A(s)} D(s) \quad (2-3)$$

The classical control uses models of control systems that are described by transfer functions. The block diagram of a open-loop control with disturbance is shown in figure 2-7. The block diagram of a closed-loop control with disturbance model is shown in the figure 2-8.

2.2 Model of control

2.2.1 Open-loop control

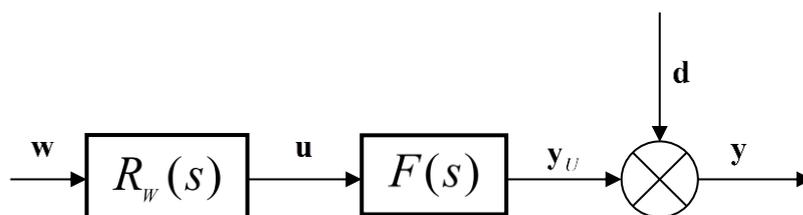


Fig. 2-7: Model of an open-loop control system

The Laplace transformation of output is equal

$$Y(s) = F(s) \cdot R_w(s) \cdot W(s) + D(s) \tag{2-4}$$

where are $F(s) = \frac{b_m s^m + \dots b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0}$.

$R_w(s)$... Transfer function of a feed forward controller,

$W(s)$... Laplace transform of set point/desired value.

$D(s)$... Laplace transform of the disturbance.

It is obvious from (2-4) that the disturbance cannot be compensated.

2.2.2 Closed-loop control

Using the general transfer function (2-1) and (2-3) the block diagram of the feedback control can be shown in the figure 2-8.

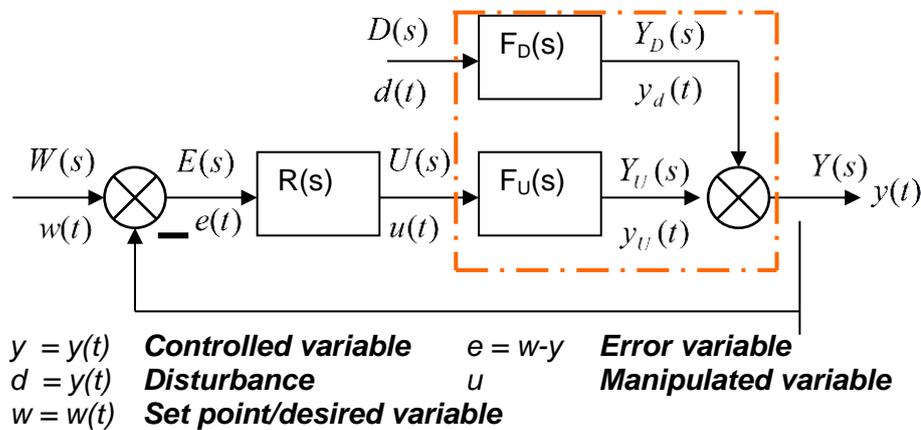


Fig. 2-8: Block diagram of a general closed-loop feedback model control system

Simplified closed loop disturbance transfer function:

$$F_{yD}(s) = \frac{Y_d(s)}{D(s)} = \frac{F_D(s)}{1 + R(s)F_U(s)} \tag{2-5}$$

Simplified set point transfer function:

$$F_{yW}(s) = \frac{Y_w(s)}{W(s)} = \frac{R(s)F_U(s)}{1 + R(s)F_U(s)} \tag{2-6}$$

The general closed-loop transfer function model **can be applied to** any specific system by substituting the transfer function models for the loop elements.

2.3 Controller

1) P controller/proportional mode: proportional only controller

The output from a P controller is proportional to the error and is equal

$$u_p(t) = P \cdot e(t), \quad (2-7)$$

where are P ... **proportional gain**, $e(t) = w(t) - y(t)$... error.

The Transfer function of P controller is

$$R_p(s) = \frac{U_p(s)}{E(s)} = P. \quad (2-8)$$

The proportional mode is simple, provides an adjustment of the **manipulated variable**, does not provide zero offset although it reduces the error, speeds the dynamic response, and can cause instability if tuned improperly.

2) PI controller: proportional-integral controller

A PI controller has two terms (and two tuning parameters), one proportional to the error and the other proportional to the integral of the error. The PI controller output is equal

$$u_{PI}(t) = P \cdot e(t) + I \cdot \int_0^t e(\tau) d\tau = K_R \left[e(t) + \frac{1}{T_I} \cdot \int_0^t e(\tau) d\tau \right], \quad (2-9)$$

where are P ... proportional gain, I ... **integral gain**, K_R ... all controller gain,
 T_I ... **integral time**, $e(t) = w(t) - y(t)$... error.

The PI controller transfer function of a industrial controller is equal

$$R_{PI}(s) = \frac{U_{PI}(s)}{E(s)} = P + \frac{I}{s} = \frac{P_s + I}{s} = K_R \frac{T_I s + 1}{T_I s} = K_R \left(1 + \frac{1}{T_I s} \right). \quad (2-10)$$

Integral mode:

$$u_I(t) = \frac{K_R}{T_I} \cdot \int_0^t e(\tau) d\tau + e(0), \quad (2-11)$$

where $e(0)$... initial condition.

2.4 Root-Locus Design Method

2.4.1 Structure of controller by using the Root-Locus Design Method for tuning

In the synthesis, the pole and zeros of a transfer function $R(s)$ of the controller define the structure of the controller.

1) PI controller

$$R(s) = \frac{r_0 s + r_1}{s} = \frac{r_0 \cdot \left(s + \frac{r_1}{r_0} \right)}{s} = \frac{r_0 \cdot (s - s_{RB1})}{s} =$$

$$= \text{Gain} \frac{(1 + sT_{B1})}{s} = \text{Gain} \cdot R_0(s),$$
(2-12)

$$\text{where } T_{B1} = -1/s_{RB1}$$

2) PID controller

$$R(s) = \frac{U(s)}{E(s)} = P + \frac{I}{s} + Ds = \frac{D \cdot s^2 + P \cdot s + I}{s}$$

$$R(s) = D \frac{(s - s_{BR1})(s - s_{BR2})}{s} = \text{Gain} \cdot R_0(s)$$
(2-13)

2.4.2 Basic properties of Root Loci

The block diagrams of open-loop transfer function and closed-loop are shown in figure 2-9a, b.

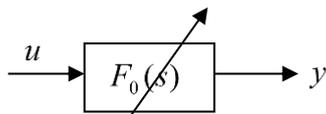


Fig. 2-9a: Open loop diagram

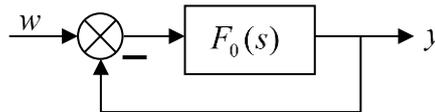


Fig. 2-9b: Closed loop diagram

The closed-loop transfer function is

$$F_{GE}(s) = \frac{F_0(s)}{1 + F_0(s)} = \frac{\text{Gain} \cdot R_0(s)F(s)}{1 + \text{Gain}R_0(s)F(s)} = \frac{\text{Gain} * Z(s)}{P(s) + \text{Gain} * Z(s)}.$$
(2-14)

The characteristic equation of the closed-loop transfer function is

$$1 + \text{Gain} \cdot R_0(s)F(s) = 1 + \text{Gain} \frac{B_0(s)}{A_0(s)} = 0,$$
(2-15)

The equation (2-15) can be rewritten in the form

$$1 + \text{Gain} * \frac{B_0(s)}{A_0(s)} = 0 \rightarrow \frac{B_0(s)}{A_0(s)} = -\frac{1}{\text{Gain}} \rightarrow \frac{\prod_{j=1}^{mQ} |s - s_{Qj}|}{\prod_{k=1}^{nM} |s - s_{Mk}|} \quad (2-16)$$

Condition to Magnitude

$$\left| \frac{B_0(s)}{A_0(s)} \right| = \frac{\prod_{j=1}^{mB_0} |s - s_{Bi}|}{\prod_{k=1}^{nA_0} |s - s_k|} = \frac{1}{|\text{Gain}|}; \quad 0 \leq \text{Gain} < \infty. \quad (2-17)$$

Condition to Angles

$$\varphi = \arg \left\{ \frac{\prod_{j=1}^{mB_0} (s - s_{Bi})}{\prod_{k=1}^{nA_0} (s - s_k)} \right\} = (2i + 1)\pi, \quad i = \pm 0, \pm 1, \dots, \text{Gain} \geq 0. \quad (2-18)$$

Root Loci: The trajectories of roots of the characteristic equation (2-16) when Gain is changed in the interval $0 \leq \text{Gain} < \infty$ is called "Root Loci".

2.4.3 Time- domain specifications

Specifications for a control system design often involve certain requirements associated with the time response of the system. The requirements for a step response are expressed in terms of the standard quantities are shown in figure 2-10.

Consider the second order and the transfer function

$$F(s) = \frac{\omega_n}{s^2 + 2\omega_n \xi s + \omega_n^2},$$

The roots are

$$s_{1,2} = \frac{-2\xi\omega_n \pm 2\omega_n \sqrt{\xi^2 - 1}}{2}$$

$$s_{1,2} = -\xi\omega_n \pm i\omega_n \sqrt{1 - \xi^2} = -\sigma \pm i\omega,$$

$$\text{where } \sigma = \xi\omega_n \rightarrow \xi = \frac{\sigma}{\omega_n}, \text{ and } \omega = \omega_d = \omega_n \sqrt{1 - \xi^2}. \quad 0 \leq \xi < 1.$$

Control performance is the ability of a control system to achieve the desired dynamic responses, as indicated by the control performance measures, over an expected range of operating conditions.

