

# PID CLOSED-LOOP CONTROL ANALYSIS FOR AUTOMATION WITH SIEMENS PLC USING TIA V13

*Yin Yin Soe<sup>1</sup>, Pann Ei San<sup>2</sup>*

*Department of Electronic Engineering, Technological University (Thanlyin), Yangon (+95), Republic of the Union of Myanmar*

## Abstract

***This work is proposed about PID closed-loop control analysis for Siemens PLC using Totally Integrated Automation (TIA) Portal V13 how to implement PLC analog input projects. In industrial automation technology, controllers are used in many different ways, for example for temperature control in processes. Influencing technical variables in systems requires controlling these variables. so, PID control system for analog input process to achieve high performance and better accuracy. In the SIMATIC (Siemens PLC) world, the "PID Compact" block, version 2.2, is provided for the S7-1500 CPUs for closed-loop process control. Siemens PLC provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks. In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions only form one element of such a concept. The aim of this analysis is especially to get knowledge of practical teaching with Siemens PLC for BE (Electronics) students.***

***Keyword: Siemens PLC, TIA portal V13, PID-Compact, S7-1500 CPU, closed-loop control***

## 1. INTRODUCTION

In industrial automation systems and applications, PLC programming is very essential for many advanced engineering programs: electronics, mechatronics, industrial engineering, process industry, etc. One of the main goals for the courses related to PLC programming consist on the learning of adequate control procedures applied to different engineering fields and real

automation plants to be maintained and supervised using these industry-wide standard controller [1].

In automation technology, controllers are used in many different ways, for example for temperature control in processes. Influencing technical variables in systems requires controlling these variables. So, PID closed-loop control system is needed for industrial automation system to get nearest output results and better performance.

## 2. OVERVIEW OF CLOSED-LOOP CONTROL SYSTEM

For an application which control the temperature of a room by switching on an electric fire. The fire will heat the room up to the maximum temperature that is possible bearing in mind the rate at which the fire heats the room and the rate at which it loses heat. This is termed open loop control in that there is no feedback to the fire to modify the rate at which it is heating the room.

To do this, it is needed a thermostat which can be set to switch the fire on when the room temperature is below the required value and switch it off when it goes above it. There is feedback of temperature information in this system and as such it is termed closed loop control. Closed loop control of some variable, e.g. the control of the temperature in a room, is achieved by comparing the actual value for the variable with the desired set value and then giving an output, such as switching on a heater, to reduce the difference. Figure 1 illustrates this by means of a block diagram. The actual value of the variable is compared with the set value and a signal obtained representing the difference or error. A controller then takes this difference signal and gives an output to an actuator to give a response to correct the discrepancy. Such a system is called closed-loop control.

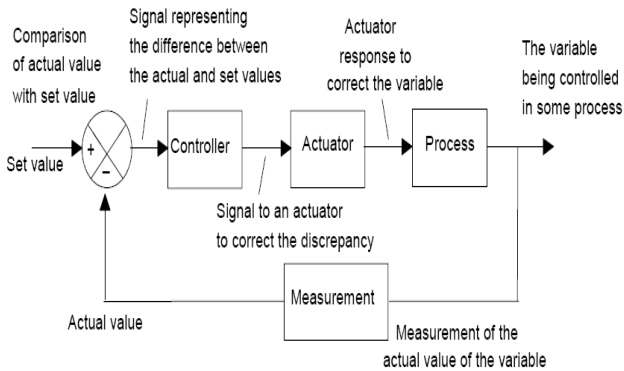


Figure 1 Closed-Loop Control System



Figure 3 USB converter (MPI port)

### 3.REQUIREMENTS OF THE PROPOSED WORK

For proposed work, the required Hardware and Software for this learning are as follows:

- S7-1500 firmware 1.8 or higher
- Software: Totally Integrated Automation (TIA) Portal V13 SP1 Update 9 or higher (Component for programming the S7-1500)
- PC computer unit work as programming media through its serial port through serial to USB converter (MPI port)



