

# A DUAL-RATE MODEL PREDICTIVE CONTROLLER FOR FIELDBUS BASED DISTRIBUTED CONTROL SYSTEMS

(Spine title: A Dual-Rate Model Predictive Controller for Fieldbus Based DCS)

(Thesis format: Monograph)

by

Mohammad Arif Hossain

Graduate Program in Engineering Science

Department of Electrical & Computer Engineering

A thesis submitted in partial fulfillment  
of the requirements for the degree of  
Master of Engineering Science

The School of Graduate and Postdoctoral Studies  
The University of Western Ontario  
London, Ontario, Canada

© Mohammad Arif Hossain 2013

THE UNIVERSITY OF WESTERN ONTARIO  
School of Graduate and Postdoctoral Studies

**CERTIFICATE OF EXAMINATION**

Supervisor

Examiners

\_\_\_\_\_  
Dr. Jin Jiang

\_\_\_\_\_  
Dr. Lyndon J. Brown

\_\_\_\_\_  
Dr. Mehrdad R. Kermani

\_\_\_\_\_  
Dr. Sohrab Rohani

The thesis by

**Mohammad Arif Hossain**

entitled:

**A Dual-Rate Model Predictive Controller for Fieldbus Based  
Distributed Control Systems**

is accepted in partial fulfillment of the  
requirements for the degree of  
Master of Engineering Science

\_\_\_\_\_  
Date

\_\_\_\_\_  
Chair of the Thesis Examination Board

# ABSTRACT

In modern Distributed Control Systems (DCS), an industrial computer network protocol known as fieldbus is used in chemical, petro-chemical and other process industries for real-time communication between digital controllers, sensors, actuators and other smart devices. In a closed-loop digital control system, data is transferred from sensor to controller and controller to actuator cyclically in a timely but discontinuous fashion at a specific rate known as sampling-rate or macrocycle through fieldbus. According to the current trend of fieldbus technology, in most industrial control systems, the sampling-rate or macrocycle is fixed at the time of system configuration. This fixed sampling-rate makes it impossible to use a multi-rate controller that can automatically switch between multiple sampling-rates at run time to gain some advantages, such as network bandwidth conservation, energy conservation and reduction of mechanical wear in actuators.

This thesis is concerned about design and implementation of a dual-rate controller which automatically switches between the two sampling-rates depending on system's dynamic state. To be more precise, the controller uses faster sampling-rate when the process goes through transient states and slower sampling-rate when the process is at steady-state operation. The controller is based on a Model Predictive Control (MPC) algorithm and a Kalman filter based observer.

This thesis starts with theoretical development of the dual-rate controller design. Subsequently, the developed controller is implemented on a Siemens PCS 7 system for controlling a physical process. The investigation has concluded that this control strategy can indeed lead to conservation of network bandwidth, energy savings in field devices and reduction of wear in mechanical actuators in fieldbus based distributed control systems.

**Keywords:** Model Predictive controller, Dual-rate MPC, Fieldbus bandwidth conservation

# ACKNOWLEDGEMENT

My first and foremost acknowledgement goes to my supervisor, Dr. Jin Jiang, for giving me the opportunity to work under his supervision for this research. I wish to express my sincere gratitude and appreciation to Dr. Jin Jiang for giving the privilege of using the state-of the art facilities of his lab. I am also grateful to my co-supervisor Dr. Qingfeng Li and my research mate Mr. Drew Rankin for their motivation, support and guidance throughout the research which significantly contributed to the timely completion of this thesis and helped me in my academic career.

I am also thankful to all the members of the Control Instrumentation and Electrical Systems research group of Western University for their support in various aspects of my research work.

I deeply express my special thanks to my wife for her motivation and support and to my parents for their endless love and prayers.

I would like to acknowledge the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) and the University Network of Excellence in Nuclear Engineering (UNENE).

# TABLE OF CONTENTS

ABSTRACT .....	III
ACKNOWLEDGEMENT .....	IV
TABLE OF CONTENTS .....	V
LIST OF TABLES .....	VIII
LIST OF FIGURES .....	IX
NOMENCLATURE .....	XII
<b>Chapter 1 .....</b>	<b>1</b>
INTRODUCTION .....	1
1.1 Background .....	1
1.2 Dual-rate sampling problem in fieldbus based control systems .....	3
1.3 Motivations .....	4
1.4 Objective, scope and limitation.....	5
1.5 Contributions.....	7
1.6 Organization of the thesis .....	8
<b>Chapter 2 .....</b>	<b>10</b>
LITERATURE SURVEY .....	10
2.1 Distributed Control Systems and Evaluation of Communications .....	10
2.2 Fieldbus Technology.....	13
2.3 Adoption of Fieldbus Technologies in Distributed Control Systems .....	14
2.4 HART: Introduction of Digital Communication over Analog Channels.....	15
2.5 Foundation Fieldbus.....	16
2.5.1 Control in the Field .....	18
2.5.2 Sequence of Execution and Scheduling in Foundation Fieldbus.....	20
2.6 DIGITAL CONTROL SYSTEMS .....	23
2.6.1 Basic Structure of Digital Control Systems .....	23
2.6.2 Multi-rate Sampling in Digital Control Systems .....	26
2.6.3 Multi-rate Sampled-data Systems .....	27
2.6.4 Example of a Multi-rate Control System .....	28
2.7 Model Predictive Control.....	29
2.7.1 MPC Strategy.....	31
2.8 PREVIOUS WORKS ON MULTI-RATE CONTROL SCHEME .....	33
2.8.1 Multi-rate LQR Controller.....	33
2.9 SUMMARY .....	35
<b>Chapter 3 .....</b>	<b>37</b>
DUAL-RATE MODEL PREDICTIVE CONTROLLER DESIGN .....	37
3.1 Model Predictive Controller Design .....	37
3.1.1 Augmented Model of the Plant .....	37
3.1.2 Predictive Control within One Optimization Window .....	39
3.1.3 Optimization .....	42
3.1.4 Receding Horizon Control .....	43
3.1.5 Discrete-time Predictive Control System (without observer).....	44

3.1.6	Observer based Predictive Controller .....	45
3.1.7	Kalman Filter .....	46
3.1.8	Predictive Control with State Estimation.....	47
3.2	Dual-rate Model Predictive Controller Mechanism.....	49
3.3	Automatic Sampling-rate Switching Strategy .....	50
3.3.1	Control Signal Switching Strategy.....	51
3.3.2	Sensor Sampling-rate Switching Strategy .....	52
<b>Chapter 4</b>	<b>.....</b>	<b>53</b>
IMPLEMENTATION OF DUAL-RATE MPC ON SIEMENS PCS 7 DCS .....		53
4.1	Why Siemens PCS 7 .....	54
4.2	Experimental Setup and Modes of Experiment .....	55
4.3	Block Diagram of the Dual-rate MPC Control Loop.....	56
4.4	Physical Setup .....	57
4.5	Dual-rate MPC Design Formulation and Parameters .....	59
4.5.1	Dual-rate MPC in MODE-1 .....	59
4.5.2	Dual-rate MPC in MODE-2.....	60
4.6	Implementing Dual-rate MPC Algorithm in SIMATIC Manager .....	61
<b>Chapter 5</b>	<b>.....</b>	<b>64</b>
EXPERIMENTAL RESULTS AND DISCUSSION .....		64
5.1	Test Scenarios .....	65
5.2	Cost-change due to Set-point Change.....	66
5.3	Sampling-rate Switching on Cost Change .....	67
5.4	Sampling-rate Switching on Step-change .....	68
5.5	Performance Comparison between Single-rate and Dual-rate Settings.....	69
5.6	Effect of External Disturbance on Sampling-rate Switching.....	71
5.7	Effects of Transients on Dual-rated Actuator Signal.....	74
5.8	Effect of Disturbance in between Samples in MODE-2 .....	75
5.9	DISCUSSION .....	76
5.9.1	Network Bandwidth Conservation Mechanism using Dual-rate MPC.....	77
5.9.2	Energy Conservation Mechanism using Dual-rate MPC.....	79
5.9.3	Communication Protocol and Hardware Requirement for Energy Conservation in Field Devices .....	81
<b>Chapter 6</b>	<b>.....</b>	<b>83</b>
CONCLUSIONS.....		83
6.1	Conclusions.....	83
6.2	Contributions.....	84
6.3	Future Works .....	85
<b>REFERENCES</b> .....		<b>89</b>
<b>APPENDIX A</b> .....		<b>94</b>
HARDWARE COMPONENTS OF SIEMENS .....		94
PCS 7 DCS AND MODEL PLANT .....		94
A.1	Siemens PCS 7 DCS Hardware Components .....	94
A.2	Siemens PCS 7 Software Components .....	99
A.3	The model plant .....	100
<b>APPENDIX B</b> .....		<b>104</b>

MODEL PLANT TRANSFER FUNCTION .....	104
B.1 Mathematical model of the plant .....	104
<b>APPENDIX C .....</b>	<b>105</b>
INTRODUCTION TO SIEMENS SIMATIC MANAGER FOR DEVELOPING	
DUAL-RATE MPC USING PCS 7 .....	105
C.1 Creating a New Project on PCS 7 SIMATIC Manager .....	105
C.2 Hardware Configuration .....	112
C.2.1 Example Procedure of Hardware Configuration.....	113
C.3 Running the Application .....	117
<b>APPENDIX D .....</b>	<b>120</b>
SIEMENS STRUCTURED CONTROL LANGUAGE (SCL) BASICS .....	120
D.1 SLC Basics.....	120
D.2 SCL Functionality .....	120
D.3 SCL Development Environment.....	121
D.4 Blocks in S7-SCL Source Files.....	122
D.5 Calling Order of the Blocks in SCL Source File .....	122
D.6 General Structure of a Block.....	123
D.7 Block Start and End Identifiers.....	124
D.8 Declarations of Local Variables, Parameters, Constants, and Labels.....	125
D.9 Data Types .....	126
D.10 Example SCL Code for a Function Block .....	127
<b>VITAE.....</b>	<b>130</b>

# LIST OF TABLES

Table 4.1: Modes of experimental setup.....	55
Table 4.2: Components of the experimental setup.....	57
Table 5.1: Test scenarios.....	65
Table 5.2: Quantitative comparison of single-rate and dual-rate controllers.....	70
Table 5.3: Power consumption by SiM3U1xx microcontroller.....	80
Table 5.4: Energy requirement in active and sleep mode .....	80
Table A.1: CPU-414-4H features .....	95
Table B.1: Plant response at different frequency .....	104
Table C.1: New project creation procedure .....	105
Table C.2: MPC_DUAL SCL source file modules .....	109
Table C.3: Steps for hardware configuration.....	113
Table D.1: Start and end syntax for various blocks .....	124
Table D.2: Syntax of declarations of different types of data .....	125
Table D.3: SCL elementary data types .....	126
Table D.4: SCL complex data types .....	127
Table D.5: Example SCL code explanation.....	129



# LIST OF FIGURES

Fig 1.1: Fieldbus based distributed control system [3] .....	2
Fig 2.1: Direct Digital Control through field devices .....	10
Fig 2.2: Hierarchical process control architecture .....	11
Fig 2.3: Control devices distributed near field devices.....	12
Fig 2.4: Distributed control system through LAN and fieldbus .....	13
Fig 2.5: HART signal [11] .....	16
Fig 2.6: A flow control loop [12].....	17
Fig 2.7: Control in the field [12].....	18
Fig 2.8: Distribution of function blocks in field devices [12].....	19
Fig 2.9: Foundation Fieldbus macrocycle task sequencing and scheduling in LAS .....	20
Fig 2.10: Multiple control loops on fieldbus and Link Master .....	21
Fig 2.11: Multiple control loop task scheduling in LAS .....	22
Fig 2.12: Block diagram of a digital control system.....	24
Fig 2.13: Multi-rate sampled-data system .....	27
Fig 2.14: Multi-rate position servo controller.....	29
Fig 2.15: (a) Model Predictive Control strategy (b) MPC block diagram .....	31
Fig 3.1: Model predictive controller block diagram with an integrator.....	37
Fig 3.2: MPC without an observer .....	45
Fig 3.3: MPC with an observer .....	46
Fig 3.4: Dual-rate model predictive controller .....	50
Fig 4.1: Siemens PCS 7 DCS block diagram.....	54
Fig 4.2: Block diagram of the dual-rate MPC control loop .....	56
Fig 4.3: Physical setup containing Siemens PCS 7 DCS, process and workstation .....	57

Fig 4.4: Dual-rate MPC on CFC editor.....	61
Fig 4.5: Dual-rate MPC Function Block and model plant block diagram.....	62
Fig 4.6: Trend Display showing various signals, resolution 1 sec (minimum) .....	63
Fig 5.1: Step-change transient causing MPC1 and MPC2 cost change on (a) MODE-1 (b) MODE-2 .....	66
Fig 5.2: Change of MPC cost initiating sampling-rate switching (a) MODE-1 (b) MODE-2, (threshold cost 2.0 in both cases).....	67
Fig 5.3: Step-change transient causing sampling-rate change (a) MODE-1, threshold cost 2.0 (b) MODE-2, threshold cost 2.0 .....	68
Fig 5.4: Performance comparison between single-rate controllers vs. a dual-rate controller (a) MODE-1, (b) MODE-2 .....	69
Fig 5.5: Effect of external disturbance in MODE-1, (a) External disturbance induced cost change, (b) Cost change initiated sampling-rate change (c) External disturbance on plate level induced actuator response in opposite phase.....	72
Fig 5.6: Effect of external disturbance in MODE-2, (a) External disturbance induced cost change, (b) Cost change initiated sampling-rate change (c) External disturbance on plate level induced actuator response in opposite phase.....	73
Fig 5.7: Step-change causing dual-rated actuator signal .....	74
Fig 5.8: Plate level change on external disturbance pulses on actuator signal causing actuator signal change (a) constant sensor sampling, variable actuator sampling, (b) both sensor and actuator sampling in variable mode .....	75
Fig 5.9: Simple Foundation Fieldbus control loop .....	78
Fig 5.10: Faster and slower sampling-rate effecting bandwidth (a) Macrocycle at fast rate, (b) Macrocycle at slow rate conserving network bandwidth .....	78
Fig 5.11: Silicon Labs Precision32 SiM3U1xx microcontroller power consumption.....	79
Fig 5.12: Devices having hardware for recognizing wake-up frames .....	82
Fig 5.13: Controller communicating with device after waking it up.....	82
Fig A.1: Siemens PCS 7 DCS components in panel board.....	94
Fig A.2: Model Plant Components .....	100
Fig A.3: (a) Wenglor CP35MHT80 is a LASER displacement sensor (b) Teknocraft 203314 Proportional flow control valve .....	101

Fig A.4: Flow curve for 0.375 balanced 24volt coil proportional solenoid valve .....	103
Fig C.1: Creating a new project in SIMATIC Manager .....	107
Fig C.2: (a) Dual-rate MPC project on SIMATIC Manager (b) SCL editor and compiler .....	108
Fig C.3: Newly generated MPC_DUAL as FB27 .....	110
Fig C.4: SIMATIC CFC showing dual-rate MPC application .....	111
Fig C.5: Block Types child window for importing function blocks to CFC .....	112
Fig C.6: General hardware configuration procedure in HW Config .....	115
Fig C.7: Hardware Configuration for dual-rate MPC project.....	116
Fig C.8: Macrocycle property setting on CPU properties.....	117
Fig C.9: Operating mode setting .....	118
Fig C.10: Cold starting the dual-rate MPC program.....	119
Fig C.11: Trend display for viewing various parameters online .....	119
Fig D.1: Supported blocks .....	121
Fig D.2: Development environment of SCL.....	122
Fig D.3: Proper sequence of definition of blocks in SCL file .....	123

# NOMENCLATURE

## Abbreviations:

A/D	Analog to Digital
AC	Alternating Current
AI	Analog Input
AO	Analog Output
CAN	Controller Area Network
CFC	Continuous Function Chart
CIF	Control In the Field
CiR	Configuration in RUN
CP	Communication Processor
CPU	Central Processing Unit
D/A	Digital to Analog
DB	Data Block
DC	Direct Current
DCS	Distributed Control System
DDC	Direct Digital Control
DP	Distributed Peripheral
DSP	Digital Signal Processor
ESM	Electrical Switch Module
FB	Function Block
FC	Function
FF	Foundation Fieldbus
FSK	Frequency Shift Keying
HART	Highway Addressable Remote Transducer protocol
HSE	High Speed Ethernet
I/O	Input Output
IC	Integrated Circuit