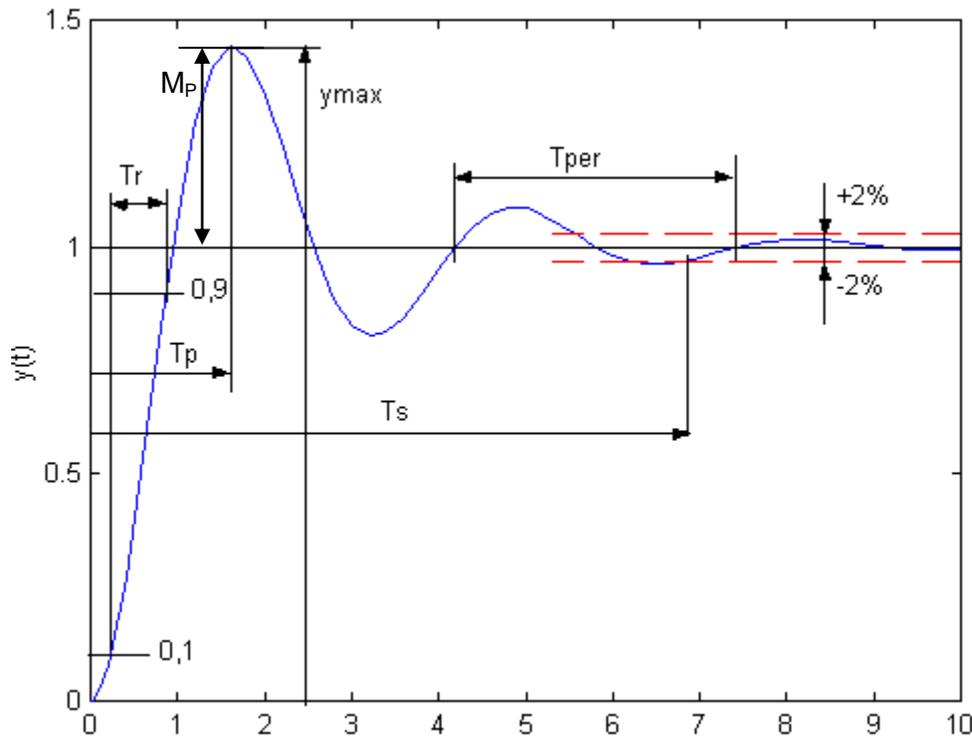


Fig. 2-10: Time domain specifications

- 1) The **rise time** T_R is the time it takes the system to reach the vicinity of its new set point. Approximately it can be calculated from the equation

$$T_R = \frac{1,8}{\omega_n} \rightarrow \omega_n = \frac{1,8}{T_R}. \quad (2-19)$$

The rise time T_R is inversely proportional to the natural frequency ω_n . Since the circle with radius ω_{n1} on the s-plane are locus of constant natural frequency ω_{n1} , it is also the locus for constant rise time T_R (see figure 2-11b).

- 2) The **settling time** T_S is the time it takes the system transients to decay.

$$T_S = \frac{4,6}{\sigma} \rightarrow \sigma = \frac{4,6}{T_S}. \quad (2-20)$$

The settling time is inversely proportional to the real part of the pole. Since vertical lines on the s-plane are lines of constant real value, they are also lines for constant settling time (see figure 2-11b)..

- 3) The **overshoot** M_p is the maximum amount the system overshoots its final value divided by its final value (and is often expressed as a percentage).

$$M_p(\%) = e^{-\pi\xi/\sqrt{1-\xi^2}} \cdot 100\% \rightarrow \xi = \frac{-\ln(M_p(\%)/100)}{\sqrt{\pi^2 + \ln(M_p(\%)/100)^2}} \quad (2-21)$$

We know that $\phi = \sin^{-1} \xi$, radial lines are lines of constant ξ . Since percent overshoot is only a function of ξ , radial lines are thus lines of constant overshoot. This is depicted in figure 2-11b.

- 4) The **peak time** T_P is the time it takes the system to reach the maximum overshoot point

$$T_P = \frac{\pi}{\omega} \rightarrow \omega = \frac{\pi}{T_P} \quad (2-22)$$

The equation (2-22) shows that T_P is inversely proportional to the imaginary part of the pole. Since horizontal lines on the s-plane are lines of the constant imaginary value, they are also lines of constant peak time (see figure 2-11b).

In design synthesis we wish to specify T_r , M_p , and T_s and to ask where the poles need to be so that the actual responses are less than or equal to these specifications.

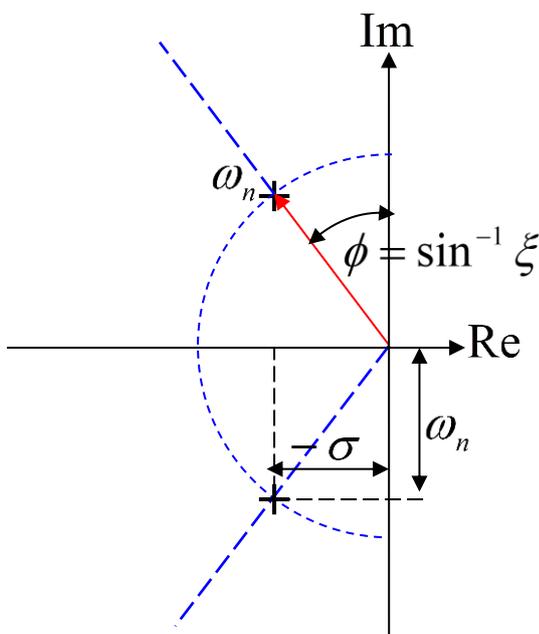


Fig. 2-11a: Pole plot for underdamped second order system

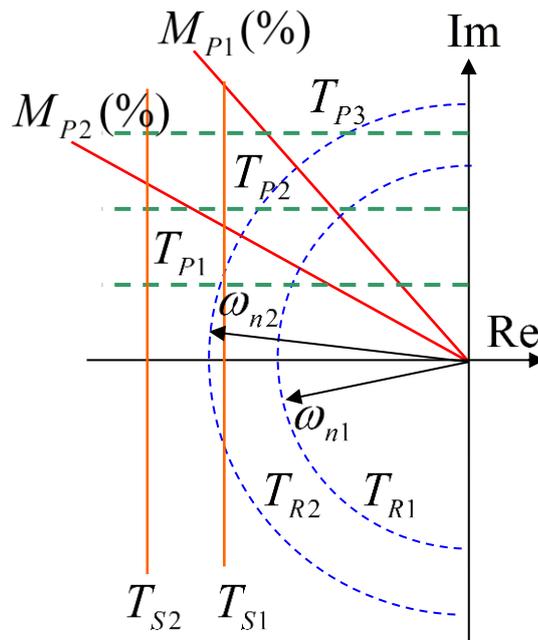


Fig. 2-11b: Line of constant peak time T_P , settling time T_S , rise time T_R , percent overshoot $M_P(\%)$

2.4.4 Program packet “sisotool”

For the Root-Locus Design Method is the program packet MATLAB / SIMULINK “SISO Design Tool” used that is called “**sisotool**”. This Tool is a graphical-user interface (GUI) that allows you to design compensators. The SISO Design Tool is made up of the following:

- 1) The SISO Design Task Node in the Control and Estimation Tools Manager, a user interface (UI) that facilitates the design of compensators for single-input, single-output feedback loops through a series of interactive panes.
- 2) The Graphical Tuning Window - a graphical user interface (GUI) for displaying and manipulating the Bode, root locus, and Nichols plots for the controller currently being designed. This window is titled SISO Design for Design Name.

- 3) The Graphical Tuning Window by default displays the root locus and Bode diagrams for your imported systems. The two are dynamically linked; for example, if you change the gain in the root locus, it immediately affects the Bode diagrams as well.
- 4) The SISO Design Task-associated LTI Viewer (For instructions on how to operate the LTI Viewer, see LTI Viewer)
- 5) A tool that automatically generates compensators using PID, internal model control (IMC), or linear-quadratic-Gaussian (LQG) methods.
- 6) A response optimization tool that automatically tunes the system to satisfy design requirements (available if you have Simulink Response Optimization installed).

2.4.5 Generally rules for controller zeros setting

It is known that the dynamic of a controlled system is predominantly given by its poles. Poles lying nearest to the imaginary axis can cause instability when feedback gain is changing. Therefore we are trying to compensate such poles with the zeros of the controller. It is practically done so:

- 1) The one zero of PI controller or both zeros of PID controller are placed in the vicinity of the poles of the controlled system that are nearest to the imaginary axis.
- 2) If it is necessary to control an oscillating plant, both the zero of the controller are to be situated in the vicinity of the complex poles of the plant that lay nearest to the imaginary axis.
- 3) The gain must be found so that the desired other parameters are satisfied.

The synthesis will be demonstrated on the example.

Example

Approximation of the plant is $F_U(s) = \frac{0.89}{12020 s^2 + 315.5s + 1}$.

Task: Tune PI, PID controllers for given model plant $F_U(s)$ using Root Locus method

Solution:

The controlled plant has poles $s_1 = -0,0223$; $s_2 = -0,0037$. The pole lying nearest to the imaginary axes is the pole s_2 .