Figure 4 Siemens Memory Card

### 4.BASICS OF CONTROL ENGINEERING



Figure 2 S7-1500 Modules Configuration

- Power Supply, PS 25W 24VDC, 6ES7 505-0KA00-0AB0 alternatively, other power supplies (24V DC) can also be used.
- SIMATIC S7-1500, alternatively, other CPUs from the S7-1500 range can also be used.
- PC station, Any PC station with appropriate software can be used. WinCC V13 SP1 Professional
- (TIA Portal V13) (Component for configuring the visualization)

Control engineering is a branch of engineering that researches how to specifically influence given variables in technical systems. The aim is to achieve and maintain the desired value of this variable under certain conditions. A controlled system contains the variable to be controlled, such as the temperature of a room. To identify the type of a system and then dynamically control it in an optimal way, the system to be controlled must be analyzed in detail.

One option to identify the type is to look at the step response of a controlled system. As an example, the following figure shows the step response of a PTn system (for example, temperature in a room).

The time response can be approximately defined by the following variables: Delay time  $T_u$ , Compensation time  $T_g$ , and Maximum value  $X_{max}$ .

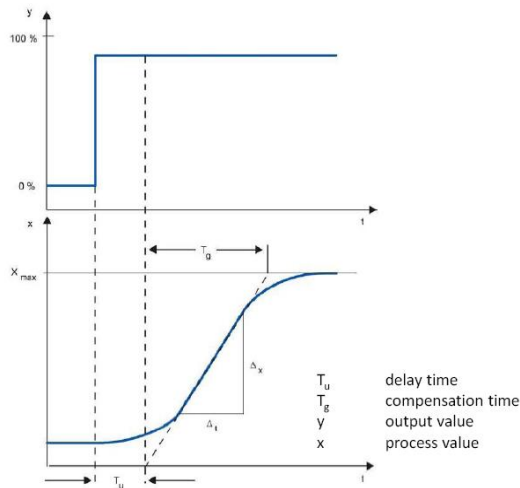


Figure 5 PTn system step response

The controller controls an actuator to bring the controlled system to a desired state. The simplest controllers are two-position controllers that only know the states "ON" and "OFF" and use them to control the controlled system via the actuator.

The frequently used PID controllers consist of three parts:

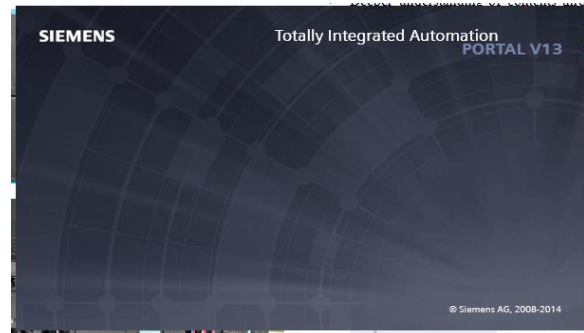
- The P component generates an output signal proportional to the system deviation.
- The I component integrates the system deviation over time and, due to this integration, affects the controlled system.
- The D component, however, responds to the changed system deviation (time derivation of the system deviation).

These three components of the ideal PID controller are weighted by the coefficients proportional gain, integral action time and derivative action time.