IDE	Integrated Development Environment
IEC	International Electrotechnical Commission
IO	Input Output
IP	Internet Protocol
ISO	International Organization for Standardization (disambiguation)
KB	Kilo Bytes
LAS	Link Active Scheduler
LASER	Light Amplification by Stimulated Emission of Radiation
LED	Light Emitting Diode
LIN	Local Interconnect Network
LQG	Linear-Quadratic Gaussian
LQR	Linear-Quadratic Regulator
MIS	Management Information System
MPC	Model Predictive Control
MPI	Multi Point Interface
NPT	National Pipe Thread
NSERC	Natural Sciences and Engineering Research Council of Canada
OB	Organization Block
OSM	Optical Switch Module
PA	Process Automation
PC	Personal Computer
PID	Proportional Integral Derivative
PLC	Programmable Logic Controller
PT	Pass Token
PWM	Pulse Width Modulation
RAM	Random Access Memory
RTC	Real Time Clock
SCL	Structured Control Language
SFB	System Function Block
SFC	Sequential Function Chart
SISO	Single Input Single Output

SNMP	Small Network Management Protocol
ST	Structured Text (a programming language)
SW1	Switch 1
SW2	Switch 2
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UDT	User Data Type
UNENE	University Network of Excellence in Nuclear Engineering
VDC	Volt DC (Direct Current)
VLSI	Very Large Scale Integration

### **Symbols:**

$y_1(t)$	First output of multi-rate sampled-data system
$m_1h$	Sampling-rate of the first output of multi-rate sampled-data system
$m_ph$	Sampling-rate of the second output of multi-rate sampled-data system
$n_1h$	Sampling-rate of the first control signal of multi-rate sampled-data system
$K(z, lh)$	LTI multi-rate controller
$y(t + k   t)$	Predicted future outputs of MPC
$u(t + k   t)$	Future control signals of MPC
$w(t + k)$	Reference trajectory of MPC
$T$	Short time period of multi-rate LQR controller
$T_1$	Long time period of multi-rate LQR controller
$L = T_1/T$	Ratio of long and short time periods of multi-rate LQR controller
$Z_{i+1}$	State vector of the system controlled by multi-rate LQR controller
$U_i$	Input vector of the system controlled by multi-rate LQR controller
$J_{LQR}$	Quadratic performance index of LQR controller
$\Delta u(k)$	Change of process input or change of MPC control signal
$u$	Process input variable or MPC control signal or manipulated variable

$x_m$	State variable vector with assumed dimension $n_1$
$y$	Process output
$N_p$	Number of samples or the prediction horizon of MPC
$N_c$	Control horizon dictating the number of parameters of MPC
$R_S^T$	Data vector containing the set-point information
$r_w$	Tuning parameter for the expected closed-loop performance in MPC
$J$	MPC cost function
$R_s$	Data vector containing the set-point data
$r(k_i)$	Set-point signal of MPC
$x(k_i)$	State variable of MPC
$K_y$	Part of MPC gain
$K_{mpc}$	MPC gain
$K_x$	Part of MPC gain
$K_{ob}$	Optimum observer gain
$d$	Gaussian process noise
$\xi$	Gaussian measurement noise
$\theta$	Covariance matrix of Kalman Filter
$\Gamma$	Covariance matrix of Kalman Filter
$\hat{x}$	Observer state variable

# Chapter 1

## INTRODUCTION

### 1.1 Background

In course of time, modern control systems evolved from analog to digital which accelerated in the past few decades due to the advent of powerful digital microprocessor and microcontroller technology. Control industry adopted the computer network based communication technology known as fieldbus to communicate between controllers, sensors, actuators and other devices for control, operation, monitoring, configuration and diagnostics. The industry has gradually replaced the 4-20mA legacy analog technology with new fieldbus technology wherever possible considering its numerous advantages. To be more precise, fieldbus is a digital network communication technology used for real time control. Like its predecessor, the computer network technology, it works on various topologies e.g. daisy chain, star, ring, branch and tree etc. topologies [1], [27].

There are numerous advantages of using fieldbus technology in modern distributed control systems. One of the major advantages is capital saving on wiring. In contrast with earlier 4-20mA schemes, i.e. point-to-point topology, where each device had to be connected to the controller separately, in fieldbus technology, devices can be connected in various multi-drop topologies which saves considerable amount of wiring and makes things simpler to configure and which, in turn, reduces cost of installation and maintenance of equipment. Fieldbus based systems are simpler to design as well, which, in turn, reduces system designing and drawing costs [2].

Lower maintenance is another important feature of fieldbus based distributed control systems. Other than cost savings on wiring, fieldbus systems due to its simplicity in



structure and well equipped with diagnostics capability makes it easier to maintain than conventional analog control systems [2].

Considering performance, fieldbus makes system design much more flexible with bus system architecture. In some fieldbus technologies, algorithms and control procedures can reside on the field devices which relieve the burden of the main controller. This is known as Control in the Fields (CIF) and it helps decentralize control loops and improve overall control performance.

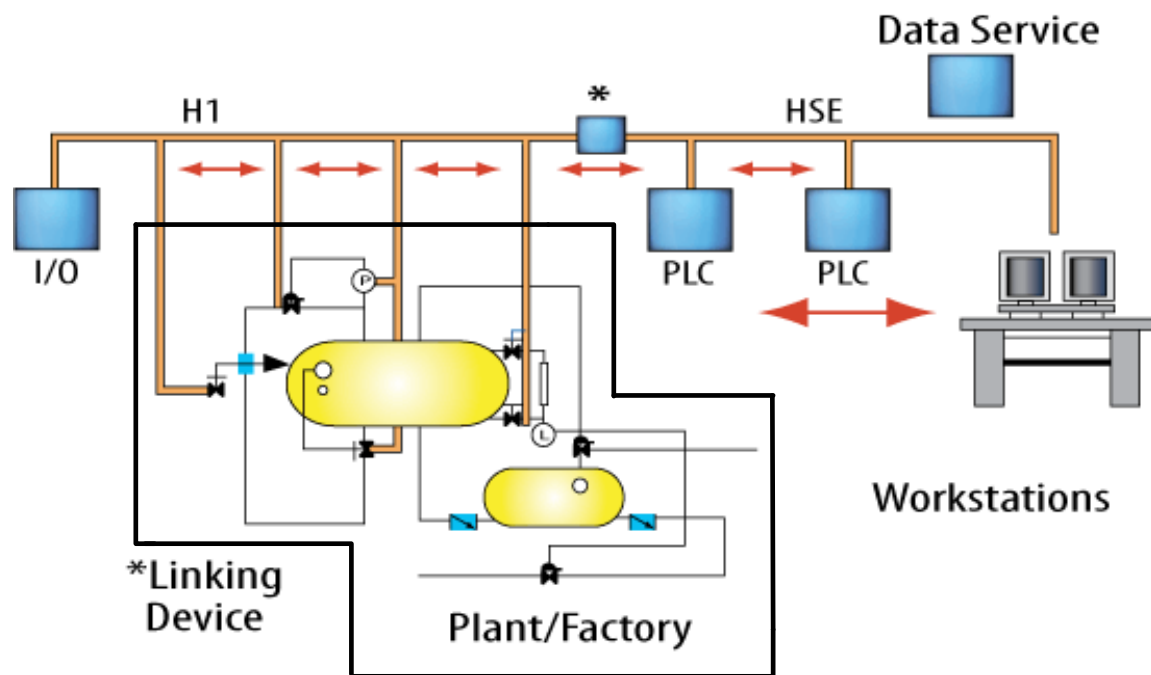


Fig 1.1: Fieldbus based distributed control system [3]

Figure 1.1 shows a typical fieldbus based distributed control system where the workstations, data servers and PLCs or Process Controllers are connected through Foundation Fieldbus HSE (High Speed Ethernet) and a Linking Device is used to connect the plant which uses Foundation Fieldbus H1 for communicating with sensors, actuators and other remote digital and analog I/O on the plant side.

## **1.2 Dual-rate sampling problem in fieldbus based control systems**

Modern fieldbus based DCS uses a single fixed sampling-rate for sensor and actuator data sampling in a closed-loop control environment. This sampling-rate also known as macrocycle is set while the system is being configured during commissioning and remains fixed during working range of the controller.

In a fieldbus based distributed control system, two or more control loops or other instruments share the same communication media for data transfer and there might be conflicts between the loops and devices for getting access to the network medium. This problem is solved by using a scheduling algorithm in the communication protocol.

As fieldbus is a digital network bus system communicating with several devices with a physical wire medium by sending data packets in a time shared mechanism, in control application, fieldbus has limitation of data bandwidth depending on speed of communication due to sharing of medium. Therefore, if any control algorithm can reduce cyclic control data transmission, the freed data bandwidth can be used for other purposes e.g. device diagnostics, operation, configuration, maintenance and other non-real time data transmissions.

In a control system, sensor and actuator sampling-rates are chosen based on the plant dynamics. Faster sampling-rate is required when a plant undergoes a transient period, such as a set-point change or external disturbance. But when the system reaches steady-state, sampling-rate needs not be that high any more without significantly affecting and performance of the system. In general, most process control systems use optimal sampling-rate calculated from the plant dynamic model. The chosen sampling-rate remains fixed throughout the operation of the control system.

In a dual-rate controller, during dynamic transient periods e.g. starting, changing set-point or external disturbance, the controller uses the faster sampling-rate and when the plant reaches steady-state, the controller automatically switches to the slower sampling-

rate. Hence, a dual-rate controller has potential to save network bandwidth in a fieldbus based industrial control application.

To design a dual-rate controller, one has to deal with the following challenges:

- 1) Developing a mechanism to distinguish transient from steady-state operation to switch between the two sampling-rates,
- 2) Selecting a suitable slower sampling-rate, so that it does not lead to degraded performance in the closed loop-system,
- 3) Needs a new kind of fieldbus protocol capable of switching sampling-rate or macrocycle on-line to facilitate the transitions,
- 4) Needs smart sensors and actuators that are capable of responding and working with the new multi-rate sampling protocol along with the proposed controller

### **1.3 Motivations**

Nowadays, use of fieldbus in DCS is becoming a de-facto standard in many process industries around the world due to its various cost, implementation, performance and maintenance advantages over traditional analog communication technologies. However, despite of these advantages, it has some limitations including possibility of bandwidth congestion when a large number of smart sensor, actuator and device nodes are used on a slow fieldbus network. The current fieldbus technologies e.g. Foundation Fieldbus H1, Profibus PA etc. are considerably slow speed (31.25 kBit/s, IEC 61158-2) communication technologies. As fieldbus technology advances with more and more smart devices being introduced, there is a need for more bandwidth requirement on fieldbus networks in the near future [4]. This lack of bandwidth problem in control application can be handled in two ways as follows:

- (a) Increasing the communication channel speed of the fieldbus network

- (b) Reducing the data transfer rate in fieldbus based control loops by switching to slower sampling-rates whenever possible by using advanced control algorithms

In this thesis, the concentration is on the latter i.e. reduction of sampling data through fieldbus using advanced control schemes. By doing so, several important advantages along with the bandwidth conservation can also be achieved:

- (a) Reduction of overall energy consumption on the devices due to slower sampling-rate by putting them in the sleep mode when they are not active
- (b) Reduction of mechanical wear of the actuators due to fewer control signal updates

The world is becoming energy conscious day-by-day. Reduction of data traffic in fieldbus is one of the ways to reduce energy consumption on a fieldbus based distributed control system. The less the data transfer needed from the devices the more they can become idle and can be put to a sleep mode. Although by increasing the communication channel speed, the bandwidth problem on a fieldbus can be solved, the energy requirement of the distributed control systems cannot be conserved that way. Also, high speed communication over long distance is prone to external noise which is undesirable in many industrial applications. Moreover, increasing fieldbus communication channel speed can, in fact, increase the energy requirement of the system rather than reducing it.

Every mechanical device, such as an actuator has finite mechanical life. Reduction of unnecessary actuator operation means reduction of wear on the mechanical parts of the plant. This can contribute to the reduction of the overall maintenance cost of the plant.

## **1.4 Objective, scope and limitation**

In this thesis, reduction of the data transfer through fieldbus communication network i.e. increase of data bandwidth through fieldbus, has been investigated by reducing the sensor and actuator sampling-rate of the control system by using a dual-rate controller. To design and implement such a controller, Model Predictive Control (MPC) scheme is

selected amongst various control schemes because of its capability of calculating optimization cost which directly relates to the status of system operation. In a MPC, when the system reaches steady-state, the cost becomes close to zero. The cost increases again when transient occurs.

A dual-rate MPC is designed to achieve following objectives:

- 1) It will reduce the amount of data transferred to conserve valuable network bandwidth by automatically switching to slower sampling-rate whenever possible,
- 2) It will use the faster sampling-rate to handle transients e.g. starting, set-point change or external disturbance so that control performance is not sacrificed,
- 3) Control performance will not degrade below a certain threshold while using the slower sampling-rate, and
- 4) It should be easily implementable on the existing DCS on the market

The major challenge of designing such a dual-rate controller is to find a function index to identify the status of the system so that switching between the two sampling-rates can be initiated. This thesis mainly devises a control algorithm which finds such an index and uses that index to automatically switch between slower and faster sampling-rates. To achieve this, an advanced control algorithm known as Model Predictive Control is used.

A dual-rate MPC is constructed through two observer based MPCs with two different sensor and actuator sampling-rates. The manipulated variable or the plant input is connected to one of the MPC outputs. The switch selects one of the MPC outputs amongst the two by comparing the costs of the two MPCs with a threshold cost which becomes the manipulated variable to the plant.

The ease of implementation of the designed dual-rate MPC has been illustrated by implementing it on a Siemens PCS 7 DCS. The implemented controller controls a model plant in the Lab to demonstrate the effectiveness of the dual-rated control scheme. All the measurements are taken from the Trend Display of the PCS 7's SIMATIC Manager.

Another major challenge is finding a suitable fieldbus technology that can handle dual or multiple sampling-rates on-line. Unfortunately, no current fieldbus technology on the market supports multiple sampling-rates on Siemens PCS 7 DCS. Therefore, although this research concentrates on reduction of the fieldbus data transfer rate using an advanced control scheme, no real physical fieldbus network can be used to demonstrate the idea. Instead, to prove the concept of dual-rate control scheme for bandwidth conservation in a process controller, traditional 4-20mA analog communication scheme is used which can simulate multiple sampling-rates when periodically read or written through the use of a software timer. The period of the timer can have multiple values to simulate multi-rate sampling technique.

In this thesis, the slower sampling-rate is chosen to be as 5 times slower in one set of experiments (MODE-1) and 15 times slower in another set (MODE-2) than the faster (optimal) sampling-rate. This thesis only demonstrates that, a slower sampling-rate is possible which can conserve network bandwidth but too slow a sampling-rate may cause system instability.

This thesis also does not address on optimizing the *threshold cost* of the dual MPCs for switching between the slower and the optimal sampling-rates.

## 1.5 Contributions

The contributions of this thesis can be summarized as follows:

1. This thesis addresses the bandwidth conservation scheme that can be implemented on a fieldbus based distributed control system. In a fieldbus based DCS, slower sensor and actuator sampling-rate than normal can be used when the system achieves steady-state conditions. This slower sampling-rate can have some distinctive advantages e.g. network bandwidth conservation, which, in turn, can improve data transmission capability on the same network medium. Slower sampling-rate can also contribute to

energy conservation if smart devices with the sleep mode in between samplings are adopted and also can reduce mechanical wear of the actuators.

2. Dual-rate controller design using Model Predictive Control for automatically switching between two sample rates is a unique contribution of this thesis.
3. To implement the proposed control scheme for achieving desired objectives, Siemens PCS 7 DCS is used which is a widely used in industries throughout the world. The main purpose of this implementation is to demonstrate the ease of implementation of the proposed control scheme in an industrial setting.

## 1.6 Organization of the thesis

The rest of the thesis is organized as follows:

**Chapter 2** presents the literature survey which covers fundamentals of modern distributed control systems and fieldbus technologies which can help the reader better understand the problem addressed as well as the solution proposed. This includes the evolution of communication technology in modern industrial control systems and an overview of fieldbus technologies used in distributed control systems and principles of advanced Foundation Fieldbus technology. It also describes fundamentals of digital control system, types, strategies and design schemes, MPC fundamentals and strategies. The chapter also discusses the previous works done on multi-rate control technology and also its applications on computer network communication based control systems used in industry.

**Chapter 3** presents the design procedure and algorithms of a dual-rate MPC. The design procedure starts from a theoretical model predictive controller then extended with an addition of a Kalman filter based observer to be used in a practical industrial computer.

Finally, it discusses the design procedure of a dual-rate MPC based on the single-rate MPC.

**Chapter 4** includes details of the experimental setup, the connection between the model plant and the controller and different modes of setup used for the experiments. The advantages of Siemens PCS 7 DCS are also presented. Implementation of the dual-rate controller with brief descriptions of function blocks used in Siemens SIMATIC Manager is finally discussed.

**Chapter 5** explains different test scenarios at the beginning. Afterwards, it includes experimental results and evaluations of each scenario in details. In the discussion section, some schemes are presented on the application of the dual-rate control in fieldbus based distributed control system for network bandwidth and energy conservation.

And conclusions are drawn in Chapter 6 together with suggestions for future work.