1) PI controller

The PI controller has the transfer function:

$$R_0(s) = \frac{r_0 s + r_1}{s} = \text{Gain} \frac{(1 + T_{RB1}s)}{s}, T_{B1} = -1 / s_{RB1}.$$

The zero point s_{BR} of the transfer function $R_0(s)$ should be set in the vicinity of the pole of the pole s_2 .

a) The zero of the controller zero is chosen: $s_{RB} = -0,003$.

The root locus curve and step response are shown in the figure 2-12a, b. Right-click on plots 2.4-4a opens a menu and by clicking on "grid" creates the grid. Right-click on plots 2.4-4b opens a menu and by clicking on "Characteristics" we checked "Peak response", "Settling Time" and "Rise Time".

The desired overall gain of the controller Gain lies in the point of intersection of the straight line for $\xi=0,7$ with the curve of the geometric locus of the roots, see figure 2-11a. The gain in this point is 0,01066. The PI controller is adjusted in MATLAB as follows

$$R(s) = Gain \cdot \frac{1 + T_{BR1}s}{s} = 0,01066 \cdot \frac{1 + 330s}{s} = 3,52 + \frac{0,0106}{s}$$

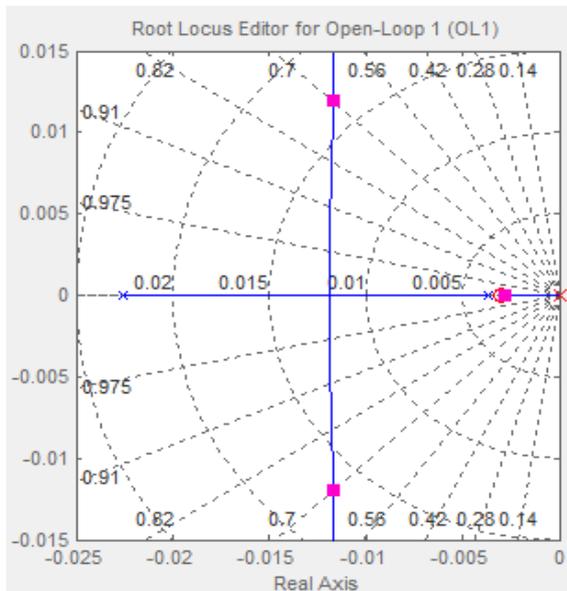


Fig. 2-12a: Roots locus curve of a closed loop with a PI controller

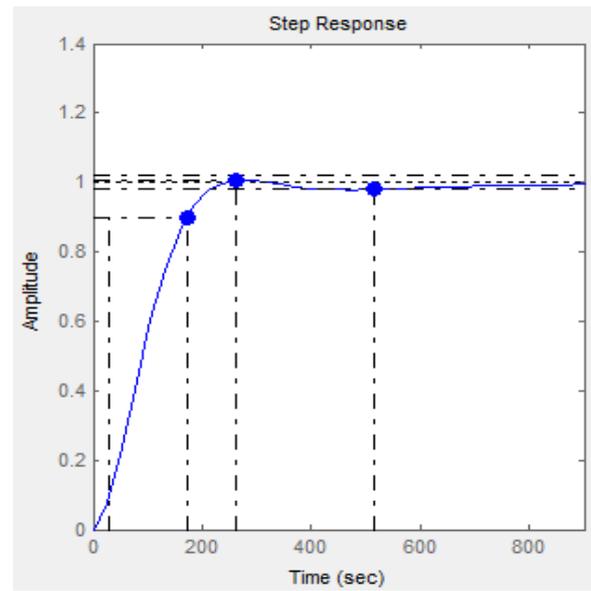


Fig. 2-12b: Step response $w(t)=1$

b) The zero of the controller zero is chosen: $s_{RB} = -0,0045$.

The desired overall gain of the controller Gain lies in the point of intersection of the straight line for $\xi=0,7$ with the curve of the geometric locus of the roots, see figure 2-13a. The amplification - gain was found 0,016. The step response is shown in the figure 2-13b.

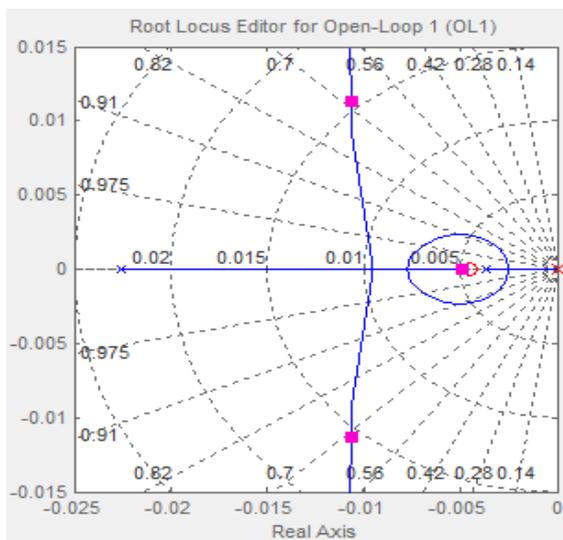


Fig. 2-13a: Roots locus curve of a closed loop with a PI controller

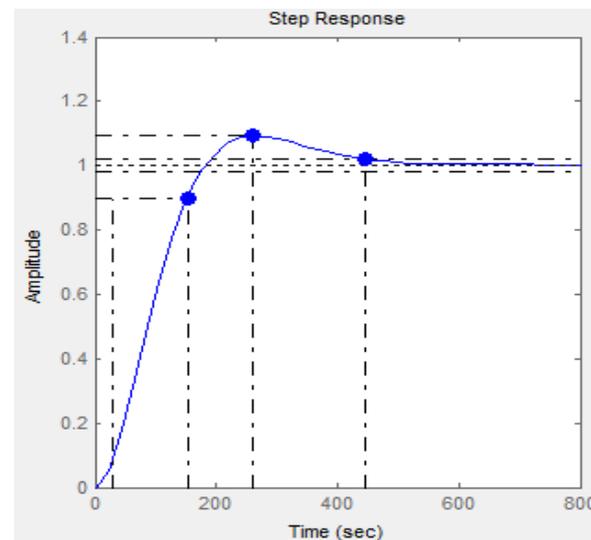


Fig. 2-13b: Step response $w(t)=1$

The PI controller is adjusted in MATLAB as follows

$$R(s) = Gain \cdot \frac{1 + T_{BR1}s}{s} = 0,016 \cdot \frac{1 + 220s}{s} = 3,52 + \frac{0,016}{s}$$

2) PID controller

The PID controller has the transfer function:

$$R(s) = Gain \cdot \frac{(1 + sT_{B1})(1 + sT_{B2})}{s}, T_{B1} = -1/s_{RB1}, T_{B2} = -1/s_{RB2}$$

The zero point s_{RB1} of the transfer function $R(s)$ should be set in the vicinity of the pole of the pole $s_2 = -0,0037$. We set $s_{RB1} = -0,003$.

You can prove if you set the second zero point s_{RB2} in the vicinity of the pole $s_1 = -0,0223$ the found controller will have too big the derivative part. The gain does not help you too. Such a controller cannot be practically used.

After some experiments the second zero of the controller was chosen: $s_{RB2} = -1$ and the gain was set 0.013. The root locus is shown in the figure 2-14a and the step response in the figure 2-14b.

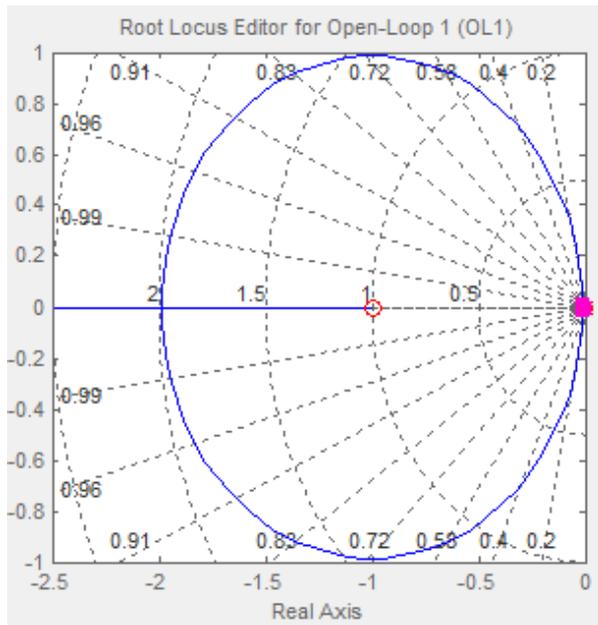


Fig. 2-14a: Roots locus curve of a closed loop with a PID controller

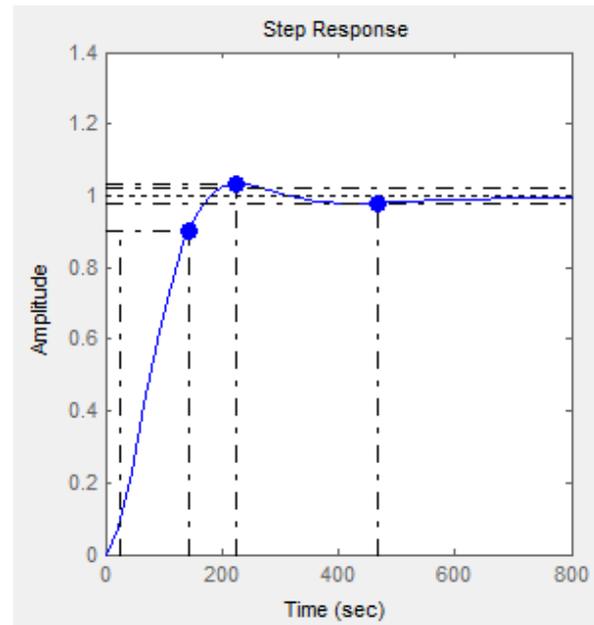


Fig. 2-14b: Step response $w(t)=1$

The PID controller is adjusted in MATLAB as follows

$$R(s) = Gain \cdot \frac{(1 + T_{BR1}s) \cdot (1 + T_{BR2}s)}{s} = 0,013 \cdot \frac{(1 + 330s)(1 + s)}{s} = 4,303 + \frac{0,013}{s} + 4,29s$$

You can compare responses without controller and with PI and PID controller in figure 2-15.