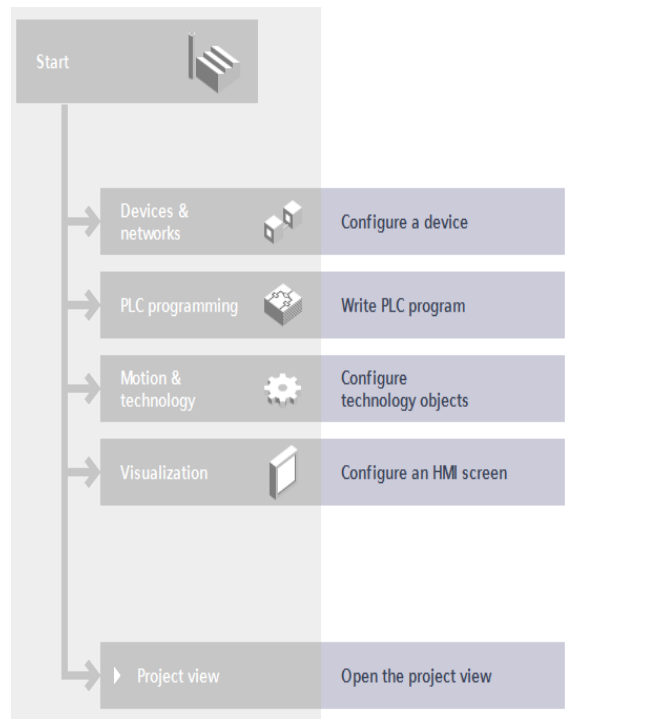
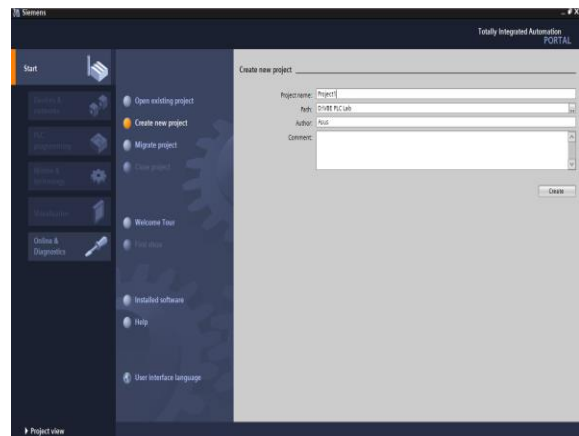
### 5.TIA PORTAL V13 SYSTEM FLOW

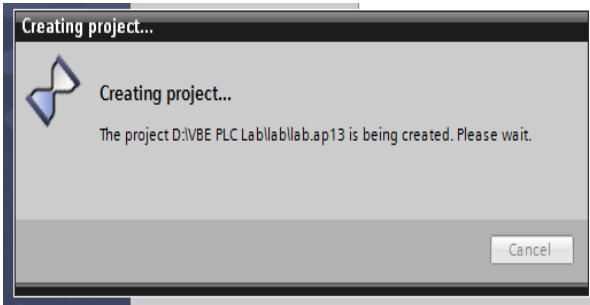
How to create Project and Basic information:

- Explore TIA portal

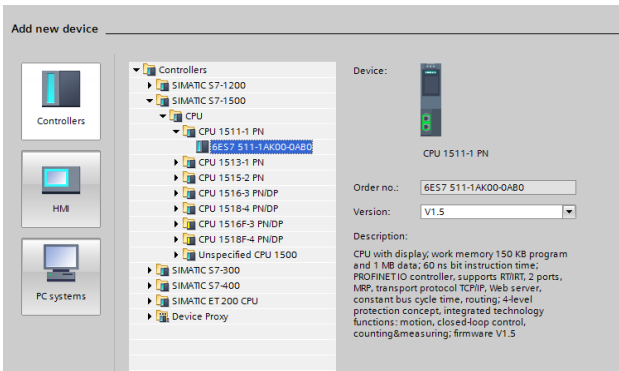


- Create New Project

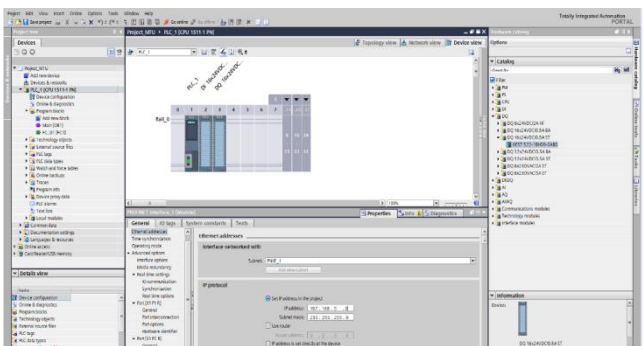




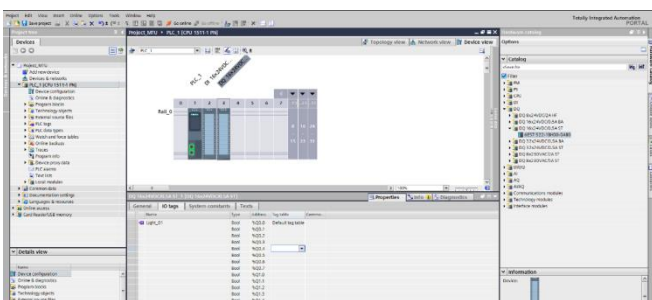
- From "add new device" window select Controller CPU 6ES7 511-1AK00-0AB0 device, to configure.