## Chapter 2

# LITERATURE SURVEY

### 2.1 Distributed Control Systems and Evaluation of Communications

Communication technology in process industry developed mainly in four stages in last four decades. These steps start from Direct Digital Control through field devices to distribution of field devices and tasks for supervisory control. The first step started in process control industry by the introduction of star topology where all the field devices are connected to a single mainframe computer in the control room. The mainframe computer is in charge of all the control and supervisory tasks. To achieve this goal, the mainframe computer transfers all the data back and forth from and to the field devices through traditional analog point-to-point methodology. Figure 2.1 shows an example configuration of Direct Digital Control (DDC), also known as the centralized control, used in the 1960s [5]. One of the pioneering DDC is Foxboro PCP-88 [6].

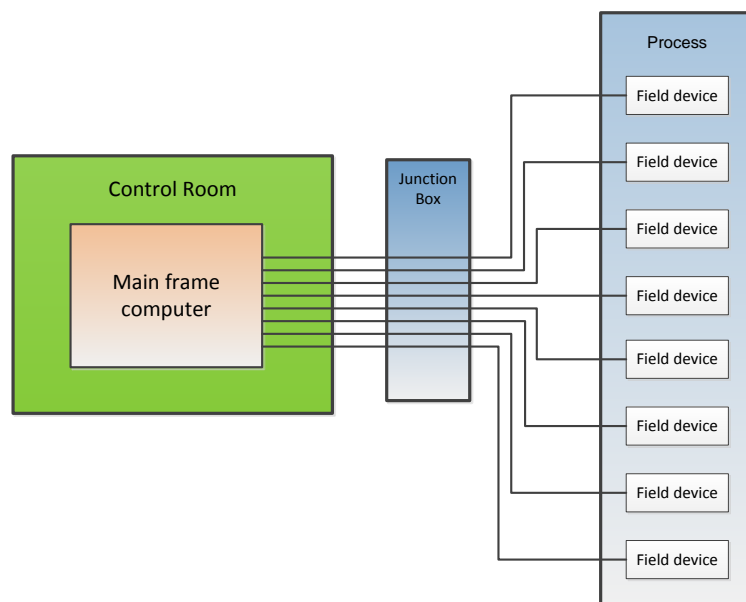


Fig 2.1: Direct Digital Control through field devices

In the second step, the process industrial controllers focused on dividing the supervisory task and control task of the process in separate controllers. These separate controllers have their own set of field devices that are attached to perform separate tasks in a point-to-point way. This architecture is known as Hierarchical Architecture and Fig 2.2 shows an example of such a system.

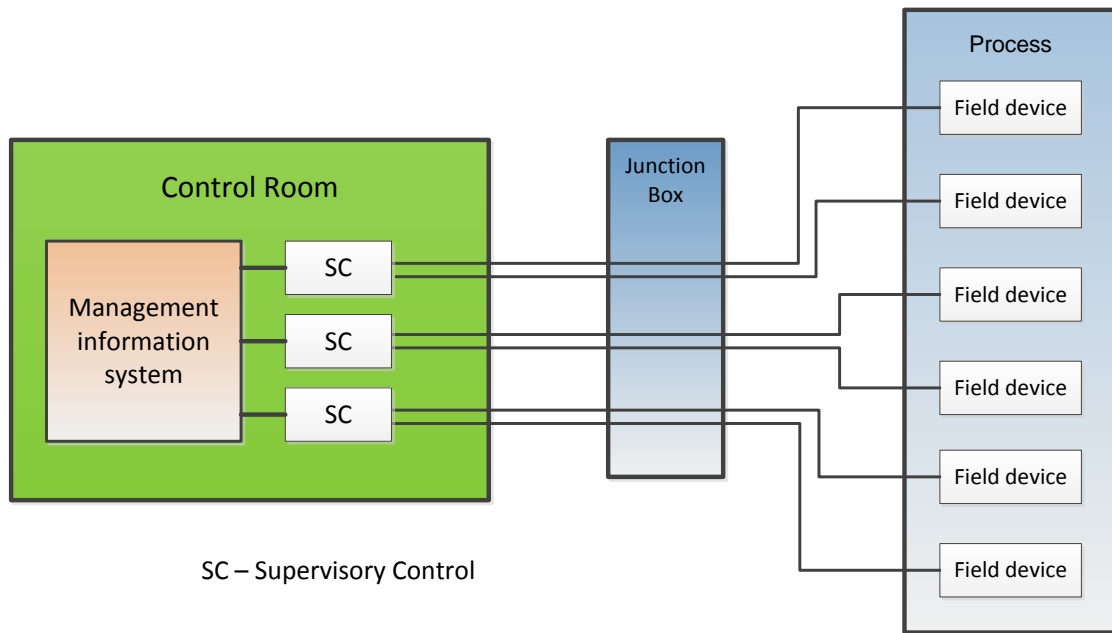


Fig 2.2: Hierarchical process control architecture

In Figure 2.2, it can be seen that each controller is separately connected to a set of field devices in the plant for performing either control or supervisory tasks and those are connected to a computer known as management information system (MIS). These separate controllers have been placed along with the same MIS computer in a single control room same as before in the early 1970's. This design provides some level of hierarchy and distribution of the tasks in different controllers and makes the system less prone to failure.

Due to the advent of integrated circuit (IC) technology, the third step of evolution happened in the process industry in the form of DCS technologies. Smaller and faster electronics led to more distribution of tasks in process control industries. Performance improvement as a result of the new technology also contributed to lower cost of the

overall system. Introduction of digital communication technology in DCS, started as a way of making things more distributed. Moreover, the controllers have been moved from the control rooms and placed near to the field devices as shown in Fig 2.3 which reduces the wiring complexity and cost. But still, the field devices are connected point-to-point with the controllers through analog communication. The controllers close to the devices are known as local controllers and are connected to the supervisory controller and operator console in the control room via serial digital communication links.

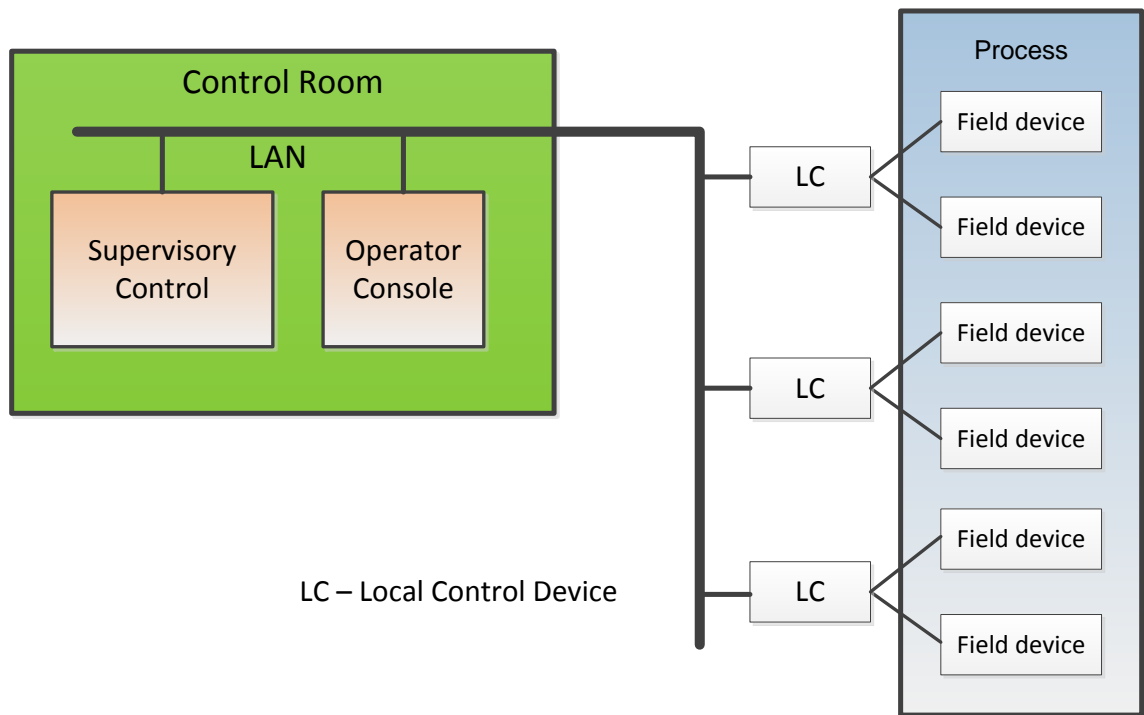


Fig 2.3: Control devices distributed near field devices

In 1975, Honeywell introduced such type of DCS based on digital serial network, known as TDC 2000 [6].

Later in early 1980's, the digital computer technology boomed by VLSI technology introduced more and more devices in the market. To make processes safer and more efficient and running with less maintenance, designers introduced digital network technology to the process control. This new technology has also reduced a lot of wiring complexity and has made system diagnostics easier and faster [7], [8]. The digital

network technology that connects the field devices to controllers is known as Fieldbus as in Fig 2.4.

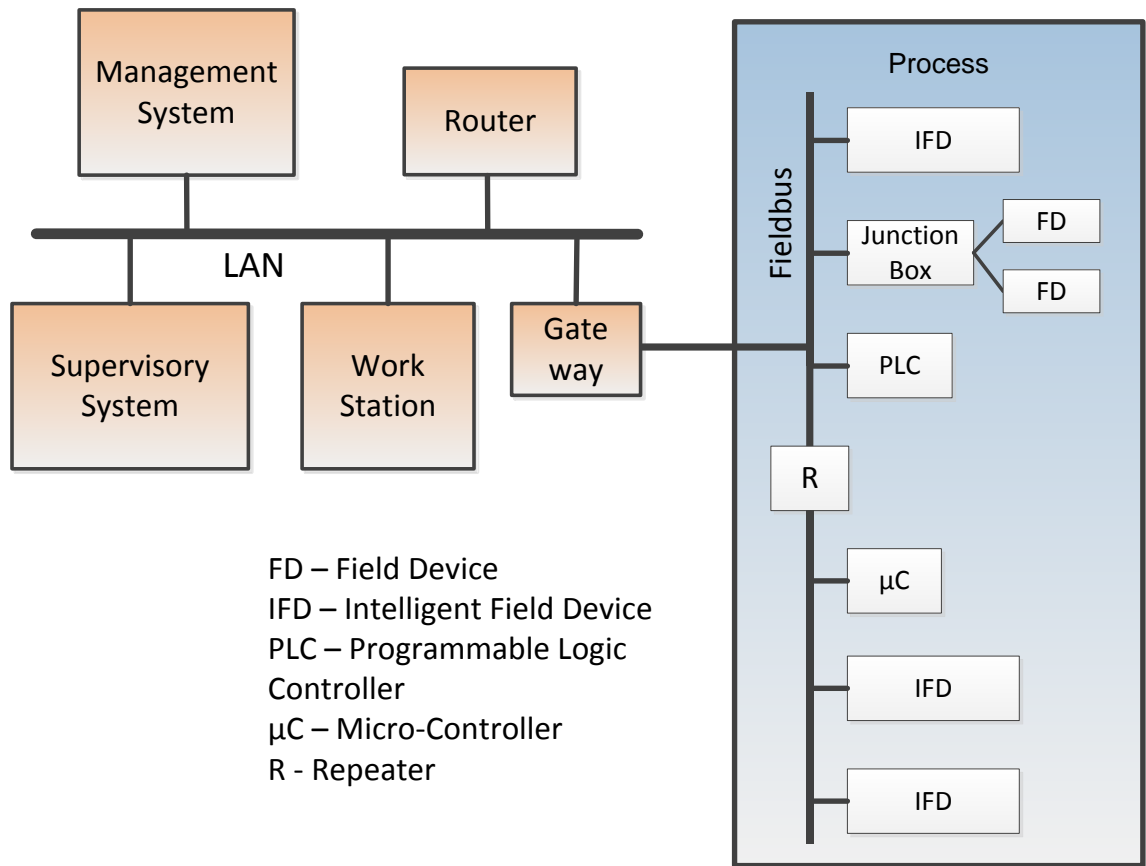


Fig 2.4: Distributed control system through LAN and fieldbus

## 2.2 Fieldbus Technology

The word Fieldbus is made up with two terms, *Field* and *Bus*. In industrial context the *Field* is used as an abstraction of the physical plant level and the *Bus* is a physical communication medium commonly known in computer industry which electrically or optically transfers data between the devices connected to it. Fieldbus has originally been introduced to replace the point-to-point 4-20mA analog communication between the field devices and the controllers by a single digital link through which all information can be transferred serially in a time shared or multiplexed manner. For data transmission,

fieldbus uses small data packets which are transferred serially through the physical medium based on a special protocol for effective information transfer. The protocol smartly abstracts the serial communication that takes place through the physical medium for all devices and connects devices as if they are parallelly connected to its controller separately, similar to point-to-point topology [9].

## **2.3 Adoption of Fieldbus Technologies in Distributed Control Systems**

In last three decades, process industries have invested large amount of efforts in developing DCS and associated instrumentation technologies. The obvious advantages of DCS in process industries have made huge satisfied customer-base. In traditional DCS, the primary means of communication between controllers and its field devices i.e. sensors and actuators is 4-20mA current loop for its advantages over voltage loops. Prior to the advent of 4-20mA signaling mechanism, historic industrial control systems has been using pneumatic signaling schemes of 3-15 psi and has been a milestone improvement in industrial control field [5].

In modern age, now the trend is to use digital technologies. Industrial communication in control application is no exception. The realization of the fact that traditional 4-20mA analog signal cannot carry any information other than natural control signal; industry is searching for alternative communication schemes capable of handling more information. Speed of communication over 4-20mA analog signal is relatively slow as well, which also pushes the industry for a faster digital alternative.

Where analog communication uses point-to-point topology between controller and field devices, fieldbus on the other hand is a digital communication technology which provides two-way, multi-drop topology between smart devices connected to digital control system. Fieldbus truly accelerated digitization in process control world and brought superior functionality in the field. This has changed the perception of process control and brought

openness and standardization in measurement, instrumentation and diagnostic with greater throughput.

But due the involvement of risks and financial investment in existing analog setups, adoption of new digital technologies are slow even if they can see the advantages of new fieldbus technologies. At the beginning, customers preferred a step-by-step upgrade of parts of DCS with new fieldbus technologies.

## **2.4 HART: Introduction of Digital Communication over Analog Channels**

The HART communication technology is a very interesting extension to the old 4-20mA analog technology. As the trend towards higher information transfer from devices to controllers grew for safer, more powerful, faster and easily maintainable process control applications, strong need for a new type of communication technology is felt by industry to meet these demands of the customers. But along with that, customers needed step by step upgrade of existing equipment for reducing risks and utilizing investment made. HART is an immediate response to offer superior digital communication over traditional 4-20mA links.

HART is a powerful communication technology based on analog medium that exploits the full potential of smart digital field devices. It preserves the legacy of 4-20mA signal but extends the system capability by the introduction of two-way digital communication between controller and smart digital field instruments [10]. Due to its capability of using analog communication channels without any modification to the field wiring, HART has become very popular in DCS industry. It has huge product base and all types of smart field devices used in process application are available in the market with capability of HART communication.

The HART communication protocol is based on BELL 202 communication standard which operates on Frequency Shift Keying principle. The digital signal is represented by two frequencies as 1,200 Hz and 2,200 Hz for bit 1 and 0 respectively. As it works over traditional 4-20mA analog channel, it superimposes the sine waves of these two frequencies over the direct current (DC) analog signal. The original analog signal is not affected as because the average of the two FSK frequencies remains zero all the time. Considering communication frequency, digital data has a throughput of 2 to 3 updates per second without disrupting the 4-20mA analog signal. A typical HART signal is shown in Fig 2.5.

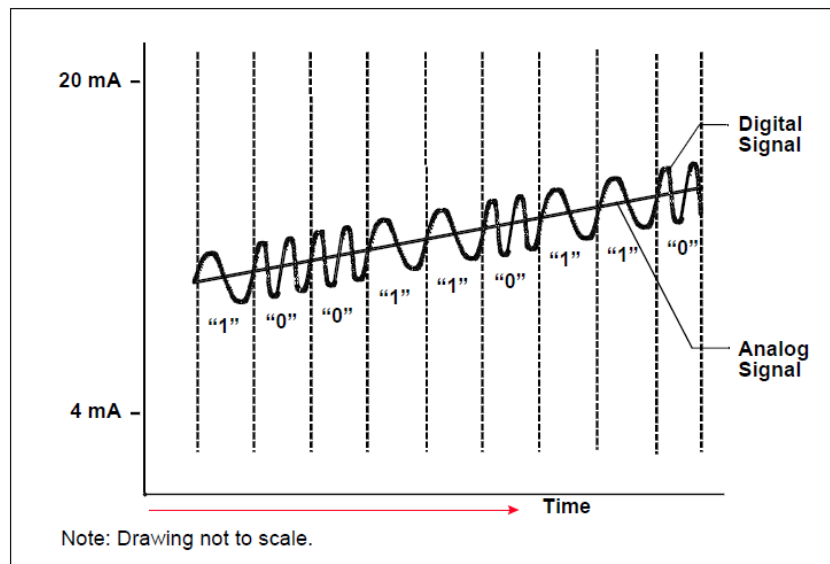


Fig 2.5: HART signal [11]

## 2.5 Foundation Fieldbus

After Highway Addressable Remote Transducer, various fully digital fieldbus technology is developed and the most prominent of them worth discussion is Foundation Fieldbus or FF. Foundation Fieldbus is an advanced form of fieldbus which evolved to become one of the standard digital fieldbus communication technologies in instrumentation and process control industry. It is different from other digital communication protocols

because it is not only providing digital data transfer schemes rather resolving other process control application needs. It is a powerful protocol in modern industry.