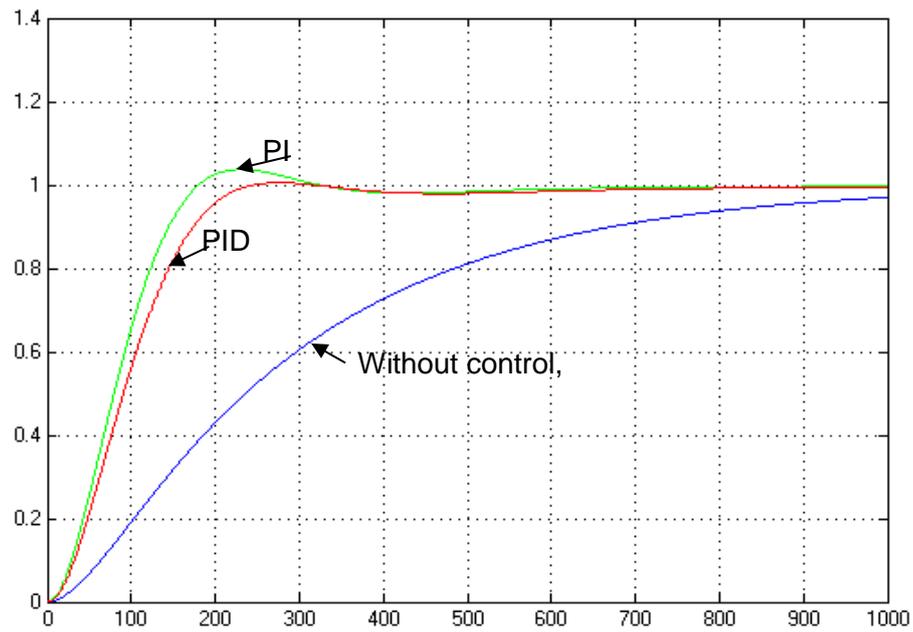


Fig. 2-15: Responses without control, PI and PID

End of Example

3 Set-up of the experiment

3.1 Structure of experiments

A picture of the experiment is shown in the figure 3-1.

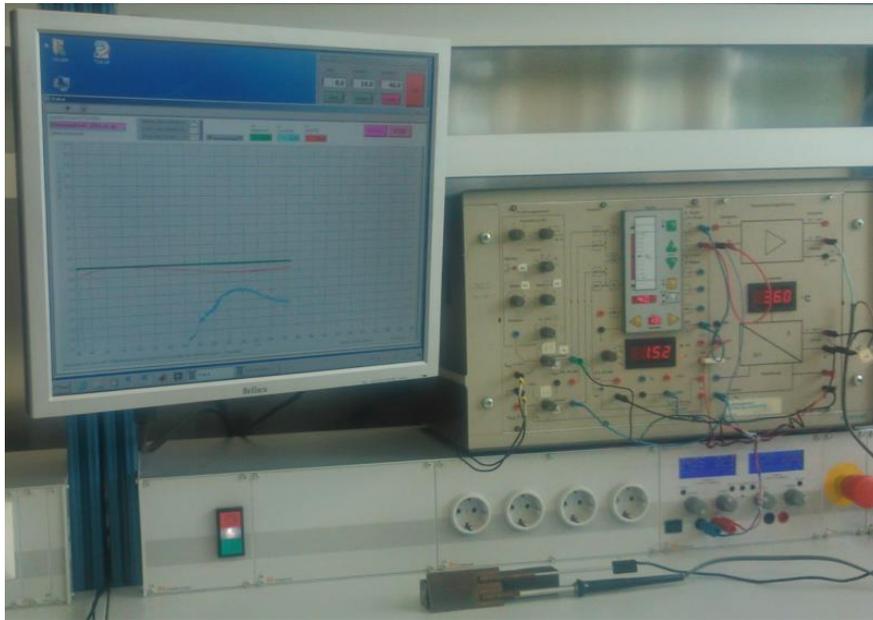


Fig. 3-1: Picture of the experiment

There is used an experimental set up training device SITRAN DR 20 that is shown in the figure 3-2. The essential part of the experimental set-up contains the control device SIPART DR 20 (**controller**) and the **controlled plant** – temperature control of a soldering gun.

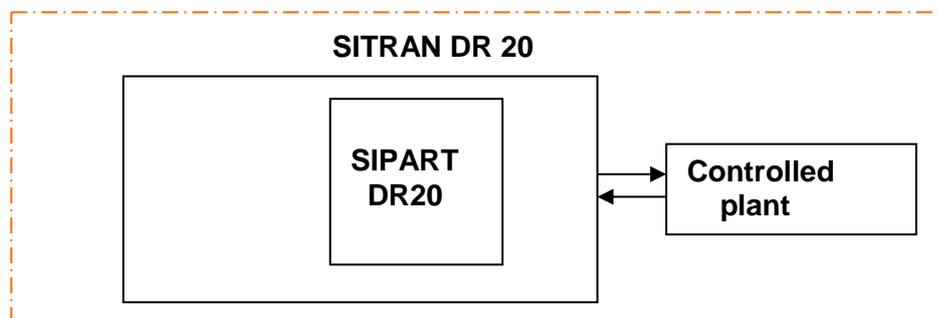


Fig. 3-2: Training device SITRAN DR 20

The measuring and evaluation instruments are digital displays and a PC including input/output card (see figure 3-3). The units will be connected by means of measuring lines and switches respectively. The software for data requisition and saving is LabVIEW, and the software for data computing is MATLAB.

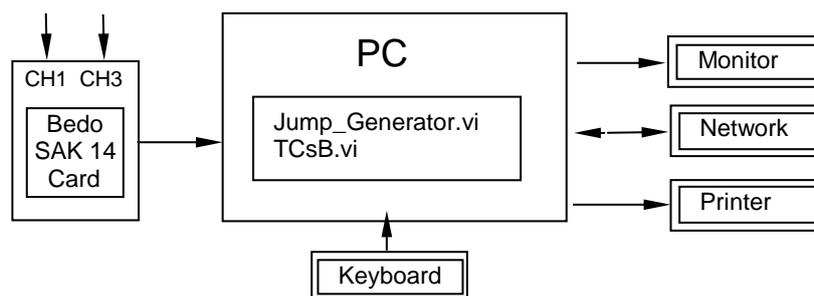


Fig. 3-3: Measuring and control on the experiment with a PC

The functions of the PC during this experiment are the following:

- data requisition and saving
- generating of step functions
- data computing (analyse, modelling, verification, evaluation)

3.2 Controlled plant

The simplified structure of the controlled system is shown in the block diagram figure 3-4. The input and output signals are the voltages $U_y(t)$ and $U_x(t)$. The input signal U_y is amplified by means of a power amplifier (non-linear) because it supplies the heater - the soldering gun with voltage. A sheet metal connects the soldering gun with a temperature sensor (PT100).

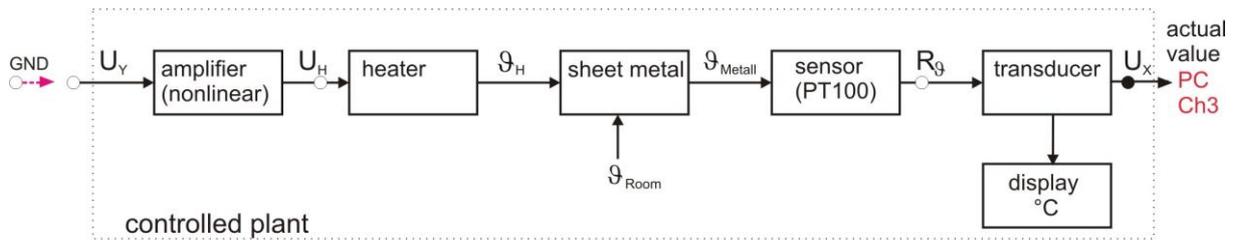


Fig. 3-4: Simplified structure diagram of controlled plant

A transducer transforms the output signal of the temperature sensor (the resistor $R_9(t)$) into the output signal of the controlled system (the voltages U_x). The disturbance is caused by the changing of the room temperature $\vartheta_{Room} = \vartheta_{Room}(t)$. The simplified structure of the feedback control with instrumentation is shown in figure 3-5. The voltages $U_y(t)$ (input signal) can be generated by the block **Jump_Generator.vi** or by the controller **SIPART DR20** by proper connection of inputs with the signal U_w or U_y by the red cable.