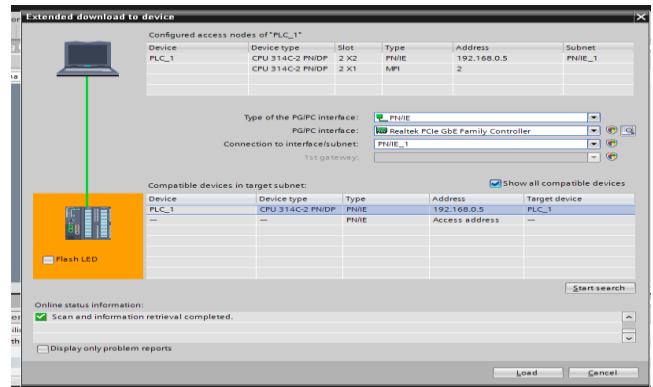
- Configure Ethernet Connection (IP Protocol).



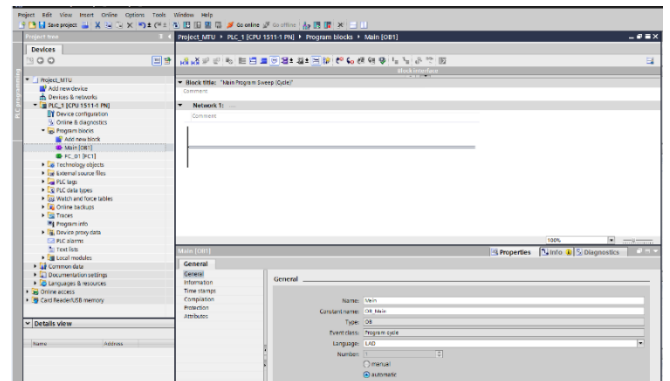
- Configure IO Addresses.



- Configure access nodes of PLC connection. (Download to Device)



There is a Program blocks folder in the Project tree. In this folder, there is one program named Main [OB1].



The following Steps for Building a Ladder Diagram are:

- Determine the No. of digital I/O
- Determine the No. of analog I/O (if needed)
- Determine if there are special functions in the process
- Estimate program capacity depending on the process
- Choose a suitable PLC series and prepare the wiring diagram
- Draw flowchart or control diagram (Optional) program the PLC using the ladder diagram