Foundation Fieldbus is an all-digital, serial, two way communication system that provides interconnections between field devices, such as sensors, actuators and controllers. In such field devices, the whole control strategy is distributed through FF. As already mentioned, it is not only a data communication protocol for devices, it also contains function blocks that can reside on FF devices containing its own microcontroller and can execute the distributed function blocks right on the devices in the field to make Control In the Field (CIF) possible. This makes distributed control system design much more flexible. According to the users need, the whole control system can be segregated in smaller control loops through FF's function blocks running on the devices i.e. within sensor and actuator as well as in controllers. Fig 2.6 shows a simple sketch of a control loop.

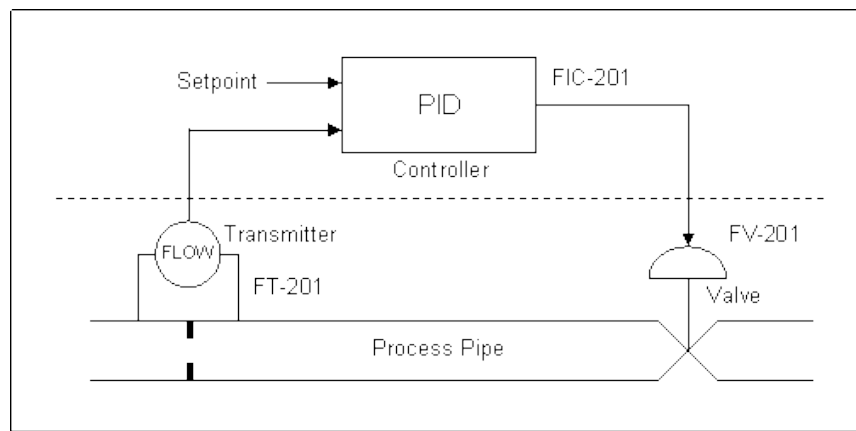


Fig 2.6: A flow control loop [12]

This drawing describes a simple flow control loop using three instruments. It mainly shows how each component in the control loop is connected and their control functions as follows:



- 1) Flow transmitter, it is an orifice based differential pressure instrument for flow measurement through a pipe where the pressure drop is the square root of water flow rate
- 2) PID controller with a set-point value and the valve control signal
- 3) Control valve to regulate the flow through the pipe

### 2.5.1 Control in the Field

When the above simple control drawing is transferred into the digital world, it becomes a diagram as shown in Fig 2.7. The dotted line separates the physical components from the software components, which are stored in the memory of one or more of the smart microprocessor based devices. The AI and AO modules above the dotted line are the software drivers for the hardware A/D and D/A converters which reside on the Transmitter and the Valve under the dotted line. In CIF, the Proportional Integral and Derivative controller, AI and AO module function blocks (with their hardware components) are distributed in the field devices.

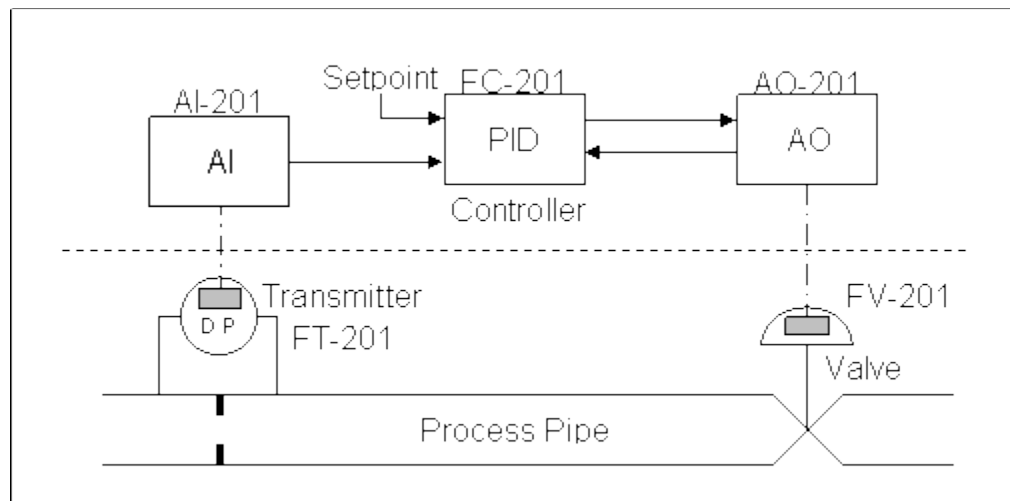


Fig 2.7: Control in the field [12]

Above the dotted line, three rectangles represent software function blocks in a function block diagram. Each block has a tag for identifying in software as well as to the operator. Under the dotted line are the physical devices, here the transmitter and the control valve

have transducer blocks shown in smaller grey boxes. Transducer blocks contain information for calibration and other use for physical sensors and actuators and internally connected to the input or the output function blocks. The arrow signs indicated the direction of data flow or communication paths. It may be a fieldbus or a conventional analog signal which is decided later after completion of the function block design. The controller is the main focal point of a control loop. If the controller is used in manual mode, the operator can change set-point, read process variable or change the output.

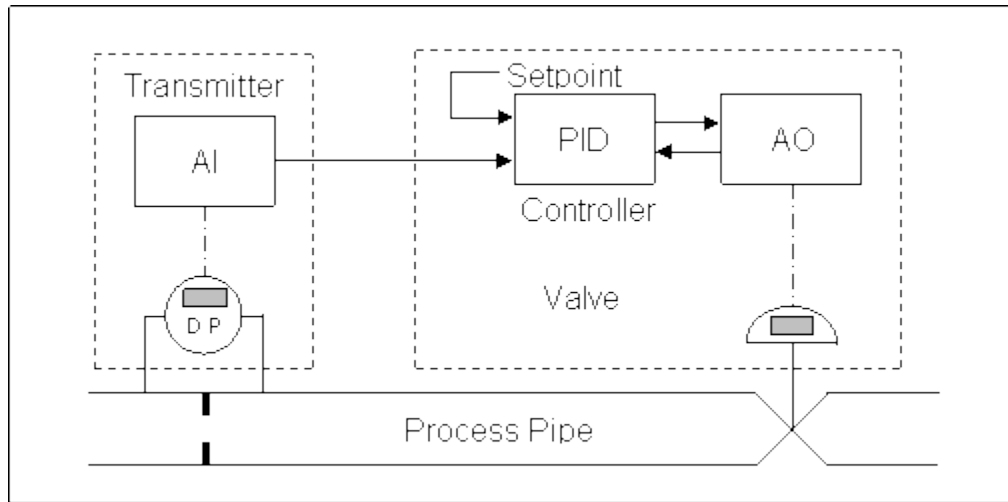


Fig 2.8: Distribution of function blocks in field devices [12]

Figure 2.8 shows that the control functions as software modules can reside in the field devices. In this design step, function blocks are assigned to the devices which determine the amount of required communication data through fieldbus. In Figure 2.8, the software function block AI is included in the transmitter and the AO function block is put on the valve block. The PID function block could be put on either the transmitter or the valve. In this case, it is included in the valve along with the AO block. This is how general control in the field application is done through FF.

## 2.5.2 Sequence of Execution and Scheduling in Foundation Fieldbus

In Distributed Control Systems using Foundation Fieldbus technology, scheduling the sequence of execution of various function blocks and data communication over the fieldbus is of utmost importance for proper functioning of the control system and efficient utilization of the shared communication resource. Control systems need process data to be cycled periodically. This cycle time must be short enough as compared to the process time constant. In process control application, usual cycle time is 1 second. In some cases, it can be faster up to 100ms or may be slower depending on the system dynamics and applications. A fieldbus communication cycle for periodic data handling is known as a macrocycle. It is possible to use shorter cycle times within macrocycle which is integer divisor of the macrocycle.

Foundation Fieldbus H1 is based on existing twisted pair technology as physical medium and its speed is limited to 4,000 bytes per second at a rate of 31,250 bits per second. There is Foundation Fieldbus HSE based on high speed Ethernet communication, but it is not feasible for long distance plant distribution in harsh industrial environment.

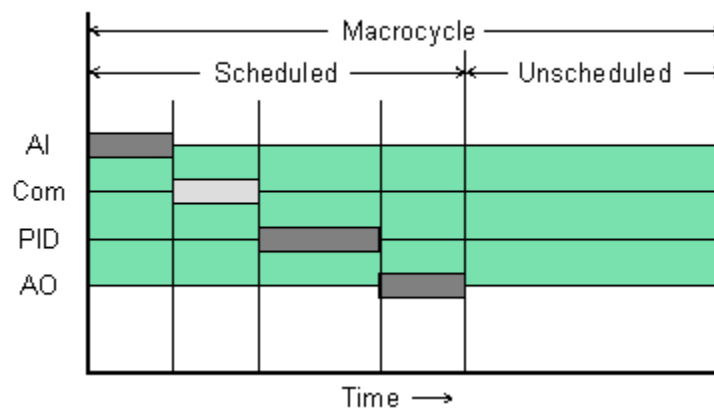


Fig 2.9: Foundation Fieldbus macrocycle task sequencing and scheduling in LAS

Link Active Scheduler which is the heart of Foundation Fieldbus is a software component in charge of sequencing and scheduling different tasks of transferring messages according to priority. In a FF network, a Link Master device has the LAS which coordinates and

manages the link scheduling and maintaining execution sequence of various function blocks. The predefined schedule must be provided to the Link Master for telling other devices to send or receive messages. There may be more than one LAS capable devices connected on a fieldbus but only one can act as a Link Master actively. If one fails, the other one can take control of the fieldbus and maintain scheduling work. The actual schedule is determined at the time of system design by configuration- software calculating the time required for executing various function blocks and determining their execution priority.

Figure 2.9 is an example macrocycle of the flow-rate controller example in Fig 2.8. The AI function block resides on the transmitter device while the PID and AO function blocks resides on the valve device. LAS schedules adequate time for function blocks to execute within the device which is done in a sequential manner and in between allocates enough time for issuing Compel Data command to the device willing to broadcast data to the devices which are configured to receive the data. In this example, at first, the AI block executes on the transmitter device. Once it is complete, the Compel Data command is issued by LAS indicating the AI block to broadcast its data to the PID block on the valve device. As PID and AO block resides on the same device, data transfer from PID to AO is done directly without any fieldbus communication.

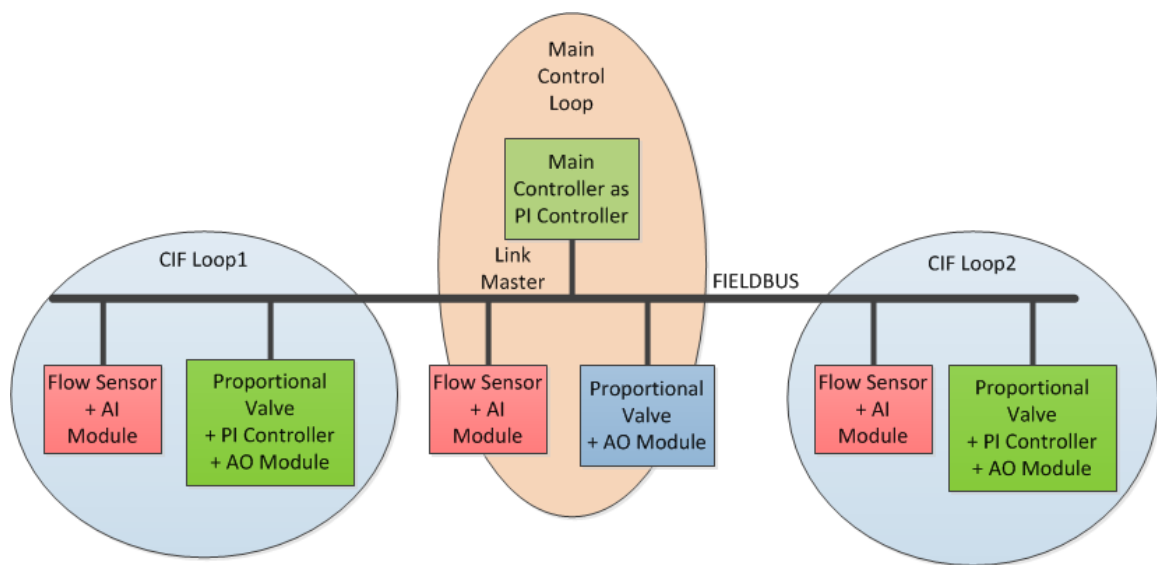


Fig 2.10: Multiple control loops on fieldbus and Link Master

In between the cyclic data transfer, LAS also keeps enough room for unscheduled data transfer between devices for manual operation, configuration, calibration, maintenance etc. LAS grants access to a device for using the fieldbus, to send message to other devices by issuing a pass token (PT). When the device gets a PT, it is allowed sending message until it is finished, or as long as the time allocated for unscheduled data transfer expires, whichever is shorter.

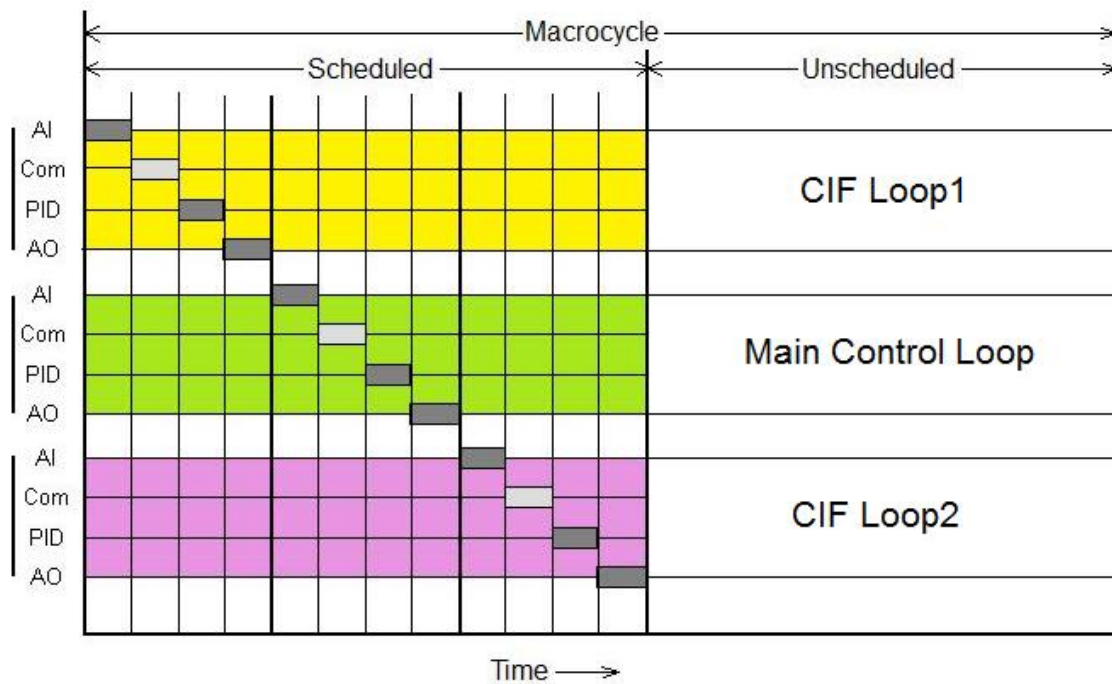


Fig 2.11: Multiple control loop task scheduling in LAS

Figure 2.10 shows a multiple control loop configuration on fieldbus including two CIF loops. The main controller also acts as a Link Master and is in charge of scheduling the fieldbus. The main controller is connected to a sensor device and a flow control valve actuator forming the main control loop. The two other control loops are CIF type and uses the same fieldbus for control data transfer. CIF Loop1 contains a sensor device and an actuator device. The actuator device contains the controller of the CIF Loop1 as well. CIF Loop2 has a similar configuration. Figure 2.11 shows the LAS schedule for the multi-loop control system. It shows timings of sequential executing of loop components of the three loops in the scheduler [50].

Along with its advantages, fieldbus introduces some limitations on the system design and performance. Since fieldbus is a shared media for communication among many devices, each of them gets limited access to this shared media during communicating with others. In addition, as the data bandwidth of a fieldbus is fixed, a limited number of devices can be efficiently used in a slow fieldbus loop. Addition of more devices beyond that limit may cause bandwidth congestion [54]. One way to improve this situation is to use an advanced multi-rate control algorithm. By resorting to a multi-rate controller, sensor and actuator sampling-rate can be reduced when the system reaches steady-state which can contribute to fieldbus bandwidth conservation. Moreover, energy conservation is not taken into account in current DCS technologies, which can be achieved by putting the field devices in sleep-mode while they are not communicating being idle.