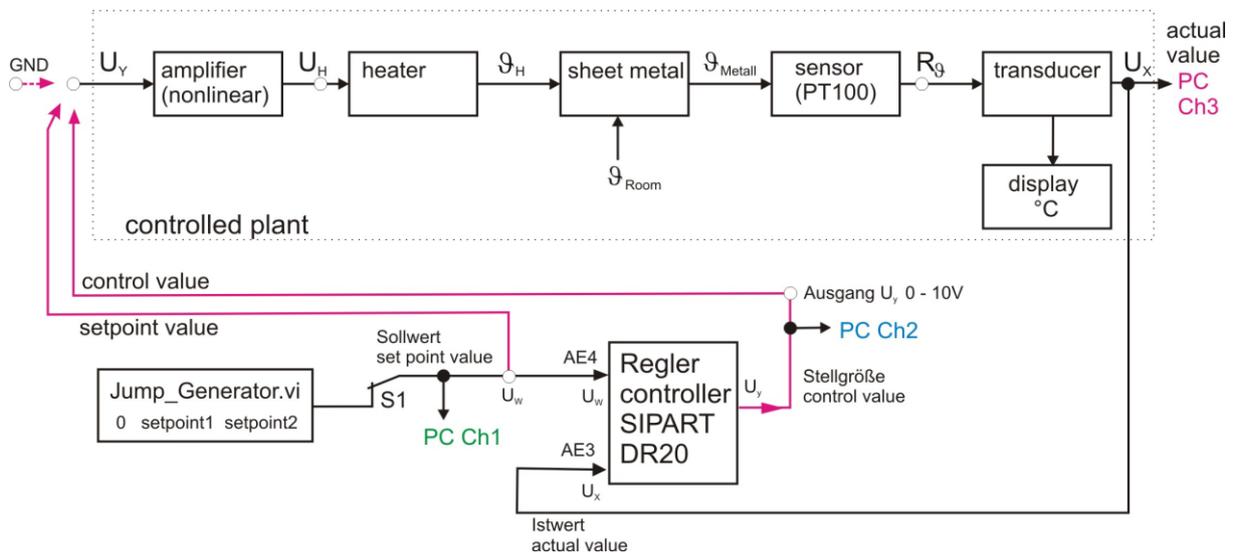


Fig. 3-5: Simplified instrumentation and control diagram of temperature control

3.3 Controller SIPART DR 20

This industrial control device is based on a microprocessor. By means of programming of this device it is possible, to create a controller of different structures. The front of SIPART DR 20 is shown in figure 3-6.

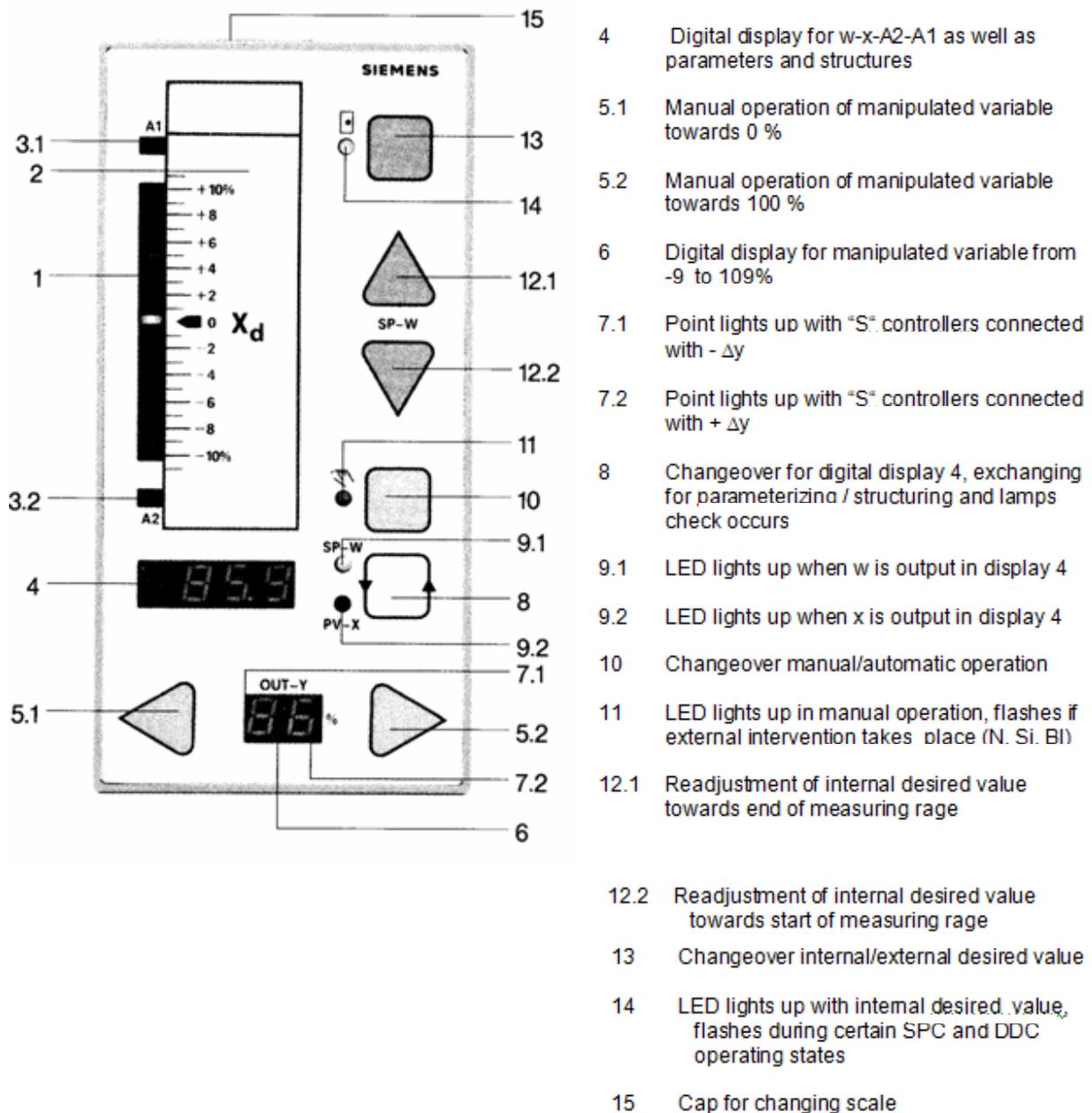


Fig. 3-6: Front view of SIPART DR 20 controller ([1] p.10)

3.4 Data acquisition and control

LabVIEW-Program „Jump_Generator.vi“ (figure 3-7)

The program generates different setpoints.

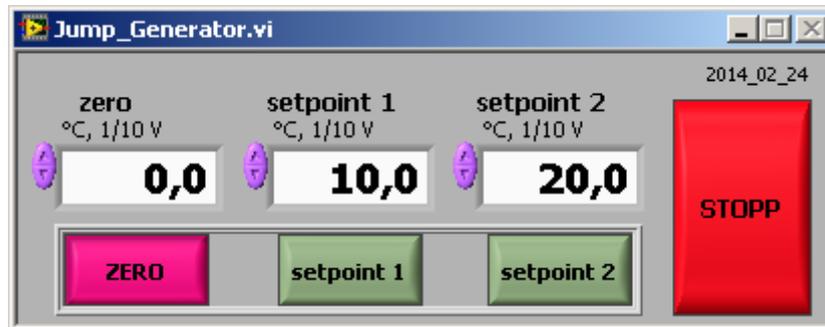


Fig.3-7: Panel of „Jump_Generator.vi“

Data Acquisition and Saving:

The measured data on the inputs CH1 to CH3 of the Bedo SAK 14 card are read in the PC with the program „TCsB.vi“. The input voltage is within the parameters of -10 V and $+10\text{ V}$. This panel is shown in figure 3-8.