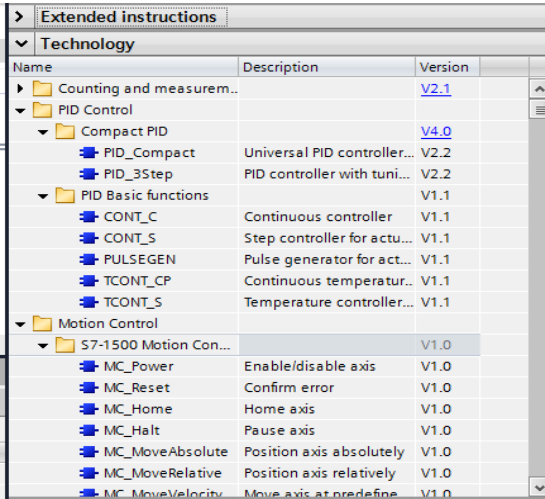


Figure 6 PID Control Box

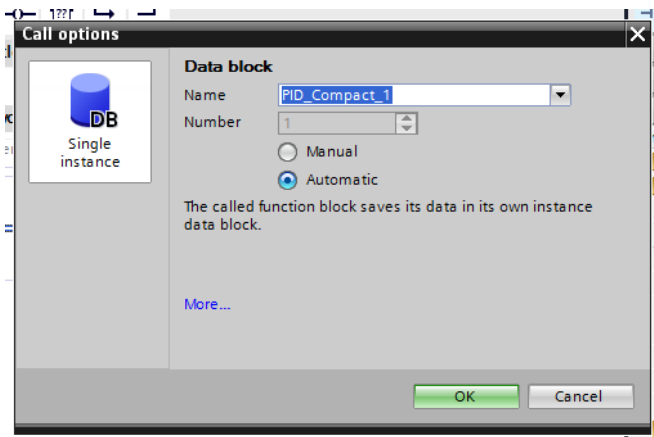


Figure 7 Data Block Call Options

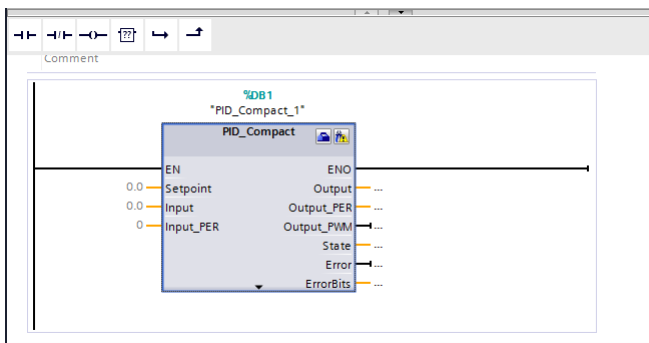


Figure 8 PID\_Compact Function

There are many different types of blocks which use within an S7 user program:

- Organization blocks (OB)

OBs determine the structure of the user program. System function blocks (SFB) and system functions (SFC), SFBs and SFCs are integrated in the S7 CPU and allow student access to some important system functions.

- Function blocks (FB)

FBs are blocks with a "memory" which student can program themselves.

- Functions (FC)

FCs contains program routines for frequently used functions.

- Instance data blocks (instance DB)

Instance DBs are associated with the block when an FB/SFB is called. They are created automatically during compilation.

- Data blocks (DB)

DBs are data areas for storing user data. In addition to the data that are assigned to a function block, shared data can also be defined and used by any blocks.

OBs, FBs, SFBs, FCs, and SFCs contain sections of the program and are therefore also known as logic blocks. The permitted number of blocks per block type and the permitted length of the blocks are CPU-specific.

## 6. MODEL STRUCTURE OF THE SAMPLE PROJECT

The sample project consists of the models listed in Table 1 that are independent of each other.

The S7 program of the CPU 1511-1PN has the following structure:

No	Models	Content of the model
1	Closed-loop control of a PT3 system simulation with the aid of "PID_Compact".	<ul style="list-style-type: none"> <li>• Parameterizing the PT3 system simulation.</li> <li>• Parameterizing and configuring "PID_Compact". Commissioning "PID_Compact" with pretuning and fine tuning.</li> </ul>
2	Closed-loop control of a more complex controlled system consisting of PT1, PDT1, lagging and PT2 element.	<ul style="list-style-type: none"> <li>• Interconnecting the individual system simulations.</li> <li>• Commissioning "PID_Compact" with pretuning and fine tuning.</li> </ul>

## 7. PID CLOSED-LOOP CONTROL SIMULATION RESULTS

For PID control simulation, download the program to the controller and the interface to the touch screen. Open the program block and turn on program monitoring by clicking the Monitoring on/off button in the toolbar. When the monitoring mode is on, actuated contacts, coils, or rungs turn green.