## **2.6 DIGITAL CONTROL SYSTEMS**

### **2.6.1 Basic Structure of Digital Control Systems**

The basic structure of a digital control system or sampled data system is shown in Fig 2.12. The system or process (or plant) to be controlled is continuous time system e.g. a robot, a motor or an electrical power plant. In a digital control system, the heart of the controller is a digital computer. The continuous time signal is converted by an A/D converter to a discrete-time signal which is generally synchronized with a predefined clock frequency. The A/D converter converts the discrete-time signal to binary digits to facilitate the digital computer to perform binary operations on numbers representing the analog signal. On the other hand, a D/A converter, in contrast, converts the discrete-time signal from the computer back to a continuous-time signal and sends to the plant actuator. To do that, the D/A converter includes a hold circuit for signal conversion from discrete-time to continuous-time [33], [34].

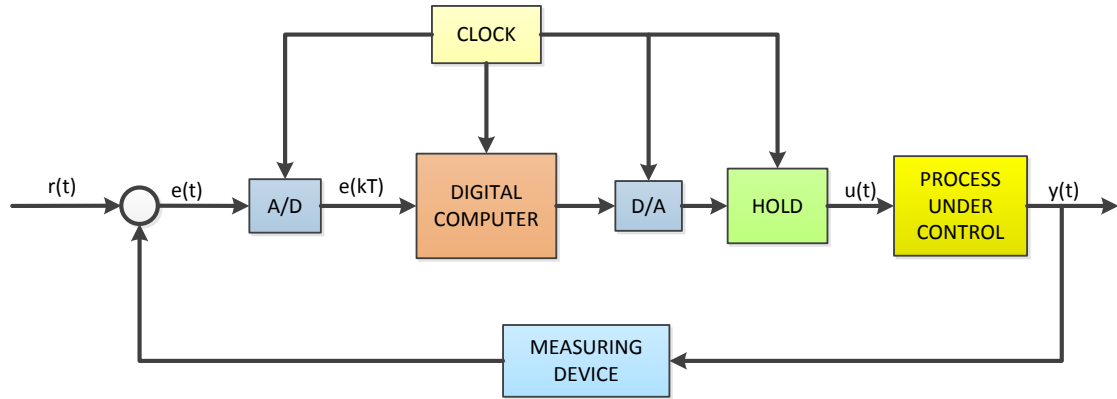


Fig 2.12: Block diagram of a digital control system

According to the design specifications, the controller design can range from simple schemes to complex schemes [21]. It may work as a programmable logic controller (PLCs) to perform simpler logical tasks or it may control more complex schemes to use the error signal  $e(kT)$  to produce suitable input  $u(t)$  to control the plant. The control input  $u(t)$  must be such that it makes the closed loop system to satisfy the desired specification.

One issue in the implementation of a digital controller is in the development of a computer program i.e. a software to be used on a computer hardware [34]. There are several distinct advantages of digital controllers over their analog counterparts, some of which can be listed as follows:

- 1) As digital controllers are based on computer software, it has tremendous flexibility in modifying its control functions, such as tuning control variables and addition of new schemes and algorithms etc. In contrast, classical analog controllers, as it is designed and constructed using active and passive components, modification to the controller is expensive because it requires to modify the physical hardware structure and components.

- 2) Data processing is much simpler because of the use of a computer. Powerful mathematical functions can be implemented relatively easily in the computer that makes complex computation much faster and convenient.
- 3) Considering performance, digital controllers are superior over analog controllers in terms of the following measures:
  - (a) Reliability
  - (b) Sensitivity
  - (c) Drift effect, and
  - (d) Internal noise
- 4) Due to the availability of inexpensive but powerful microprocessors and digital components, digital controllers are more cost effective than analog controllers in practice.
- 5) Use of VLSI technology in digital controllers makes them more compact in footprint as compared to their analog counterparts.

Despite of the above mentioned advantages, digital controllers have certain disadvantages as compared with analog controllers. The most noteworthy disadvantage is that it introduces errors during sampling and quantization process.

Digital computer based control systems suffer from limited CPU and memory resources. Generally speaking, the faster the sampling-rate, the better the control performance becomes. But with the faster sampling-rate, the more CPU resource is consumed as well. Due to its limited resources in digital control system, the sampling-rate has to be optimized to free up CPU and memory resources for control-algorithms, communication and other purposes. Furthermore, one controller may have multiple control loops within a process running in parallel. To cope with these limitations, sampling-rates of multiple control loops are determined based on a minimum possible value that guarantees desired performance of the control loops within a process. Different components in a process can



have different dynamic characteristics. Some are faster, and the others are slower. Depending on the dynamic characteristics, different control loops can have different sampling-rates and it is not feasible to run all the loops in an identical rate due to resource constraints. Thus, multi-rate control scheme is very useful in such a situation.

### **2.6.2 Multi-rate Sampling in Digital Control Systems**

There are several reasons for using a multi-rate sampling scheme in digital control system:

- It is often unrealistic or even sometimes impossible to sample uniformly all physical signals at one single rate in complex, multivariable control system. Multi-rate sampling-rates have to be used in such situations.
- Generally speaking, closer approximation of digital signal to its analog counterpart can be achieved when faster sampling and hold is used. But the cost goes higher with faster A/D and D/A conversion. More reasonable trade-offs between implementation cost and performance can be obtained for signals with different bandwidths by using A/D and D/A converters at different rates.
- In general, multi-rate controllers are inherently time-varying in nature. Thus, it is possible to achieve improved performance that may not be achievable with a single rate control system; for example, gain margin improvement [14], [15], simultaneous stabilization [14], [16] and decentralized control [13], [17], [18].
- Multi-rate controllers are by nature more complex than single-rate ones; but often they are periodic in a certain way, they can be implemented on digital computers via difference equations with finite number of coefficients. Therefore, multi-rate controllers do not violate the finite memory constraints similar to its single-rate counterpart.

### 2.6.3 Multi-rate Sampled-data Systems

A typical multi-rate sampled-data control system is shown in Fig 2.13. The output channels of the multi-rate controller are sampled at different rates [30], [31]. Here, the first output  $y_1(t)$  is sampled in every  $m_1h$  second and similarly the second output is sampled in every  $m_2h$  second and so on. If the number of plant output is  $p$ , then it can be shown that:

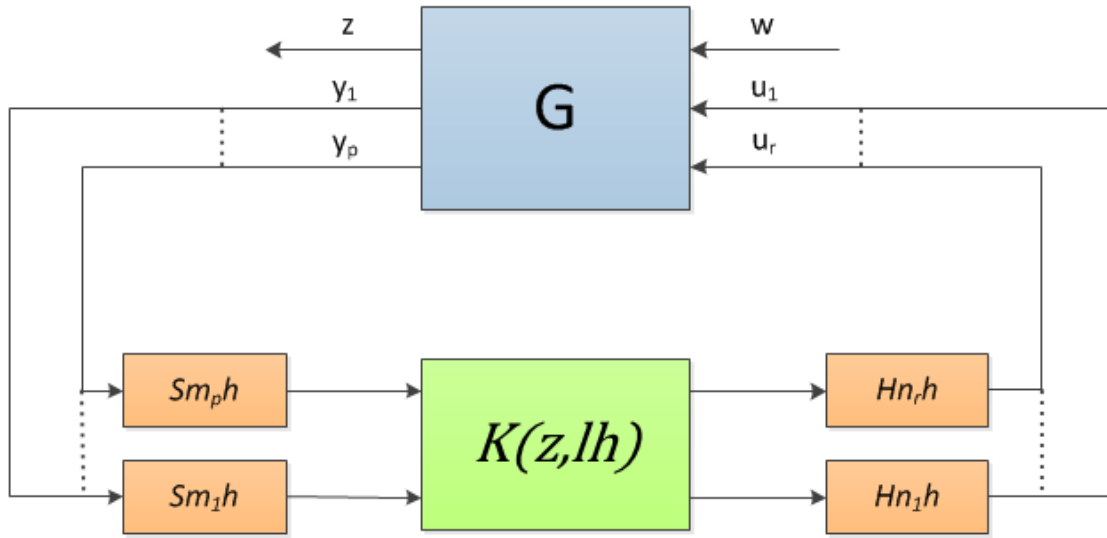


Fig 2.13: Multi-rate sampled-data system

$$y(k_{m1}) = S_{m_1h}y_1(t), \quad (2.1)$$

$$y(k_{m2}) = S_{m_2h}y_2(t), \quad (2.2)$$

$\vdots$

$$y(k_{mp}) = S_{m_ph}y_p(t). \quad (2.3)$$

Similar to the plant outputs, different sampling-rates can be used to generate different control signals, for example, the first control signal is sampled at every  $n_1h$  seconds, the second at  $n_2h$  and so on. If there are  $r$  number of control signals, then, it follows:

$$u_1(t) = H_{n_1h}u(k_{n1}), \quad (2.4)$$

$$u_2(t) = H_{n_2h}u(k_{n2}), \quad (2.5)$$

$\vdots$

$$u_r(t) = H_{n_r h} u(k_{nr}), \quad (2.6)$$

By assuming that, the least common multiple of  $(m_1, \dots, m_p, n_1, \dots, n_r)$  and the greatest divisor are 1 and  $l$  respectively. That makes the sampling period of the fast-rate and slow-rate controllers  $h$  and  $lh$  respectively.

In general, a multi-rate controller can be viewed as a periodically time-varying discrete-time system with period  $l$  [37]. But if a multi-rate controller is considered in one repetition period of  $lh$  second, with lifted inputs and outputs, it becomes a higher dimension discrete-time LTI system with sampling period  $lh$ . The LTI multi-rate controller can then be denoted by  $K(z, lh)$ .

#### 2.6.4 Example of a Multi-rate Control System

A good example of a multi-rate control system is a position servo control of PMSM or DC motors. Where different control loops are cascaded one over the other. Figure 2.14 shows a block diagram of a position servo controller for a motor [32], [38]. The fastest loop is the current control loop which controls the torque of the motor. A current measuring device measures the current going into the motor and isolates it from the motor voltages and processed by the signal conditioners to be fed to the digital computer. The sampling period is assumed to be  $T_1$ . The current loop or the torque control loop is preceded by the speed control loop and has sampling period of  $T_2$  where  $T_2 > T_1$ . An absolute encoder or a quadratic encoder is used to sense the speed for the relative angular position change on the encoder. Both angular speed and angular position can be measured through the encoder.

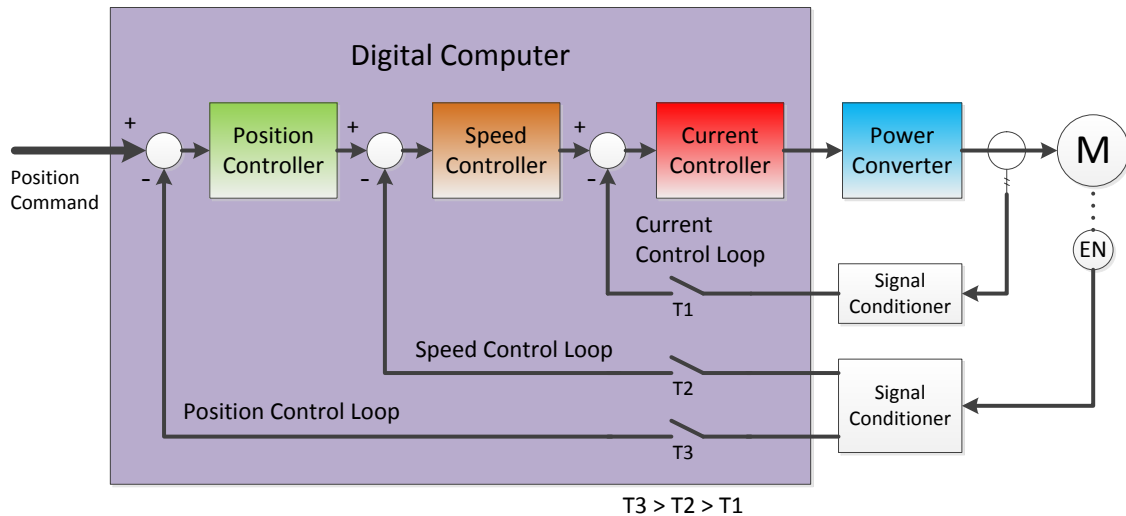


Fig 2.14: Multi-rate position servo controller

The speed controller is preceded by the position controller which has the longest sampling period of  $T3$  where  $T3 > T2$ . The position controller takes position feedback from the encoder. As can be seen, a position servo controller is an example of a true multi-rate controller as it has used different sampling-rates  $T1$ ,  $T2$  and  $T3$  in its respective control loops for making efficient use of CPU and other resources.

## 2.7 Model Predictive Control

Originated in the late seventies, Model Predictive Control (MPC) has been developing considerably since last few decades. It now has become one of the most effective control strategies in modern process control industries. The term Model Predictive Control is not bonded to a specific control strategy, rather it signifies an ample range of control methods that explicitly make use of the process model for obtaining the control signal based on a minimized objective function. Thus, these design methods lead to the design of linear controllers with practically the same structure, presenting adequate degrees of freedom [19]. The features that are common to all the predictive control family are basically as follows:

- Predicting process output at future time instants (horizon) using explicit model of the process
- Use of an optimization cost function over the prediction horizon
- Receding strategy, uses past controls for predicting the future ones i.e. based on the past instants, the horizon is displaced towards the future [35], [36].

The difference between various MPC algorithms is mainly on the models used to represent the process, various noises and types of cost function to be minimized. As this type of control scheme is by nature very open, it has attracted much development work and is widely used for academic research and by industry [19], [23]. Many applications have been successfully carried out based on predictive control schemes, which are not only limited to process industries, but also include diversified processes ranging from robot manipulators to medical instruments e.g. clinical anesthesia [52], [53]. MPC has many advantages over other control methods which can be summarized as follows:

- In industry, being an intuitive method, it is particularly advantageous to the staff in charge of operating the process with limited knowledge of control
- It covers a wide range of applications and variety of processes, from simple dynamics to complex systems involving long time delays or unstable systems with non-minimum phase characteristics
- It is easy in dealing with multivariable control system scenario
- Intrinsic compensation for dead times is one of its features
- It allows natural introduction of feed forward control for compensation of measurable disturbances
- MPC is easy to implement using linear control law
- Constraint handling is conceptually simple in MPC and can be systematically introduced during the design process

### 2.7.1 MPC Strategy

The methodology for MPC family can be characterized by the following strategies represented in Fig 2.15 (a).

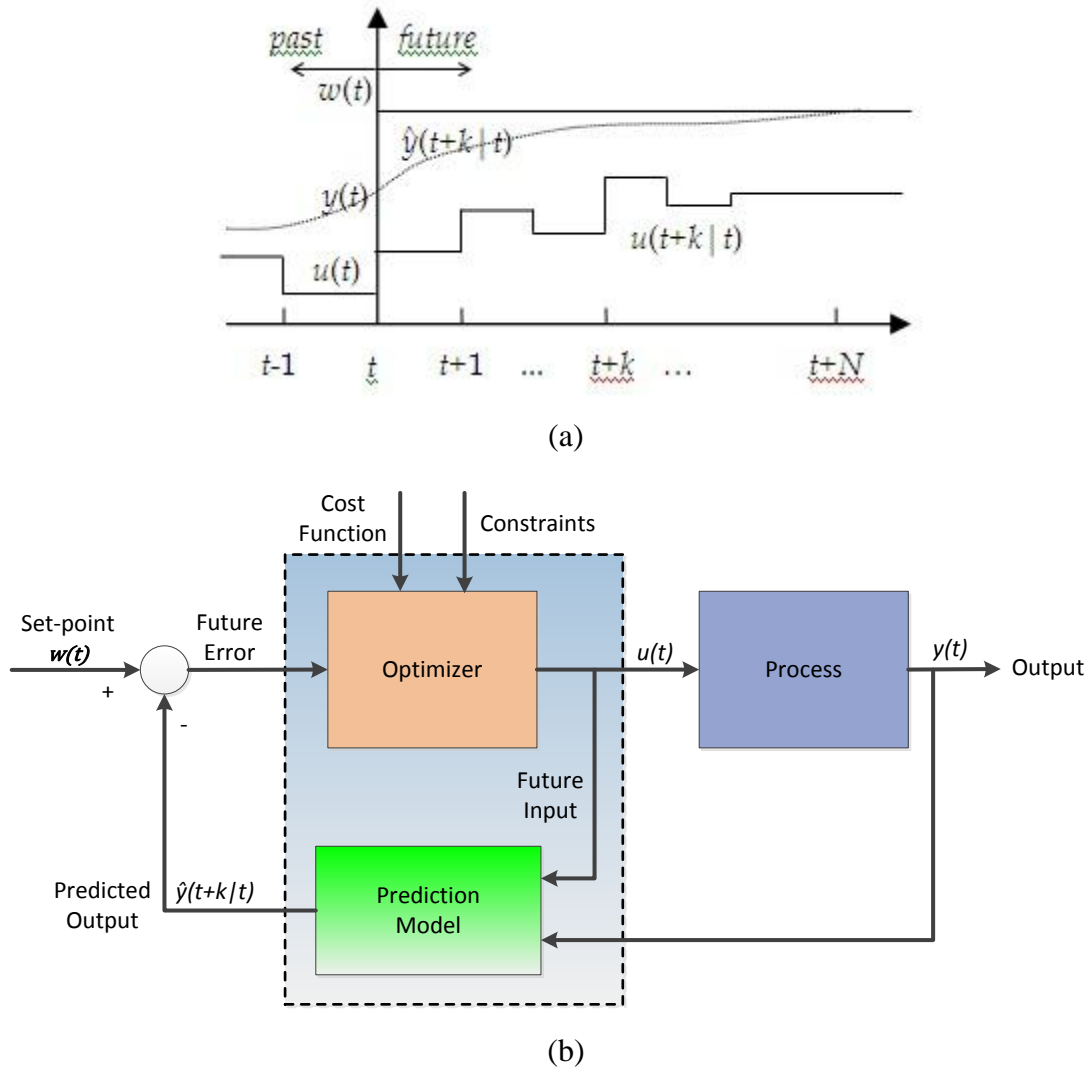


Fig 2.15: (a) Model Predictive Control strategy (b) MPC block diagram

1. As shown in Fig 2.15 (a), future outputs are predicted at each instant  $t$  using the model of the process for a determined horizon  $N$ , called the prediction horizon. These predicted future outputs  $y(t + k | t)$  for  $k = 1 \dots N$  depending on known values up to instant  $t$  (past inputs and outputs) and also on future control signal  $u(t + k | t)$ ,  $k = 0 \dots N - 1$ , which are to be sent to the controller for calculation [19], [35].