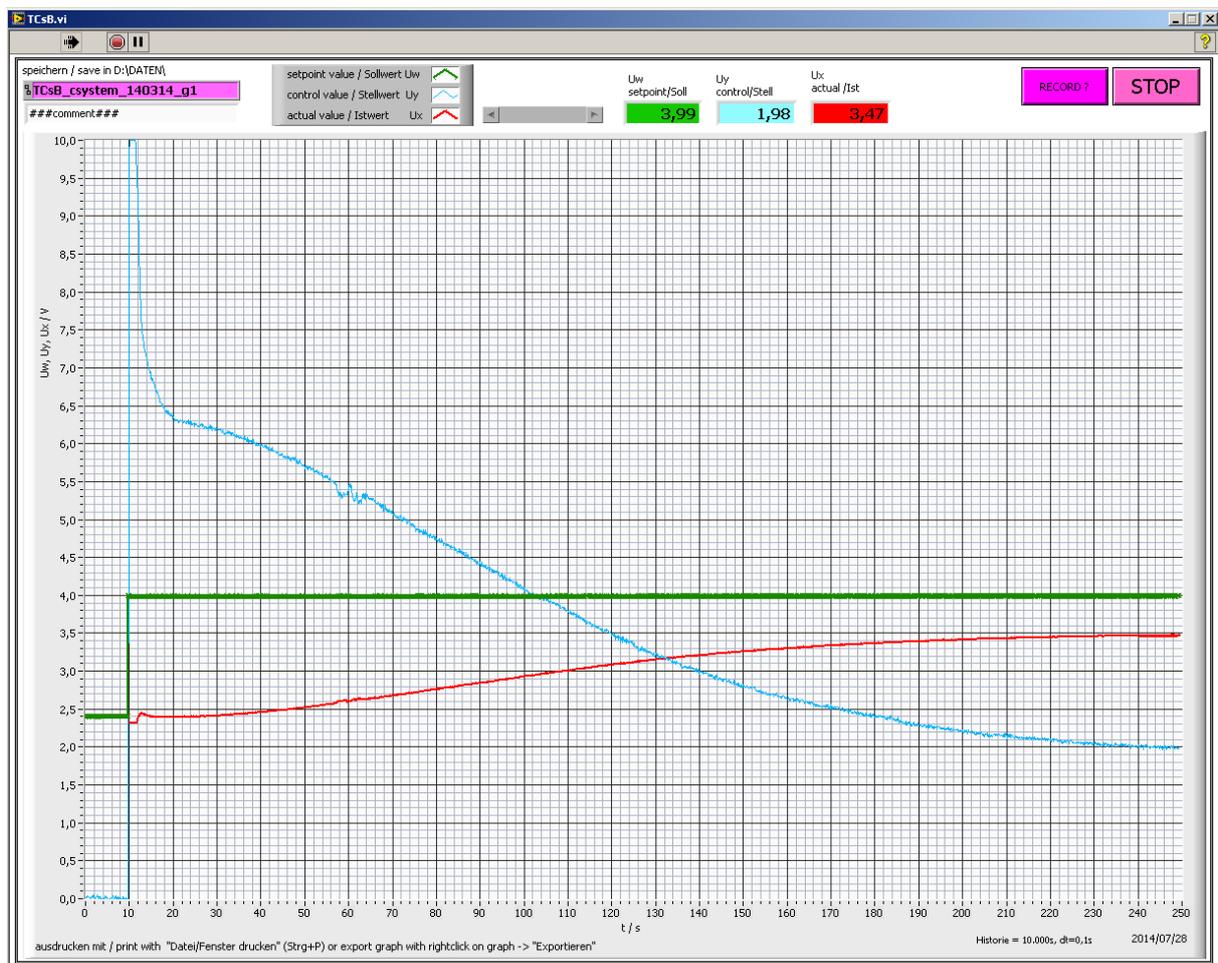


Fig.3-8: Panel of “TCsB.vi

4 Operating the equipment and software used in the experiment

4.1 Controller SIPART DR20

The location of the various buttons and displays is shown in figure 3-5. The digital display 4 shows the reference input variable w and the controlled variable x , respectively, in the unit of measurement (10^{-1} V). The digital display 6 shows the manipulated variable y in the unit of measurement (10^{-1} V).

The SIPART DR 20 has 3 operating levels: **process operating, parameterisation and structuring**. The operation of the device in process operation is, to a large extent, self-explanatory.

In the **parameterisation level** the individual parameters for adaptation of the device to the system data are called up and set in on-line operation, i.e. the controller remains in the set mode.

In the **structuring level** a large number of prepared functions for controlling industrial systems are filed in the program memory. Using adjustable switches, the diverse structures of the controller are composed. Table 4-1 shows the essential structuring switches with their adjustable values.

Structuring Switches	Adjustable values	Remarks
1	4	Slave controller without internal/external switchover
8	1	Allocation of input x1
10	1	Allocation of input x2
25	-2; (0; 1)	Display range of the display 1 (2.5%; 10%; 20 %)
26	0	Sense of the controller on the controlled system ($K_p > 0$) Referred to $X_d = w - x$
27	0	D element input : X_d
28	0 1	PI (D) – controller P (D) - controller
36	1	Components supplied with the optional module GW

Table 4-1: The essential structuring switches with their adjustable values

For security reasons, the changeover from process operation to parameterization or structuring takes place in several steps. These steps must be carried out within 20 seconds, otherwise there is an automatic return to process operation. The operation of the controller takes place in the following steps ([1], p.:17).

Preparation for changeover to parameterization and structuring: Press key 8 until the „PS“ appears on the two-digit display. Release key 8. The display „PS“ changes to steady light.

4.1.1 Parameterization

Changeover to parameterization level:

When „PS“ lights up, press key 12.1 several times until the letters „**PAR**“ appear in the four-digit display 4; „**Str**“ and then „**PAR**“ again appear if „**PAR**“ was bypassed by mistake. Press key 5.1 or 5.2 once; the last parameter selected appears, the name in display 6 and the appertaining value in display 4. Display 1 still shows the negative deviation for the observation of the process.

Selecting the parameters:

The parameter list (table 4-2) is paged forwards and backwards using the 5.1 and 5.2 keys. The essential characters of the parameterization with the adjustable values are shown in table 4-2.

Setting the parameters:

The parameter which is currently selected can be set using the keys 12.1 and 12.2.

Switching back to process operation level:

During setting the device can be switched back to process operation at any time using key 8.

Character	Adjustable value	Parameter
Vv	1	Derivative-action gain
Cp	? -	Proportional gain
Yo	0	Working point
Tn	? / s	Reset time
Tv	? / s	Derivative-action time
TF	off	Filter time constant for xd
tS	off	Setpoint ramp
ty	60	Valve positioning time/ Period y+
t -	60	Period y-
LA	0	Start of scale value 0 %
L1	0	Vertex
.	.	.
.	.	.
.	.	.
L7	0	Vertex
LE	100	Full-scale value 100
SA	-5	Setpoint limitation start
SE	105	Setpoint limitation end
SH	0.0	Safety point
A2	-199.9	Limit value, minimum
A1	999.9	Limit value, maximum
A	0.0	Response threshold of xd
YA	-5	Manipulated variable limitation, start of ya
YE	105.0	Manipulated variable limitation, end of ya
yS	0.0	Safety manipulated variable
c1	0.0	Constant 1 (zero)
c2	0.0	Constant 2 (factor)

Table 4-2: The parameterization with the adjustable values

4.1.2 Structuring**Changeover to the structuring level:**

When „PS“ lights up, press key 12.1 several times until the letters „Str“ appear after „PAr“ in the four-digit display 4. Press key 5.1 or 5.2 once: the last selected structuring switch appears; the number in display 6 and the switch position in display 4. The controller goes into manual operation mode (LED 11 is on), the output OUT-Y (manipulated variable) can be changed with the buttons 5.1 and 5.2. Display 1 now displays a striped pattern.

Selecting structuring switches:

The structuring switch list (table 4-1) is paged forwards or backwards using keys 5.1 and 5.2

Setting the structuring switches:

The structuring switch which is currently active can be set using the keys 5.1 and 5.2.

Switching back to process operation or parameterisation:

During setting, the device can be switched back to process operation at any time by pressing key 8 twice. Because the controller is in manual mode it has to be switched back to automatic mode with button 10. Mind the bumpless switchover (see hints in 5.)!

If you press key 8 only once then "PS" and "PAR" are displayed and you are in parameterisation mode (see above).

4.1.3 Controller output functions of controllers are:

The characteristic functions for the transfer behaviour of the controller are adjusted by the characteristics of tables 4-1 and 4-2.

$$\text{P-Controller} \quad y = C_p \cdot e(t) \quad (4-1)$$

e : error variable
y : manipulated variable

$$\text{PI-Controller} \quad y = C_p \left[e(t) + \frac{1}{T_n} \int_0^t e(\tau) d\tau \right] \quad (4-2)$$

C_p, T_n : see table 4-2

4.2 Data acquisition and control

LabVIEW-Program „Jump_Generator.vi“ (figure 3-6)

You can put in the values in the corresponding field and jump step between the different values by clicking on the buttons „Zero“, „setpoint1“ or „setpoint2“.

The measured data on the inputs CH1 to CH3 of the Bedo SAK 14 card are read in the PC with the program „TCsB.vi“. The input voltage is within the parameters of –10 V and + 10 V.

This desktop is shown in figure 3-7.

The buttons under the menu bar are self-explanatory. Place the cursor over the button to read a description field.

Measured and shown are the charts of the setpoint U_w (blue), the actual value U_x (red) and the control value U_y (black). You can adjust the scaling of the chart by entering of the new final values on the co-ordinate axis.

In the violet field "speichern / save in D:\Daten\" you enter the filename of the data which will be measured and saved after the start of recording.

The form of the file name for this experiment is: TCsB_u_v_w with

u (character of analysed part)

analysis of controlled system : u = "csystem"

analysis of controlled control loop: u = "cloop"

v (character of date) : year month day (for instance : "150515")

w (character of group) : g + number of group (for instance: "g2")

- You can read in the recorded data with MATLAB, Excel or Open Office.
- The datafile has the following format:
 1. Headerline with % name, date, time
 2. Headerline with the comment
 3. Headerline with the name of the measured signals
 - ...Datalines with tab separated ascii data values
- The data values are in stored in mV and ms without decimal separator to avoid problems with the decimal separator format!
- If you append the data to an existing file you get additional header lines!

Start of measurement: click on the button "Run" 

- the violet fields "STOP" and "RECORD ?" can be seen
- you can see the charts of the measured data

Start of recording: click on the button "RECORD ?"

- the field "RECORD ?" changes to a yellow field "Recording!"

End of measurement: click on the button "STOP"

Data computing

The handling of MATLAB is the content of the curriculum and is not described here [2], [3].

Printing of windows and exporting of graphs in LabVIEW (figure 4-1)

In LabVIEW you can print out the window with „Strg+P“ (if the program is not running) or export the graph with rightclick on the graph -> "Exportieren" (export) -> "Vereinfachtes Bild exportieren" (export simple picture) -> "Bitmap.bmp" -> "In Datei speichern" (save in file) or "In Zwischenablage exportieren" (export to clipboard) -> "Speichern" (save).