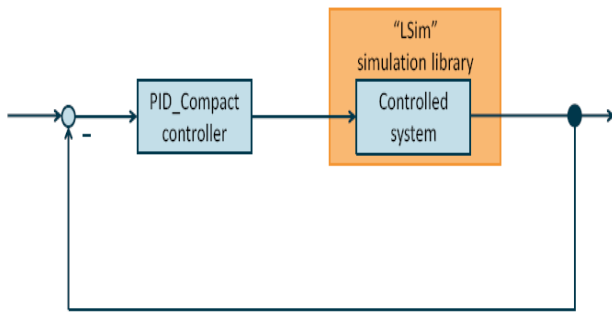


Figure 9 PID\_Compact Controller System Block Diagram

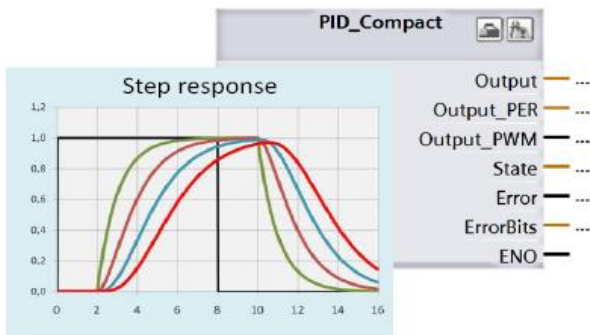


Figure 10 Response Curve

The automation task is to set up a control loop for influencing physical parameters in a technical process. The control loop is to consist of the following elements:

- "PID\_Compact" as the controller.
- Simulated technical processes as the controlled system.

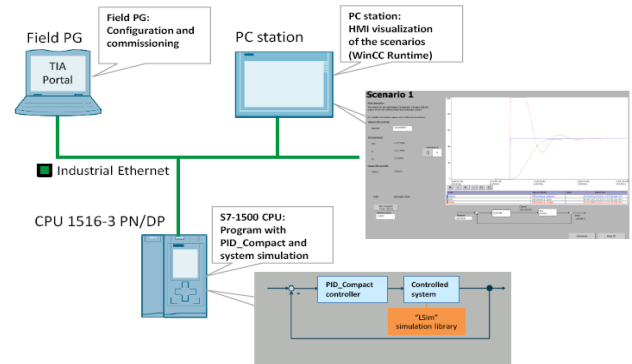


Figure 11 Solution overview

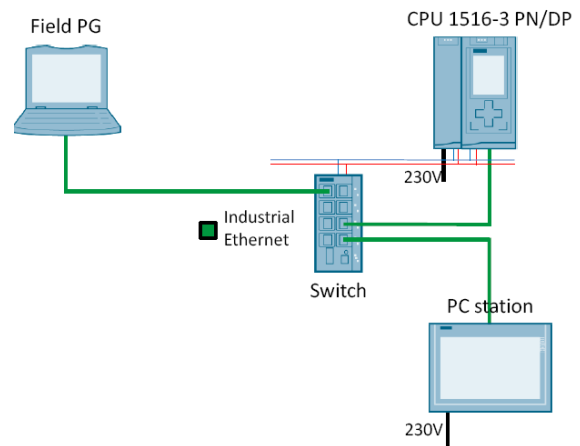


Figure 12 Connection with PC station and CPU

A simple example of a control loop is the control of the room temperature using a heater. The room temperature is measured with a sensor and fed to a controller. This controller compares the current room temperature to a setpoint and calculates an output value (manipulated variable) for controlling the heater.

### 7.1. Procedure Task

The following tasks have to be performed to implement the user program:

- Insert the PT3 system simulation into the user program and parameterize it.
- Add the "PID\_Compact" block to the user program.
- Configure the "PID\_Compact" block.
- Commission the software controller with pretuning and fine tuning.