2. In order to keep the process as close to the reference trajectory  $w(t + k)$  (which can be a set-point itself or its close approximation), by optimizing a determined criteria, a set of future control signals can be calculated. This criterion generally takes form of a quadratic function of errors between the predicted output signal and the predicted reference trajectory. In most cases, control effort is included in optimization cost function. If the criterion is quadratic and the model is linear with no constraint, an explicit solution can be obtained; otherwise an iterative optimization method has to be used. Some assumptions regarding the future control law structure are also made in some cases.
3. The control signal  $u(t | t)$  is sent to the process and the next calculated control signals are rejected because  $y(t + 1)$  is already known at the next sampling instant and step 1 is repeated with this new value and all the sequences are made up to date. Thus, using receding horizon concept,  $u(t + 1 | t + 1)$  can be calculated.

Figure 2.15 (b) shows the basic structure for implementing this strategy. For the prediction of future plant outputs and the proposed optimal future control actions, a model is used based on past and current values. These actions use the optimizer and its cost function for the calculation of optimal value taking constraints into account as well.

For utilizing this methodology, the process model plays a decisive role in the controller. The chosen methodology must be relatively simple for implementation and it must be able to precisely predict the future outputs. There are many types of models used in various formulations in MPC which are not unique, rather a set of different methodologies [36].

It is noteworthy to draw a parallel between MPC strategy and the control strategy used in driving a car. In a driving scenario, the driver knows the desired reference trajectory for a finite control horizon and by taking this into account the driver makes the decision as which control actions to take in order to follow the desired trajectory. In a receding

horizon control, only the one action is taken at each instant and the procedure repeats for the next control decision.

## **2.8 PREVIOUS WORKS ON MULTI-RATE CONTROL SCHEME**

### **2.8.1 Multi-rate LQR Controller**

Before Model Predictive Control, in 1960's, LQR/LQG became immensely popular in process industries [22]. In industrial applications, a successful controller must consider the constraint limitations while in operation as the maximum throughput is often achieved near the constraint boundary [20]. But in LQR formulation no constraint is considered. Also accurate industrial process model is often hard to obtain as it changes with time. LQR application is limited due to its unmodifiable quadratic cost. This problem of LQR is solved by the introduction of MPC through its constraint optimal control in finite and receding horizon.

$$\text{Simple MPC} = \text{LQR} + \text{constraint} + \text{finite Horizon}$$

Previous works have been done on designing Linear Quadratic Optimal Controller (LQR) based multi-rate controllers [22]. In the design, it is assumed that the sampling-rates involved are of integer ratios. For designing the sampled data control system, a suitably high sampling-rate is chosen in the process model. The output sampling-rate is determined upon the different approximate time constants of the sensors and actuators. The advantage of such controller is that it can easily be implemented in a computer in terms of gains without much performance degradation as compared to a high sampling-rate single-rate controller [24]. Multi-rate control techniques are predominantly used in aerospace industries where widely differing bandwidth and computational constraints exist simultaneously.

Some work has been done on multi-rate LQR controller design on motor control application [25]. Some notations and assumptions are used e.g.  $T$  is the short time period,



$T_1$  is the long time period,  $k$  is an index denoting number of short time periods.  $L$  denotes the integer, where  $L = T_1/T$  and  $i$  is an index denoting number of long time periods  $T_1$ .

A system is considered linear time invariant (LTI) in every  $T$ . However, the system may be time-varying in between every  $T$ . The system is assumed to be periodic in every  $T_1$ . The approach of modeling the system is representing the LTI difference equations over each  $T$ . The difference equations can be written as follows:

$$x_{k+1} = F_k x_k + G_k U_k, \quad k = 1, 2 \dots L \quad (2.7)$$

where,  $F_k$  and  $G_k$  are constant matrices of order  $(n \times n)$  and  $(n \times m_k)$ , respectively.  $x_k$  is a  $(n \times 1)$  vector denoting the states of the multi-rate system and  $u_k$  is a  $(m_k \times 1)$  vector denoting the input to the system.

From the state transition equations of (2.7) the model of the system can be found as follows:

$$Z_{i+1} = AZ_i + BU_i \quad (2.8)$$

where,

$$Z_i = [x_1, x_L, x_{L-1}, \dots \dots \dots x_2]^T ; \quad U_i = [u_1, u_L, u_{L-1}, \dots \dots \dots u_2]^T$$

$$A = \begin{bmatrix} 0 & F_L & 0 & 0 & \dots & 0 \\ 0 & 0 & F_{L-1} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & F_2 \\ F_1 & 0 & 0 & 0 & \dots & 0 \end{bmatrix}; \quad B = \begin{bmatrix} 0 & G_L & 0 & 0 & \dots & 0 \\ 0 & 0 & G_{L-1} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & G_2 \\ G_1 & 0 & 0 & 0 & \dots & 0 \end{bmatrix}$$

The LTI block cyclic matrix equation (2.8) summarizes the behavior over  $T_1$ . The overall system equation under feedback can be presented as follows:

$$Z_{i+1} = (A + B.Kl)Z_i = Al.Z_i \quad (2.9)$$

where,

$$Kl = \begin{bmatrix} K_1 & 0 & 0 & \dots & 0 \\ 0 & K_L & 0 & \dots & 0 \\ 0 & 0 & K_{L-1} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & K_2 \end{bmatrix}$$

The quadratic performance index of an LQR controller  $J_{LQR}$  from equation (2.7) becomes

$$J_{LQR} = \left(\frac{1}{2}\right) \sum_{k=0}^{\infty} (x_k^T Q_k x_k + u_k^T R_k u_k) \quad (2.10)$$

where,  $Q_k$  and  $R_k$  is periodic with  $T_1$ . The quadratic performance index from equation (2.8) becomes

$$J_{LQR} = \left(\frac{1}{2L}\right) \sum_{i=0}^{\infty} Z_i^T Q Z_i + U_i^T R_k u_k \quad (2.11)$$

where,

$$Q = \begin{bmatrix} Q_1 & 0 & 0 & \dots & 0 \\ 0 & Q_L & 0 & \dots & 0 \\ 0 & 0 & Q_{L-1} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & Q_2 \end{bmatrix}; \quad R = \begin{bmatrix} R_1 & 0 & 0 & \dots & 0 \\ 0 & R_L & 0 & \dots & 0 \\ 0 & 0 & R_{L-1} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & R_2 \end{bmatrix}$$

$Q$  is positive semi-definite, and  $R$  is positive definite.

## 2.9 SUMMARY

Digital control systems have become a de-facto standard in process control industries. Digital controllers are based on computer software technology and have exceptional flexibility in modifying its control functions and tuning parameters and also have simpler data processing capability. Digital control systems are much more reliable than its analog counterpart as well. But despite of its numerous advantages, they can still suffer from limited CPU and memory resources. To make efficient use of the limited resources, multi-rate control algorithms are used in complex but resource limited processes. Some

processes in industries are so large that multiple controllers distributed throughout the process are connected and synchronized together with some communication technology which has given birth to modern DCS.

Distributed control systems have gone through gradual transformations in various stages in the last four decades. Modern DCSs use digital fieldbus technologies to reduce design and implementation cost along with performance improvements in terms of diagnostics and maintenance, operation and signal integrity. Traditional 4-20mA analog communication has been replaced by mixed signal HART and other digital fieldbus technologies e.g. Profibus, Foundation Fieldbus etc. Foundation Fieldbus has introduced advanced features, such as CIF to distribute the control loops and reduce main CPU load. Modern process controllers are fairly complex and powerful machines. It is now possible to implement and run complex math intensive MPC algorithms on those fairly easily.

Most modern DCSs use fieldbus for communication between sensors, actuators and controllers. As fieldbus is a shared digital media for many devices on the process, it may suffer from network bandwidth congestion if too many devices are used on a slow speed fieldbus network. To improve this situation, multi-rate control principle can also be applied in the fieldbus based control paradigm. In a fieldbus based multi-rate control scheme, the main problem is to find an index of system dynamics to initiate sampling-rate switching among different rates. MPC algorithm with its optimization cost function can be used for this purpose as an index, which is directly proportional to the system's dynamic states. A multi-rate MPC capable of switching sampling-rates depending on system's demand can contribute to the improvement of the data transfer rate through a slow speed fieldbus network in control scenarios. Using this scheme, energy conservation in field devices and reduction of mechanical wear of actuators can also be achieved.

## Chapter 3

# DUAL-RATE MODEL PREDICTIVE CONTROLLER DESIGN

### 3.1 Model Predictive Controller Design

#### 3.1.1 Augmented Model of the Plant

A mathematical model of the plant is needed to design a Model Predictive Control system from [26]. A state space model is used in the design. The state variables represent the states of the system as a function of time.

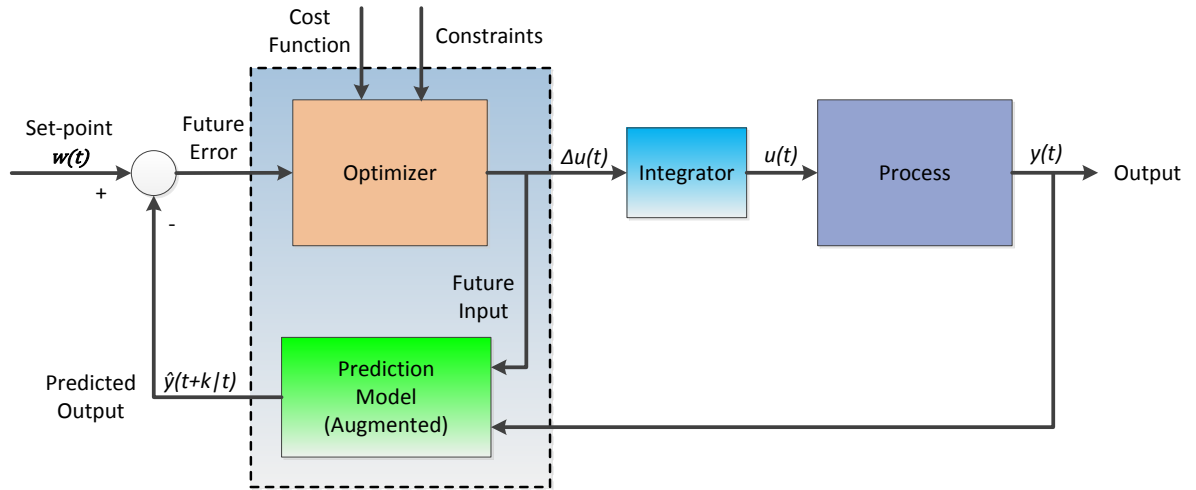


Fig 3.1: Model predictive controller block diagram with an integrator

An augmented model in terms of  $\Delta u(k)$ , instead of  $u(k)$ , of the plant state space model is required in the optimization cost function to calculate MPC gains and also for embedding an integrator in the design process. Figure 3.1 shows an MPC with an integrator. The integrator is used for eliminating the tracking error in the control loop [28], [29].

A single-input and single-output system is represented as:

$$x_m(k+1) = A_m x_m(k) + B_m u(k), \quad (3.1)$$

$$y(k) = C_m x_m(k) + D_m u(k) \quad (3.2)$$

where,

$u$  is the manipulated variable or input variable

$y$  is the process output

and,  $x_m$  is the state variable vector with assumed dimension  $n_1$ .

Because of the principle of receding horizon, it is assumed that  $y(k)$  will not be affected by the input  $u(k)$  at the same time.

Thus,  $D_m = 0$  in the plant model.

Now, by applying difference operation on (3.1)

$$x_m(k+1) - x_m(k) = A_m(x_m(k) - x_m(k-1)) + B_m(u(k) - u(k-1)).$$

And denoting difference of the state variable and control variable respectively by

$$\Delta x_m(k+1) = x_m(k+1) - x_m(k); \Delta x_m(k) = x_m(k) - x_m(k-1),$$

$$\Delta u(k) = u(k) - u(k-1).$$

$$\Delta x_m(k+1) = A_m \Delta x_m(k) + B_m \Delta u(k). \quad (3.3)$$

In the next step, by connecting  $\Delta x_m(k)$  to the output  $y(k)$  by a new state variable

$$x(k) = [\Delta x_m(k)^T y(k)]^T,$$

And the output difference equation becomes

$$\begin{aligned} y(k+1) - y(k) &= C_m(x_m(k+1) - x_m(k)) = C_m \Delta x_m(k+1) \\ &= C_m A_m \Delta x_m(k) + C_m B_m \Delta u(k). \end{aligned} \quad (3.4)$$

The augmented state-space plant model is found from (3.3) and (3.4) as

$$\begin{aligned} \overbrace{\begin{bmatrix} \Delta x_m(k+1) \\ y(k+1) \end{bmatrix}}^{x(k+1)} &= \overbrace{\begin{bmatrix} A_m & O_m^T \\ C_m A_m & I \end{bmatrix}}^A \overbrace{\begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix}}^{x(k)} + \overbrace{\begin{bmatrix} B_m \\ C_m B_m \end{bmatrix}}^B \Delta u(k) \\ y(k) &= \overbrace{\begin{bmatrix} O_m & I \end{bmatrix}}^C \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} \end{aligned} \quad (3.5)$$

where,  $O_m = \overbrace{[0 \ 0 \ 0 \ \dots \ 0]}^{n_1}$ ,  $I$  is the identity matrix with dimensions  $q \times q$ , which is the number of outputs; and  $O_m$  is a  $q \times n_1$  zero matrix. In (1.38),  $A_m$ ,  $B_m$  and  $C_m$  have dimension  $n_1 \times n_1$ ,  $n_1 \times m$  and  $q \times n_1$ , respectively.

Now, this A, B, C triplet based augmented model can be used in the design of predictive controller.

### 3.1.2 Predictive Control within One Optimization Window

After the formulation of the augmented mathematical model of the plant, in designing a predictive control system, the next step is to synthesize desired control input to achieve the desired plant output. Within an optimization window this prediction is described, where,  $k_i$  is the current time and  $N_p$  is number of samples.

## Prediction of State and Output Variables

Assuming that, the state variable vector  $x(k_i)$  is available through measurement where, at the sampling instant  $k_i$ ,  $k_i > 0$ , the state  $x(k_i)$  provides the current plant information. The future control trajectory is denoted as

$$\Delta u(k_i), \Delta u(k_i + 1), \dots, \Delta u(k_i + N_c - 1),$$

where,

$N_c$  = control horizon dictating the number of parameters

$N_p$  = number of samples, is also known as the prediction horizon or the length of the optimization window

Let the future state variables be denoted by,

$$x(k_i + 1|k_i), x(k_i + 2|k_i), \dots, x(k_i + m|k_i), \dots, x(k_i + N_p|k_i),$$

where, the predicted state variable is  $x(k_i + m|k_i)$  at  $(k_i + m)$  with given current plant information  $x(k_i)$ . The control horizon  $N_c$  is chosen to be less than (or equal to) the prediction horizon  $N_p$ .