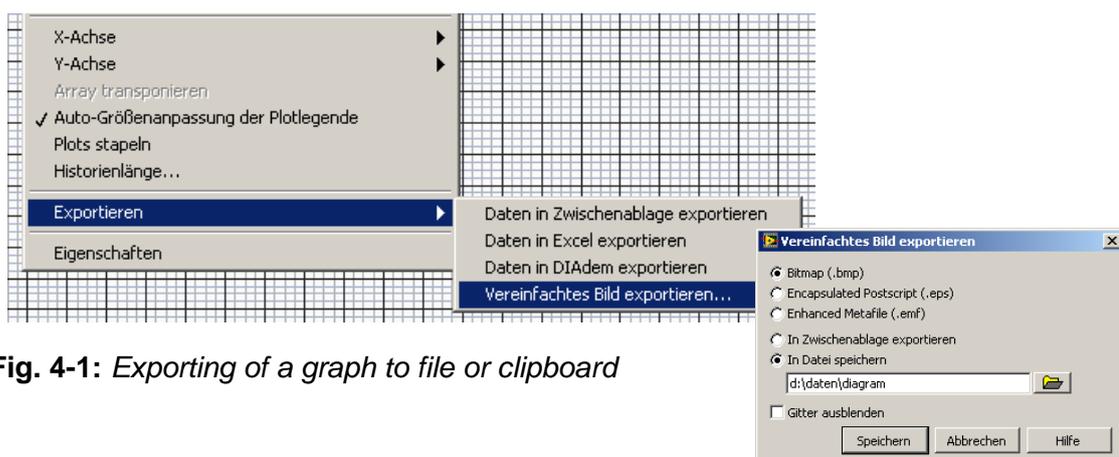


Fig. 4-1: Exporting of a graph to file or clipboard

5 Conducting the experiment

5.1 Task of the experiment

- 1) Make the acquaintance of the experiment and instruments, measuring and data acquisition and limitations.
- 2) Create the mathematical model by identification, plot your measurements and model structure, compare the model and plant outputs.
For evaluation of the parameters you can use:
 - a) Modification of MATLAB programs from lecture
 - b) The MATLAB System Identification Tools
- 3) Design the PI controller:
 - a) Manually
 - b) Using the Root Locus method (MATLAB SISO Design “sisotool”).
- 4) Simulate the feedback control and compare the responses without and with PI controller.
- 5) Check the controller by real control.

5.2 Conducting of measuring and recording on the SITRAIN DR 20

The input steps to the controlled plant can be changed with the “**Jump_Generator.vi**” (see figure.3-6).

The step response, the input signal of the controlled system and the set-point signal can be recorded by the software “**TCsB.vi**” that is shown in the figure 3-7.

5.3 Instruction to the identification measurement

The static characteristic shows us if linearization is possible by approximation of the non linear dynamic by a transfer function. We shall find a linear part of the static characteristic where the working point can be placed. You are using the measured static characteristic in the figure 5-1 because the dynamic of this experiment is very slowly and the measurements would take a long time. The static characteristic depends on the room temperature ϑ_{Room} , so that the measurement of the static characteristic was carried out for two room temperatures: $\vartheta_{Room} = 23^0 \text{ and } 28^0 C$. The temperature $\vartheta(\infty)$ was measured on the output as the voltage U_x by the scale $1V = 10^0 C$.

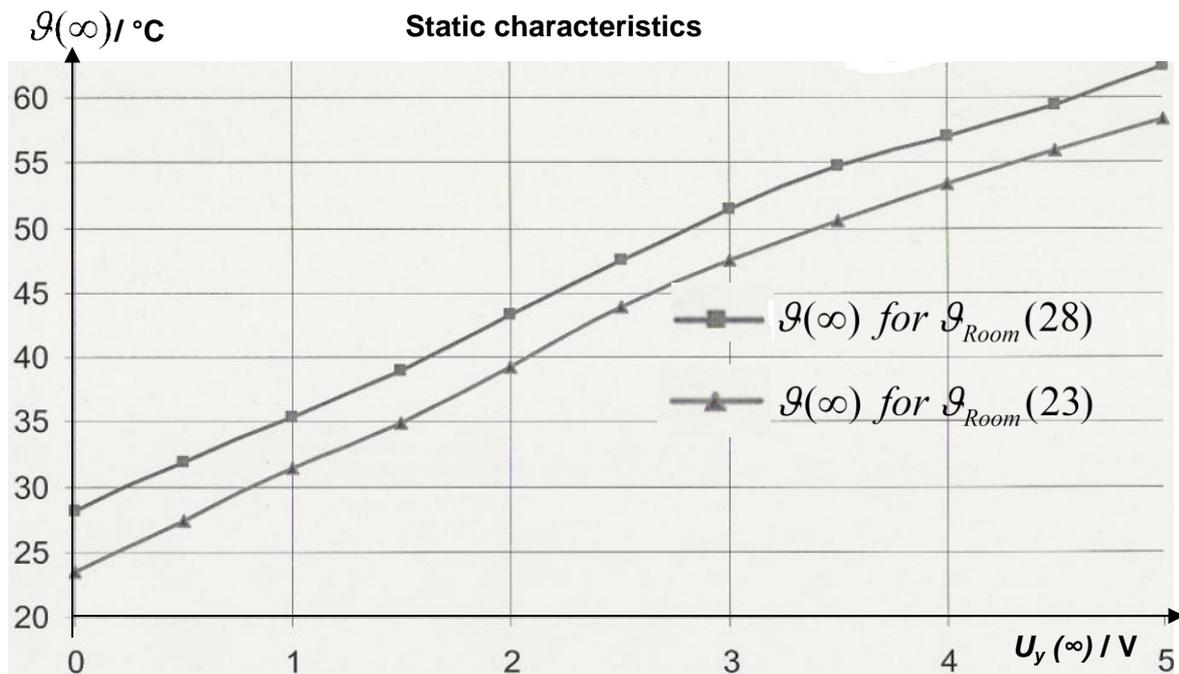


Fig. 5-1: Measured static characteristic

An identification measurement is shown in figure 5-2.

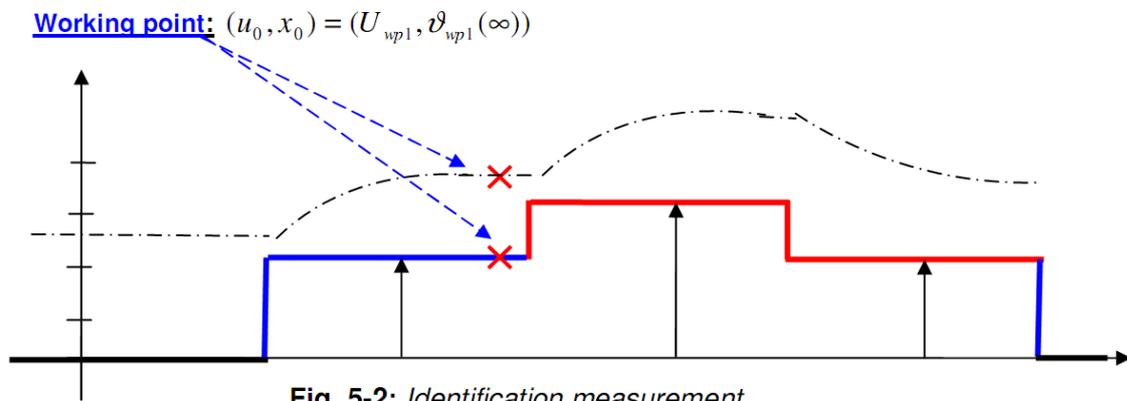


Fig. 5-2: Identification measurement

The identification measurement procedure:

1. Determine the working point $(u_0, x_0) = (U_{wp}, \vartheta_{wp}(\infty))$ on the static characteristic and voltage U_{ID} .
2. Connect the input of the plant U_y with GND (red cable) to prevent the plant from heating up (see figure 5-3).

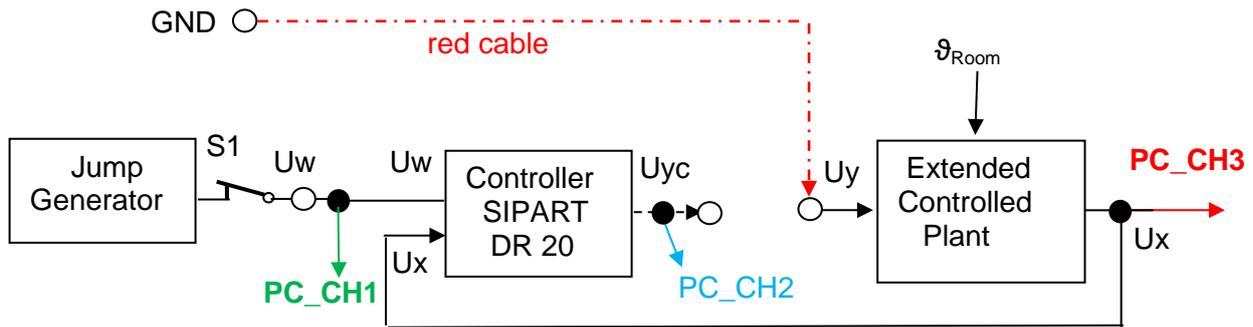


Fig. 5-3: Block diagram for preventing from heating

3. Cool down the plant to room temperature.
4. Put in the "setpoints" in „Jump_Generator.vi“ and run the program.
5. Put in the filename (and comment) in „TCsB.vi“.
6. The "setpoint" can be monitored in the SIPART Display 4 (if activated by switch 8 -> LED 9.1 on).
7. Connect the input of the plant U_y (red cable) with the "set point"- output U_w (Output after switch S1, see the figure 5-4).