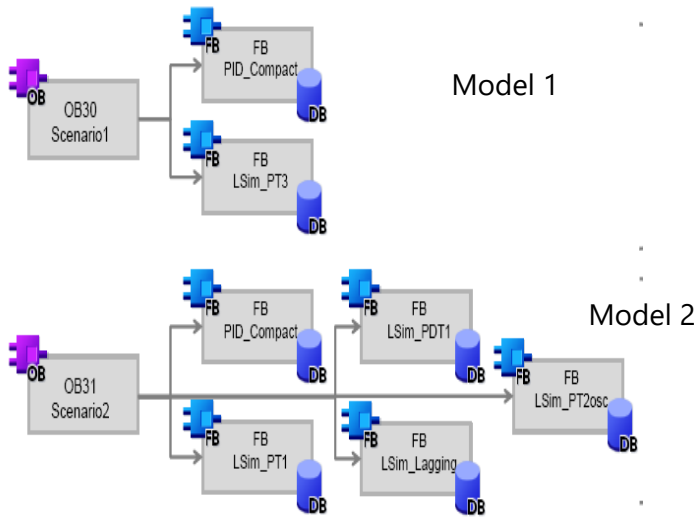


Figure 13 System Models

**7.2. Parameters and formula for “PID\_Compact”**

After fine tuning, the following parameters are active in the “PID\_Compact” software controller:

**Table 1 Symbols and parameters**

Symbol	Description	Value
Kp	Proportional gain	10.770338
Ti	Integral action time	21.10933
Td	Derivative action time	5.337515
a	Proportional action weighting	0.1
b	Derivative action weighting	0.2586402
c	Output value of the PID algorithm	0.0
y	Laplace operator	-
s	Setpoint	-
w	Process Value	-
x		-

The PID algorithm of “PID\_Compact” (FB1130) is based on the following formula:

$$y = K_p \left[ (b * w - x) + \frac{1}{T_i * s} (w - x) + \frac{T_d * s}{a * T_d * s + 1} (c * w - x) \right]$$

**7.3. Monitoring the controlled system**

In the PC station of the TIA project, a visualization user interface (WinCC Runtime) is provided that allows the user to use the examples.

WinCC Runtime allows the user to:

- Monitor the state of the project’s scenarios
- Modify individual tags of the scenarios.

In the PC station of the TIA project, a visualization user interface (WinCC Runtime) is provided that allows the user to use the examples.

WinCC Runtime allows the user to:

- Monitor the state of the project’s scenarios
- Modify individual tags of the scenarios.

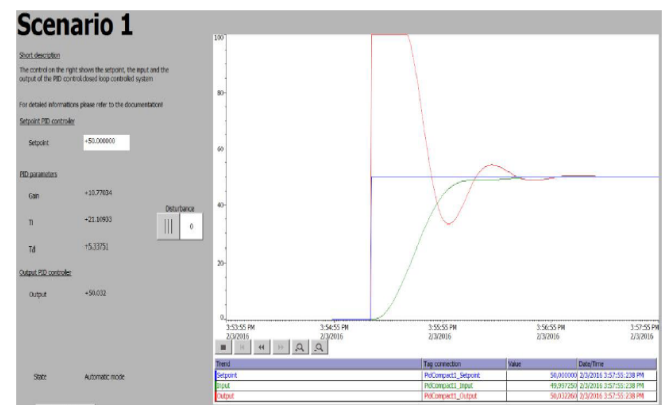


Figure 14 Visualization user interface WinCC Runtime (model1)

**7.4. Calling and commissioning “PID\_Compact”**

Calling and commissioning show how to simulate a PT3 controlled system with the simulation library and control the PT3 controlled system with the “PID\_Compact” block.

Fi

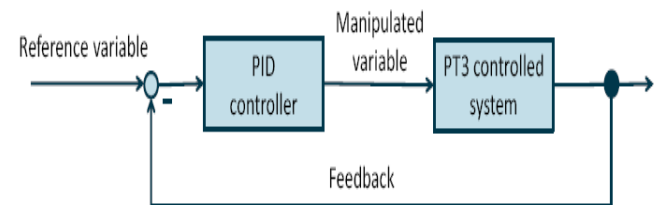


Figure 15 PID Controller Schematic diagram

The figure below shows the step response of the PT3 controlled system when the input jumps from 0•50:

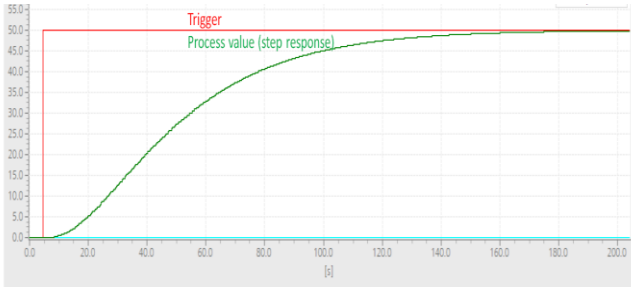


Figure 16 Step response

This simulation describes how to monitor and control the controlled system with the aid of the provided visualization using WinCC Runtime Advanced.