Now, based on the augmented state-space model (A, B, C), using the set of future control variables, the future state variables are calculated sequentially:

$$\begin{aligned} x(k_i + 1|k_i) &= Ax(k_i) + B\Delta u(k_i) \\ x(k_i + 2|k_i) &= Ax(k_i + 1|k_i) + B\Delta u(k_i + 1) \\ &= A^2x(k_i) + AB\Delta u(k_i) + B\Delta u(k_i + 1) \\ &\vdots \\ x(k_i + N_p|k_i) &= A^{N_p}x(k_i) + A^{N_p-1}B\Delta u(k_i) + A^{N_p-2}B\Delta u(k_i + 1) \\ &\quad + \dots + A^{N_p-N_c}B\Delta u(k_i + N_c - 1). \end{aligned}$$

Predicted output variables are then calculated from the predicted state variables, by substitution

$$y(k_i + 1|k_i) = CAx(k_i) + CB\Delta u(k_i) \quad (3.6)$$

$$y(k_i + 2|k_i) = CA^2x(k_i) + CAB\Delta u(k_i) + CB\Delta u(k_i + 1)$$

$$y(k_i + 3|k_i) = CA^3x(k_i) + CA^2B\Delta u(k_i) + CAB\Delta u(k_i + 1) \\ + CB\Delta u(k_i + 2)$$

$\vdots$

$$y(k_i + N_p|k_i) = CA^{N_p}x(k_i) + CA^{N_p-1}B\Delta u(k_i) + CA^{N_p-2}B\Delta u(k_i + 1) \\ + \dots + CA^{N_p-N_c}B\Delta u(k_i + N_c - 1). \quad (3.7)$$

All predicted variables are formulated in terms of current state variable information  $x(k_i)$  and the future control input  $\Delta u(k_i + j)$ , where  $j = 0, 1, \dots, N_c - 1$ .

Defining vectors,

$$Y = [y(k_i + 1|k_i)y(k_i + 2|k_i)y(k_i + 3|k_i) \dots y(k_i + N_p|k_i)]^T \\ \Delta U = [\Delta u(k_i)\Delta u(k_i + 1)\Delta u(k_i + 2) \dots \Delta u(k_i + N_c - 1)]^T,$$

In SISO case, the  $N_p$  is the dimension of  $Y$  and  $N_c$  the dimension of  $\Delta U$ . Combining (3.6) and (3.7) together in a compact matrix form as

$$Y = Fx(k_i) + \Phi\Delta U, \quad (3.8)$$

where,

$$F = [CA \ CA^2 \ CA^3 \ \dots \ CA^{N_p}]^T ;$$

$$\Phi = \begin{bmatrix} CB & 0 & 0 & \dots & 0 \\ CAB & CB & 0 & \dots & 0 \\ CA^2B & CAB & CB & \dots & 0 \\ \vdots & & & & \\ CA^{N_p-1}B & CA^{N_p-2}B & CA^{N_p-3}B & \dots & CA^{N_p-N_c}B \end{bmatrix}$$



### 3.1.3 Optimization

Assuming that the set-point information is given in the form of the following data vector

$$R_S^T = \overbrace{[1 \ 1 \ \dots \ 1]}^{N_p} r(k_i)$$

Defining the cost function  $J$ , which reflects the control objective as

$$J = (R_S - Y)^T (R_S - Y) + \Delta U^T \bar{R} \Delta U, \quad (3.9)$$

where, the first term of the cost function is associated with the errors between the predicted error and set-point and the second term is the size of  $\Delta U$ . The optimization is to make the objective function  $J$ , as small as possible.  $\bar{R}$  is a diagonal matrix where  $\bar{R} = r_w I_{N_C \times N_C} (r_w \geq 0)$  and  $r_w$  is a tuning parameter to achieve the expected closed-loop performance.

The problem is how to find the optimal  $\Delta U$  that minimizes  $J$ . By using (3.8),  $J$  can be written as follows

$$J = (R_S - Fx(k_i))^T (R_S - Fx(k_i)) - 2\Delta U^T \Phi^T (R_S - Fx(k_i)) + \Delta U^T (\Phi^T \Phi + \bar{R}) \Delta U. \quad (3.10)$$

Now, by taking the first derivative of the cost function  $J$ ; it follows that

$$\frac{\partial J}{\partial \Delta U} = -2\Phi^T (R_S - Fx(k_i)) + 2(\Phi^T \Phi + \bar{R}) \Delta U, \quad (3.11)$$

The minimum  $J$  is obtained as the necessary condition for optimization

$$\frac{\partial J}{\partial \Delta U} = 0,$$

And the optimal solution for the control signal can be found as

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (R_S - Fx(k_i)), \quad (3.12)$$

with the assumption that  $(\Phi^T \Phi + \bar{R})^{-1}$  exists. In the optimization literature, the matrix  $(\Phi^T \Phi + \bar{R})^{-1}$  is called the *Hessian matrix*.  $R_s$  is a data vector that contains the set-point data as

$$R_s = \overbrace{[1 \ 1 \ 1 \ \dots \ 1]^T}^{N_p} r(k_i) = \bar{R}_s r(k_i),$$

where,

$$\bar{R}_s = \overbrace{[1 \ 1 \ 1 \ \dots \ 1]^T}^{N_p}$$

The optimal control signal links the set-point signal  $r(k_i)$  and the state variable  $x(k_i)$  through the following equation:

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (\bar{R}_s r(k_i) - Fx(k_i)). \quad (3.13)$$

### 3.1.4 Receding Horizon Control

With the receding horizon control principle, only the first sample of this sequence, *i.e.*  $\Delta u(k_i)$  needs to be implemented, although the optimal parameter vector  $\Delta U$  contains the controls  $\Delta u(k_i)$ ,  $\Delta u(k_i + 1)$ ,  $\Delta u(k_i + 2)$ , ...,  $\Delta u(k_i + N_c - 1)$  and the rest of the sequence can be ignored for the time being. When the next sample period arrives, more recent measurement from the state vector  $x(k_i + 1)$  is taken for calculating the new sequence of control signals. This procedure is repeated in real time to form a receding horizon control law.

The optimal solution for the control signal in a closed-loop control system is

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} (\Phi^T R_s - \Phi^T Fx(k_i))$$

where,  $(\Phi^T \Phi + \bar{R})^{-1} \Phi^T R_s$  corresponds to the set-point change and  $-(\Phi^T \Phi + \bar{R})^{-1} \Phi^T F$  corresponds to the state feedback control within the framework of predictive control,

According to the receding horizon control principle

$$\begin{aligned} \Delta u(k_i) &= [1 \ 0 \ \dots \ 1](\Phi^T \Phi + \bar{R})^{-1}(\Phi^T \bar{R}_s r(k_i) - \Phi^T F x(k_i)) \\ &= K_y r(k_i) - K_{mpc} x(k_i), \end{aligned} \quad (3.14)$$

where,  $K_y$  is the first element of the  $(\Phi^T \Phi + \bar{R})^{-1} \Phi^T \bar{R}_s$ ,  
 $K_{mpc}$  is the first row of  $(\Phi^T \Phi + \bar{R})^{-1} \Phi^T F = [K_x, K_y]$ ,  
 $K_x$  is the elements excluding  $K_y$  in the last of  $K_{mpc}$

Hence, the augmented state-space model (3.5) can be written as

$$x(k+1) = Ax(k) + B\Delta u(k) \quad (3.15)$$

By using (3.14), (3.15) becomes,

$$x(k+1) = Ax(k) - BK_{mpc}x(k) + BK_y r(k) \quad (3.16)$$

$$= (A - BK_{mpc})x(k) + BK_y r(k). \quad (3.17)$$

Thus, from the closed-loop characteristic equation, closed-loop eigenvalues can be evaluated:

$$\det [\lambda I - (A - BK_{mpc})] = 0.$$

### 3.1.5 Discrete-time Predictive Control System (without observer)

Applying the algorithm described in section 3.4, the following closed-loop control system is designed as shown in Fig 3.2. Here,  $K_y$  is the set-point error gain and  $K_x$  is the state feedback gain. An integrator is incorporated before the process for tackling the tracking

error, which in practical systems arises due to system modeling inaccuracies and disturbances. Model Predictive Controller in Fig 3.2 consists of two loops, where one is the state feedback loop, and the other is output feedback loop. As this is a theoretical MPC formulation, the states are known from the mathematical model of the process. But in practical systems, the states of the process are not measurable through sensor feedback. For addressing this issue, in practical systems, an observer is used to estimate the states of the process.

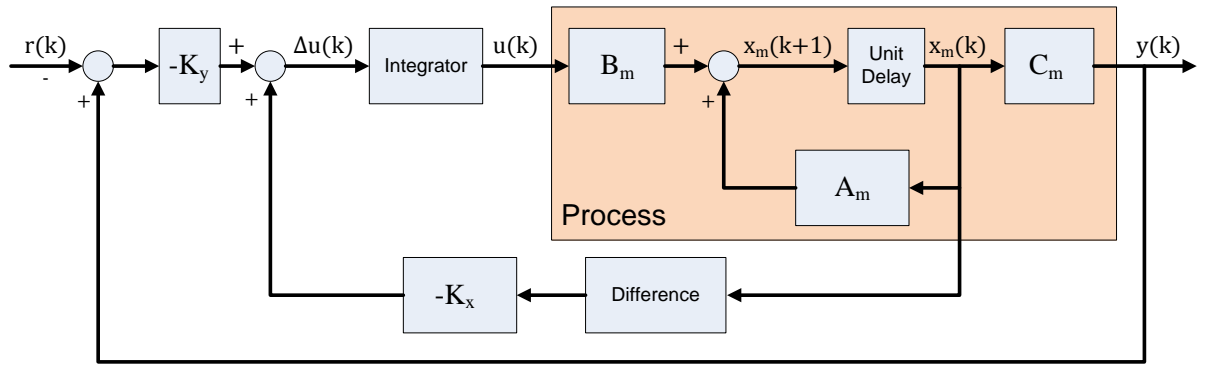


Fig 3.2: MPC without an observer

### 3.1.6 Observer based Predictive Controller

Now, the above design can be improved to be used in practical systems where all the states of the system are not measurable. In such a system, an observer is needed to estimate the states of the system through the observer, provided that the system is observable.

The observer can be constructed using the following equation

$$\hat{x}(k_i + 1) = A\hat{x}(k_i) + B\Delta u(k_i) + K_{ob}(y(k_i) - C\hat{x}(k_i)).$$

The observer gain  $K_{ob}$  is calculated by using a Kalman Filter. The system is shown in Fig 3.3 after the addition of an observer as follows:

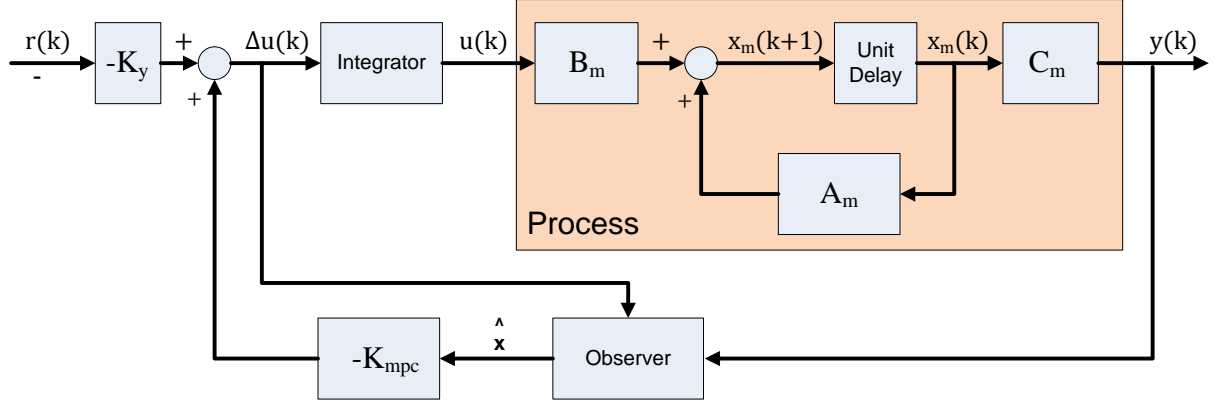


Fig 3.3: MPC with an observer

### 3.1.7 Kalman Filter

For a single-output case, if the pair  $(A_m, C_m)$  is observable then pole-placement technique can be used to determine  $K_{ob}$  and place eigenvalues of the observer (i.e. of the matrix  $A_m - K_{ob}C_m$ ) at the desired locations. A Kalman filter can be used in both SISO and MIMO systems. Assuming that,

$$\begin{aligned} x_m(k+1) &= A_m x_m(k) + B_m u(k) + d(k) \\ y(k) &= C_m x_m(k) + \xi(k), \end{aligned} \quad (3.18)$$

where, (3.18) is the model dynamic with Gaussian process noise  $d$  and Gaussian measurement noise  $\xi$  added with (3.1) and (3.2). Their covariance matrices are defined as follows:

$$E\{d(k)d(\tau)^T\} = \Theta \delta(k - \tau)$$

$$E\{\xi(k)\xi(\tau)^T\} = \Gamma \delta(k - \tau),$$

where,  $\delta(k - \tau) = 1$ , if  $k = \tau$  and  $\delta(k - \tau) = 0$  if  $k \neq \tau$ .

The optimum observer gain  $K_{ob}(i)$  is solved recursively for  $i = 0, 1, \dots$ , using

$$K_{ob}(i) = A_m P(i) C_m^T (\Gamma + C_m P(i) C_m^T)^{-1}, \quad (3.19)$$

And

$$P(i+1) = A_m \{P(i) - P(i) C_m^T (\Gamma + C_m P(i) C_m^T)^{-1} C_m P(i)\} A_m^T + \Theta. \quad (3.20)$$

More specifically,  $P(O)$  satisfies the following equation

$$E\{[x(O) - \hat{x}(O)][x(O) - \hat{x}(O)]^T\} = P(O).$$

Also, assuming that the system  $(C_m, A_m)$  is detectable from the output  $y(k)$ , then, as  $k \rightarrow \infty$ , the discrete-time algebraic Riccati equation is solved from (3.19) and (3.20) as

$$P(\infty) = A_m \{P(\infty) - P(\infty) C_m^T (\Gamma + C_m P(\infty) C_m^T)^{-1} C_m P(\infty)\} A_m^T + \Theta, \quad (3.21)$$

and

$$K_{ob}(\infty) = A_m P(\infty) C_m^T (\Gamma + C_m P(\infty) C_m^T)^{-1}. \quad (3.22)$$

### 3.1.8 Predictive Control with State Estimation

An observer is used in the implementation of the model predictive controller, where the state variable  $x(k_i)$  at time  $k_i$  is not measurable. Where, the state variable  $x(k_i)$  can be estimated via an observer of the form:

$$\hat{x}(k_i + 1) = A\hat{x}(k_i) + B\Delta u(k_i) + K_{ob}(y(k_i) - C\hat{x}(k_i)). \quad (3.23)$$

The cost function of the predictive controller becomes,

$$\begin{aligned} J = (R_s - F\hat{x}(k_i))^T (\bar{R}_s r(k_i) - F\hat{x}(k_i)) - 2\Delta U^T \Phi^T (R_s - F\hat{x}(k_i)) \\ + \Delta U^T (\Phi^T \Phi + \bar{R}) \Delta U, \end{aligned} \quad (3.24)$$

And, the optimal solution of the above is

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (R_s - F \hat{x}(k_i)). \quad (3.25)$$

Furthermore, application of the receding horizon control principle leads to the optimal solution of  $\Delta u(k_i)$  at time  $k_i$ :

$$\Delta u(k_i) = K_y r(k_i) - K_{mpc} \hat{x}(k_i), \quad (3.26)$$

Therefore, (3.15) becomes,

$$\begin{aligned} x(k+1) &= Ax(k) + B\Delta u(k) \\ &= Ax(k) + BK_y r(k) - BK_{mpc} \hat{x}(k), \end{aligned} \quad (3.27)$$

where  $\Delta u(k)$  is substituted with (3.26).

The closed-loop observer error equation becomes:

$$\tilde{x}(k+1) = (A - K_{ob}C)\tilde{x}(k), \quad (3.28)$$

where,  $\tilde{x}(k) = x(k) - \hat{x}(k)$ . Replacing  $\hat{x}(k)$  by  $x(k) - \tilde{x}(k)$ , (3.27) is rewritten as,

$$x(k+1) = (A - BK_{mpc})x(k) - BK_{mpc}\tilde{x}(k) + BK_y r(k). \quad (3.29)$$

By combining (3.28) with (3.29):

$$\begin{bmatrix} \tilde{x}(k+1) \\ x(k+1) \end{bmatrix} = \begin{bmatrix} A - K_{ob}C & O_{n \times n} \\ -BK_{mpc} & A - BK_{mpc} \end{bmatrix} \begin{bmatrix} \tilde{x}(k) \\ x(k) \end{bmatrix} + \begin{bmatrix} O_{n \times m} \\ BK_y \end{bmatrix} r(k), \quad (3.30)$$

Where,  $O_{n \times n}$  is a  $n \times n$  zero matrix and  $O_{n \times m}$  is a  $n \times m$  zero matrix.