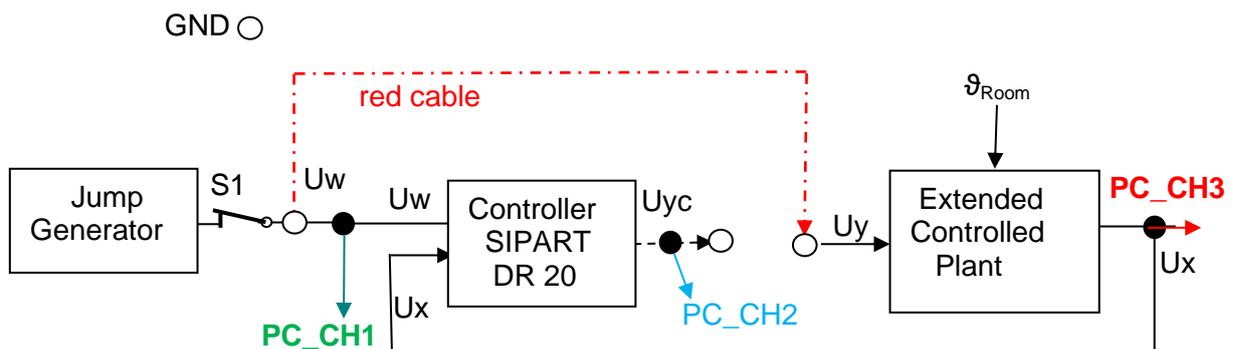


Fig. 5-4: Cable connection for identification measurement

8. Run the program „TCsB.vi“ and activate recording by a click to "record".
9. The input voltage is made by the "**Jump_Generator.vi**". Carry out:
 - a) Step from the "setpoint ZERO" to the "setpoint 1" (U_{WP} on the „Jump_Generator.vi“). The "setpoint 1" is the voltage of the working point U_{WP} . The steady output is the steady temperature in the working point $\vartheta_{WP}(\infty)$
 - b) Step from the "setpoint 1" (U_{WP}) to the "setpoint 2" (U_{ID}). The steady output U_x is the steady temperature $\vartheta_{ID}(\infty)$
 - c) Step from the "setpoint 2" to the "setpoint 1"
10. Stop „TCsB.vi“.
11. Step to setpoint ZERO.
12. Connect the input of the plant U_y with GND (red cable), to prevent the plant for heating up.

5.4 Modelling

5.4.1 Modelling of the controlled system

Using the MATLAB software to determine the mathematical model of the controlled system from experimental data (with an identification parameter techniques).

1. Give reasons for the structure of the model.
2. Decide what structure of transfer function $F(s)$ shall be used.
3. Use identification input as a step from the working point u_0 to the defined input u_1 . For this input perform the identification measure and calculate the parameters of the transfer function $F(s)$. After parameterization the function is denoted as $F_1(s)$.
4. Use identification input as a step from the working point u_0 to the defined input u_1 and back to the input u_0 . For this input you shall perform the identification measure and calculate the parameters of the transfer function that is denoted as $F_2(s)$.
5. Check the outputs of the both transfer functions $F_1(s)$, $F_2(s)$ and of the real plant if the inputs are the same identification steps! Explain the differences!
6. Verify the model. Describe the procedure used and your results

5.4.2 Theoretical analysis of the control loop

1. Analyse the real control loop, which is composed of the controlled system and a PI-controller. What are the limitations?
2. Define the range of stability of the control loop as function of the parameters of the PI-controller.

5.4.3 Studies on the model of the control loop

1. Using the software package MATLAB to create a model of the control loop
2. Determine the parameters of the PI controller using the model of the control loop for both transfer functions $F_1(s)$, $F_2(s)$ according to the following methods:
 - a) manually tuning
 - b) Root locus method
3. Check the behaviour of the control loop outputs with transfer functions $F_1(s)$, $F_2(s)$
4. Illustrate and check this modelling control results graphically.

5.5 Studies on the real feedback system (control loop)

1. Create the real control loop on the experiment device.
2. Adjust the parameters on the controller, one after another.
3. The reference inputs (set points) are volts in this feedback system and are generated by the "Jump_Generator.vi".

In this experiment the relation between the temperature ϑ / °C and the voltage U / V is

$$\vartheta = k_{\vartheta} * U_{\vartheta} \quad \text{with} \quad k_{\vartheta} = 10^{\circ}\text{C} \cdot \text{V}^{-1} . \quad (5-1)$$

The set point temperature ϑ to the feedback system is given in volts.

The volts must be calculated by the equation

$$U_{\vartheta} = \frac{\vartheta}{k_{\vartheta}} \quad (5-2)$$

4. We give as the set point first the steady temperature of the working point $\vartheta_{WP}(\infty)$ and after the output temperature of the closed loop system will reach its steady state we give as the next set point the steady temperature of the sheet metal $\vartheta_{metal}(\infty)$.
5. Check the disturbance rejection.
6. Instruction for measurement:
 - Cool down the plant to room temperature.
 - Put in the setpoints in „Jump_Generator.vi“ and run the program with Setpoint ZERO
 - Put in the filename (and comment) in „TCsB.vi“.
 - Program the controller with the calculated parameters.
 - Switch to automatic operation mode (SIPART switch 10) after going back to process operation from structuring level and mind the hints at the end of the chapter.
 - Close the control loop by connecting the input of the plant U_y (red cable) with the controller output U_y (“ U_y 0 – 10 V”, see the figure 5-5)

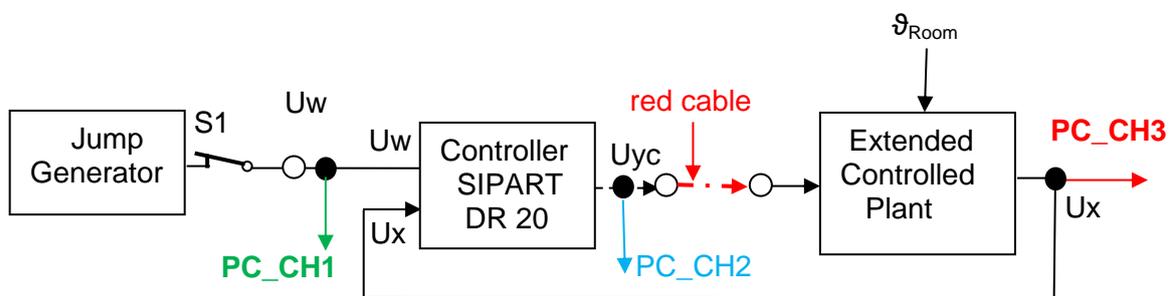


Fig. 5-5: Cable connection for feedback control

Run the program „TCsB.vi“ and activate recording.

1. Step from **setpoint ZERO** to **setpoint 1** - $U_{gWP}(\infty)$ on „Jump_Generator.vi“.
2. Step from **setpoint 1** ($U_{gWP}(\infty)$) to **setpoint 2** ($U_{gD}(\infty)$)
3. Disturb the plant by dunking the sheet metal in water briefly **after steady state** has been reached
4. Step from **setpoint 2** ($U_{gD}(\infty)$) to **setpoint 1** ($U_{gWP}(\infty)$)
5. Disturb the plant by dunking the sheet metal in water briefly **after steady state** has been reached
6. Stop „TCsB.vi“.
7. Step to set point ZERO.
8. Connect the input of the plant U_y with GND (red cable), to prevent the plant from heating up.

Hints!

Let the system go into steady state before you jump to another set point to get comparable results.

With activated integral part of the controller (switch S28=0) the controller ensures a bumpless switchover. When you change from manual to automatic mode the controller generates therefore an offset y_0 to the output OUT-Y (manipulated variable) if there would be a jump at the output otherwise. This happens when you leave the structuring level (4.1.2) or when you switch on the device!

The compensation of the offset y_0 can take a very long time, especially with high values for the integral part (reset time T_n) and therefore lead to a seemingly absurd control behaviour!

To avoid this behaviour you have to set the control deviation/error to zero with a due setpoint (setpoint = actual value) and then set the output value OUT_Y also to zero (button 5.1 and 5.2). The offset y_0 will therefore be zero when you switch to automatic mode with button 10.

This handling is also advisable after switching on the device (the controller switches internally from manual to automatic) to have a defined value zero for the offset y_0 .

6 Analysis of the experiment

1. Discuss and evaluate the behaviour of the feedback system that is working with PI controller, which parameters has been calculated by using transfer functions $F_1(s), F_2(s)$!
2. Discuss and evaluate all step responses and disturbances!
3. Compare the step responses of the real control loop to the model of the controlled system with the transfer functions $F_1(s), F_2(s)$ as a plant model!
4. What are your conclusions?

7 Literature

- [1] SIEMENS: SIPART DR 20 S/K Operating Instructions
Ord. No.: C73000-M7464-C30-8
- [2] The Student Edition of MATLAB. Version 4, User's Guide. The Math Works, Inc.1995, Prentice Hall, Englewood Cliffs. ISBN 0-13-184979-4
- [3] Simulink Dynamic System Simulation for MATLAB. Using Simulink, Version 2, The Math Works, Inc. 1997