After commissioning Scenario1, a setpoint step-change from 0•50% results in the following response:

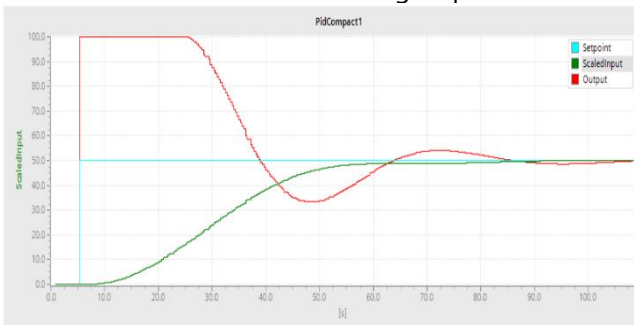


Figure 17 Control response of the system

## 8.DISCUSSION AND CONCLUSIONS

For a better overview of the behavior of the implemented scenarios, several options are available to the user:

- Insight into the behavior of the control loops using the WinCC Runtime Advanced HMI system.
- Detailed insight into the current status of the control loop through the watch tables already prepared in the CPU.

Analysis of PID control system using TIA portal can select proportional, integral, derivative gain for better response. Observation response curve in running, applications can be controlled how much output results are needed for this application. For students, it is intended how to create closed loop system for analog applications without real hardware PLC device. Using these Siemens TIA portal software tools favorably completes the development of skills for students and they become familiar with software tools widely used in

industrial companies where its use is extensive. All of them can be configured, programmed and tested with Siemens TIA portal V13 software. The different model can be selected from the start screen of the WinCC Runtime system running on the "VisuPC" PC station for future work.

## 9.ACKNOWLEDGMENT

Especially, the author special thanks to her family who have given support and encouragement.

## 10.REFERENCES

### Article/ Research Paper

- [1] Jose Luis Vazquez-Gonzalez, "An Industrial Automation Course: Common Infrastructure for Physical, Virtual and Remote Laboratories for PLC Programming", iJOE – Vol. 14, No. 8, 2018, pp. 4–19.

### Handbooks

- [1] Müller/ Pfeiffer/ Wieser, SIMATIC S7 und SIMATIC PCS, Publicis Publishing, Erlangen, ISBN: 978-3-89578-340-1
- [2] Hans Berger, "Automating with SIMATIC S7-1500 Configuring, Programming and Testing with STEP 7 Professional", by Publicis Erlangen, Zweigniederlassung der PWW GmbH, 2014.

### Online Sources

- [1] Function Manual: SIMATIC S7-1200, S7-1500: PID Control  
<https://support.industry.siemens.com/cs/ww/en/view/108210036>
- [2] Download page of this entry  
<https://support.industry.siemens.com/cs/ww/en/view/79047707>
- [3] System Manual: STEP 7 Professional V13.1  
<https://support.industry.siemens.com/cs/ww/en/view/109011420>
- [4] Function Manual: SIMATIC S7-1200, S7-1500: PID Control  
<https://support.industry.siemens.com/cs/ww/en/view/108210036>



[5] System Manual: SIMATIC S7-1500 Automation System

<https://support.industry.siemens.com/cs/ww/en/view/59191792>

[6] System Manual: WinCC Professional V13 SP1

<https://support.industry.siemens.com/cs/ww/en/view/109096785>

[7] Getting Started: SIMATIC S7-1500: Installing the Assembly

<https://support.industry.siemens.com/cs/ww/en/view/71704272/52316681355>

[8] Application Example: „Single and Multi Loop Controller Structures (Cascade Control) with PID\_Temp“

<https://support.industry.siemens.com/cs/ww/en/view/103526819>