The characteristic equation of the closed-loop state-space system is determined by

$$\det [\lambda I - \begin{bmatrix} A - K_{ob}C & O_{n \times n} \\ -BK_{mpc} & A - BK_{mpc} \end{bmatrix}] = 0,$$

Or, it can be written as:

$$\det (\lambda I - (A - K_{ob}C)) \det (\lambda I - (A - BK_{mpc})) = 0,$$

The above equation essentially means that the closed-loop model predictive control system with state estimate can be decomposed into two independent characteristic equations

$$\det (\lambda I - (A - K_{ob}C)) = 0 \tag{3.30}$$

$$\det (\lambda I - (A - BK_{mpc})) = 0. \tag{3.31}$$

The solutions to the above two equations give the closed-loop eigenvalues. They correspond to predictive control loop eigenvalue and observer-loop eigenvalues. This means that designing observer and predictive control law can be carried out separately and in that process the eigenvalues remain unchanged.

### 3.2 Dual-rate Model Predictive Controller Mechanism

In a dual-rate MPC, two MPCs are running in parallel as shown in Fig 3.4, where MPC2 has an n-times slower sampling-rate than MPC1. In this case, both the costs  $J_{mpc1}$  and  $J_{mpc2}$  of MPC1 and MPC2 respectively is calculated online and compared with a threshold value for switching between the two sampling-rates. Detailed switching strategy for sensor feedback and actuator sampling-rates through SW1 and SW2 are discussed in section 3.3. Set-points for the two MPCs are identical in the dual-rate MPC. The threshold value for switching sampling-rates is selected by trial and error for best performance and no optimization has been used. Fig 3.4 shows the block diagram of the dual-rate MPC scheme considered in this thesis.



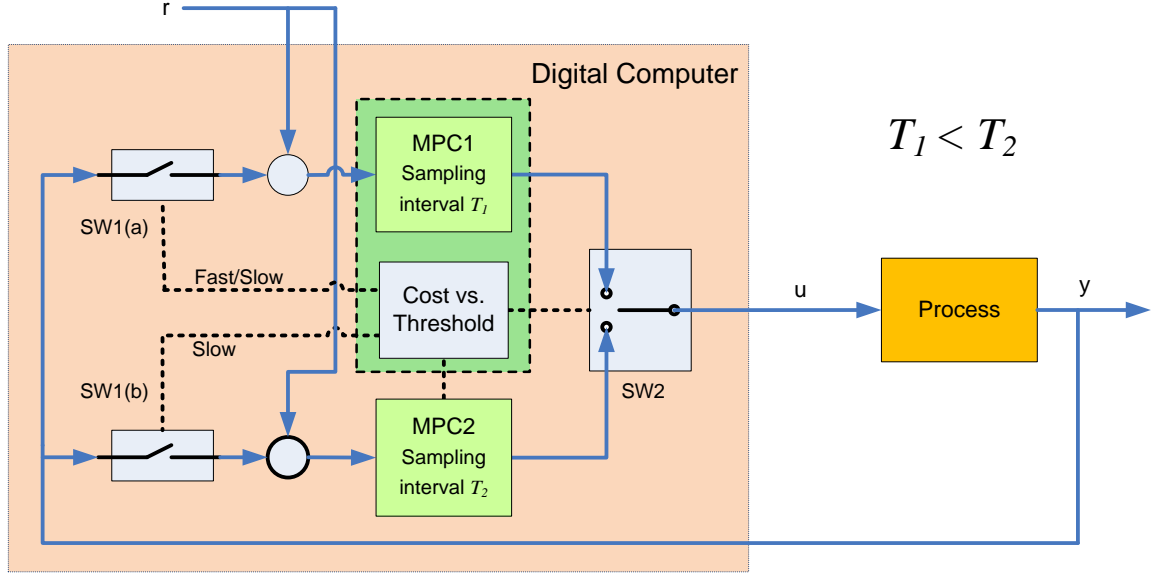


Fig 3.4: Dual-rate model predictive controller

It is important to note that MPC1 always runs at the fast sampling-rate or at sampling interval of  $T_1$  shown in Fig 3.4 and MPC2 always runs at the slow sampling-rate or at sampling interval of  $T_2$  shown in Fig 3.4. The sampler SW1(a) runs in two rates but the other sampler SW1(b) always runs at slow sampling-rate. Also, the two MPC cost functions  $J_{mpc1}$  for MPC1 and  $J_{mpc2}$  for MPC2 are calculated online at the fast sampling-rate of MPC1 (or at sampling interval of  $T_1$  shown in Fig 3.4). When SW1(a) switches at the slower sampling-rate, the input to the fast running MPC1 is hold constant between updates.

### 3.3 Automatic Sampling-rate Switching Strategy

In this thesis, a strategy based on MPC cost function is adopted for switching between the two controller outputs and sensor sampling-rates automatically. Model Predictive Control has an optimizing cost function which in matrix form is as (3.9) and can generally be expressed as follows:

$$J = \sum_{i=1}^N w_{y_i} (r_i - y_i)^2 + \sum_{i=1}^N w_{u_i} \Delta u_i^2$$

where,

$y_i = i^{th}$  controlled variable

$r_i = i^{th}$  reference variable

$u_i = i^{th}$  manipulated variable

$w_{y_i}$  = weighting coefficient reflecting the relative importance of  $y_i$

$w_{u_i}$  = weighting coefficient reflecting the relative importance of  $r_i$

The first term deals with the quality of control and the second term is the control effort. The value of the cost function changes with the dynamic states of the system. Cost is high when the system variable is far from the reference signal and also in transient mode during an external disturbance. This cost becomes low and close to zero when the system reaches steady-state. Therefore, the MPC cost can directly be used for switching between two sampling-rates.

### 3.3.1 Control Signal Switching Strategy

For switching control signal sampling-rate between the two rates, the cost for the two MPCs,  $J_{mpc1}$  for MPC1 and  $J_{mpc2}$  for MPC2 (in the dual-rate controller) are calculated at the fast sampling-rate of MPC1 and compared with the threshold value to determine which MPC output should be used to the process actuator for control using Switch 2 (SW2). Although we have calculated both the costs  $J_{mpc1}$  and  $J_{mpc2}$ , for switching sampling-rates, only MPC2 cost  $J_{mpc2}$  is considered for comparing with the threshold cost because from experiment it is found that MPC2 cost is more stable than MPC1 cost and has less rate of change due to slower sampling-rate. When  $J_{mpc2} > \text{Threshold}$ , the controller switches to the MPC1 control signal for fast-rate control in transient situations. Otherwise, the controller remains at MPC2 control signal for slow-rate control at steady-state conditions.

### 3.3.2 Sensor Sampling-rate Switching Strategy

Along with the actuator sampling-rate switching through SW2, the sensor feedback sampling-rate of MPC1 is switched (together) between slower and faster rates through SW1(a) depending on the comparison between the threshold and  $J_{mpc2}$ . MPC2 is always sampled at the slow sampling-rate through SW1(b). When  $J_{mpc2} > \text{Threshold}$ , the MPC1 is sampled with the sensor values at the fast sampling-rate to cope with the transient condition and the MPC2 is sampled with the sensor values at the slow sampling-rate. Otherwise, MPC1 and MPC2 both are sampled with the sensor feedback values at the slow sampling-rate in steady-state conditions while MPC1 keeps running at fast-rate.

In steady-state conditions, both the MPCs are sampled with the sensor feedback values at the slow sampling-rate (at sampling interval of  $T_2$  shown in Fig 3.4). The MPC1 runs at the fast sampling-rate (at sampling interval of  $T_1$  shown in Fig 3.4) and MPC2 runs at the slow sampling-rate (at sampling interval of  $T_2$  shown in Fig 3.4). The two cost functions  $J_{mpc1}$  and  $J_{mpc2}$  algorithms also keep running at the fast sampling-rate (at sampling interval of  $T_1$  shown in Fig 3.4).

Thus, sensor feedback and the actuator sampling-rate selection both switch together using SW1 and SW2 depending on the dynamics of the system with the intension to conserve network bandwidth.

## Chapter 4

# IMPLEMENTATION OF DUAL-RATE MPC ON SIEMENS PCS 7 DCS

This chapter demonstrates the implementation procedure of the proposed dual-rate MPC on Siemens PCS 7 Distributed Control System for automatic switching between two sampling-rates based on dynamic state of the system. By doing so, it can contribute to conserve costly fieldbus network bandwidth in slow speed fieldbus based distributed control systems. The general diagram of the physical setup is shown in Fig 4.1. The DCS is connected to the sensor and the actuator of the model plant through 4-20mA analog signal channels. The workstation is a PC which is connected to the DCS through Ethernet. The dual-rate MPC is designed in the workstation and downloaded to the DCS and then run to achieve the following:

- Demonstrate the simplicity of implementation of the proposed scheme on a readily available process controller, such as Siemens PCS 7 DCS
- Demonstrate the effectiveness of the proposed dual-rate MPC on an industrial grade control system

It is to be noted here that the physical Siemens PCS 7 DCS and the physical level process used in this experiment are already in place and are a part of the Control Instrumentation and Electrical Systems (CIES) Lab. Implementation of the dual-rate MPC implies only on developing a software algorithm for the dual-rate controller. Structured Control Language (SCL) has been used to develop the software controller and it is then converted to a custom function block. It is then downloaded from the workstation to the PCS 7 system to run and control the physical process. At hardware level, only minor wiring

from plant sensor to PCS 7 AI module and PCS 7 AO module to the plant actuator has been carried out to run experiments.

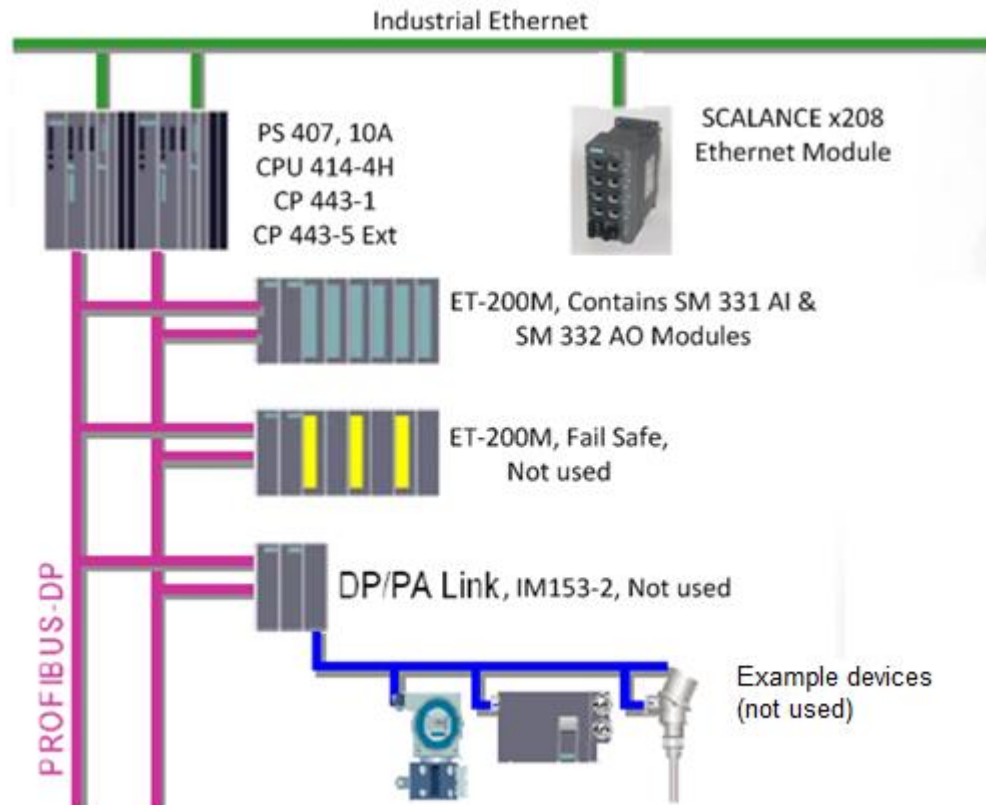


Fig 4.1: Siemens PCS 7 DCS block diagram

## 4.1 Why Siemens PCS 7

Siemens PCS 7 has some powerful features that make it suitable for this particular application. Model Predictive Control is a resource intensive algorithm which needs powerful matrix manipulations. The following features of PCS 7 are important for this thesis:

- i) Structured Control Language (SCL) for programming PCS 7 which uses the high level language syntax of PASCAL

- ii) SCL supports Function, Function Block, Organization Block, Data Block etc. for structured and modular programming. Some details on SCL can be found in Appendix D
- iii) Multidimensional array for easily implementing complex matrix operations
- iv) Fast processing capability of executing control loops as low as 10ms macrocycle

## 4.2 Experimental Setup and Modes of Experiment

In this experiment, Siemens PCS 7 DCS system is chosen as the controller. A physical plant is used as the process. It consists of a cylinder and an aluminum plate that can move vertically allowing air to pass through the gap between the plate and the cylinder wall. Air nozzles are distributed evenly under the plate and compressed air can pass through the nozzles to push the plate upwards. There is a proportional flow control valve to regulate the air supply through the nozzles (The details of the proportional valve and its flow curves can be found in Appendix A). The plate level is measured by a LASER based level sensor located on top of the cylinder. There is a hole on top of the cylinder for exhausting the air to maintain equilibrium.

The level sensor and the proportional air flow control valve are connected to the PCS 7 controller with 4-20mA signal connections.

Two modes of experiment have been done and presented for comparison as follows:

Table 4.1: Modes of experimental setup

Modes	Explanation
MODE- 1 (n=1/5, five times slower than standard)	Dual-rate MPC where faster rate MPC1 is at 5 Hz sampling-rate i.e. 200ms sampling time and slower rate MPC2 is at 1 Hz sampling-rate i.e. 1000ms sampling time (5 times slower)

MODE-2 ( $n=1/15$ , fifteen times slower than standard)	Dual-rate MPC where faster rate MPC1 is at 5 Hz sampling-rate i.e. 200ms sampling time and slower rate MPC2 is at 0.333 Hz sampling-rate i.e. 3000ms sampling time (15 times slower)
--	--

### 4.3 Block Diagram of the Dual-rate MPC Control Loop

Figure 4.2 shows the block diagram of the dual-rate MPC control loop. The digital software controller is implemented on the Siemens PCS 7 and it controls an electronic valve driver through 4-20mA signal to drive the mechanical Proportional Flow Valve to control the air flow. The proportional flow control valve used has a linear relationship between the solenoid armature position, which controls air flow rate to the input current (Appendix A). The LASER based level sensor sends the plate level to the Siemens PCS 7 through 4-20mA analog signal.

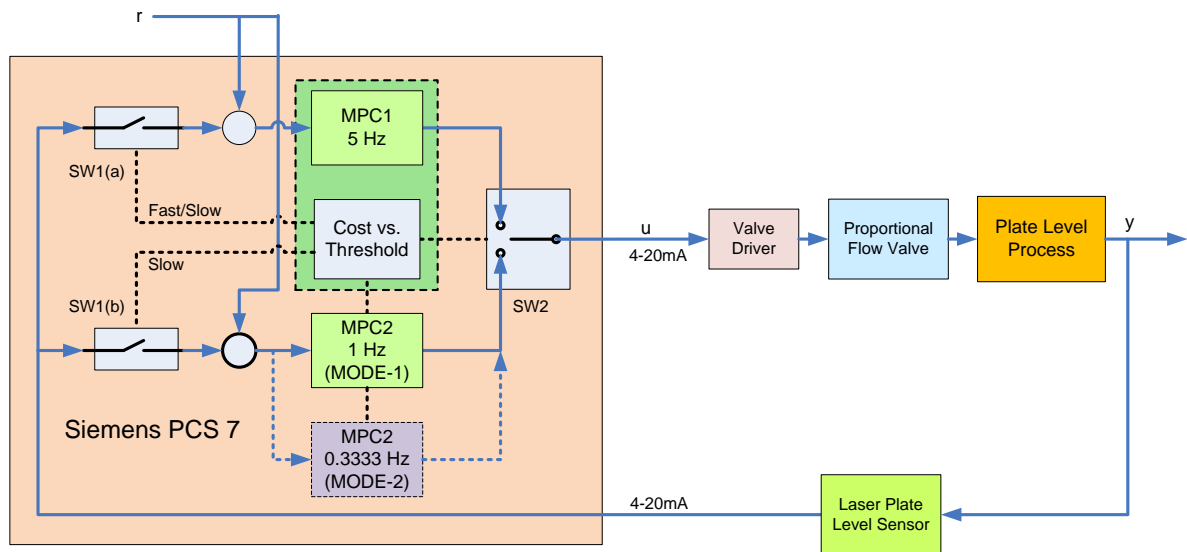


Fig 4.2: Block diagram of the dual-rate MPC control loop

## 4.4 Physical Setup

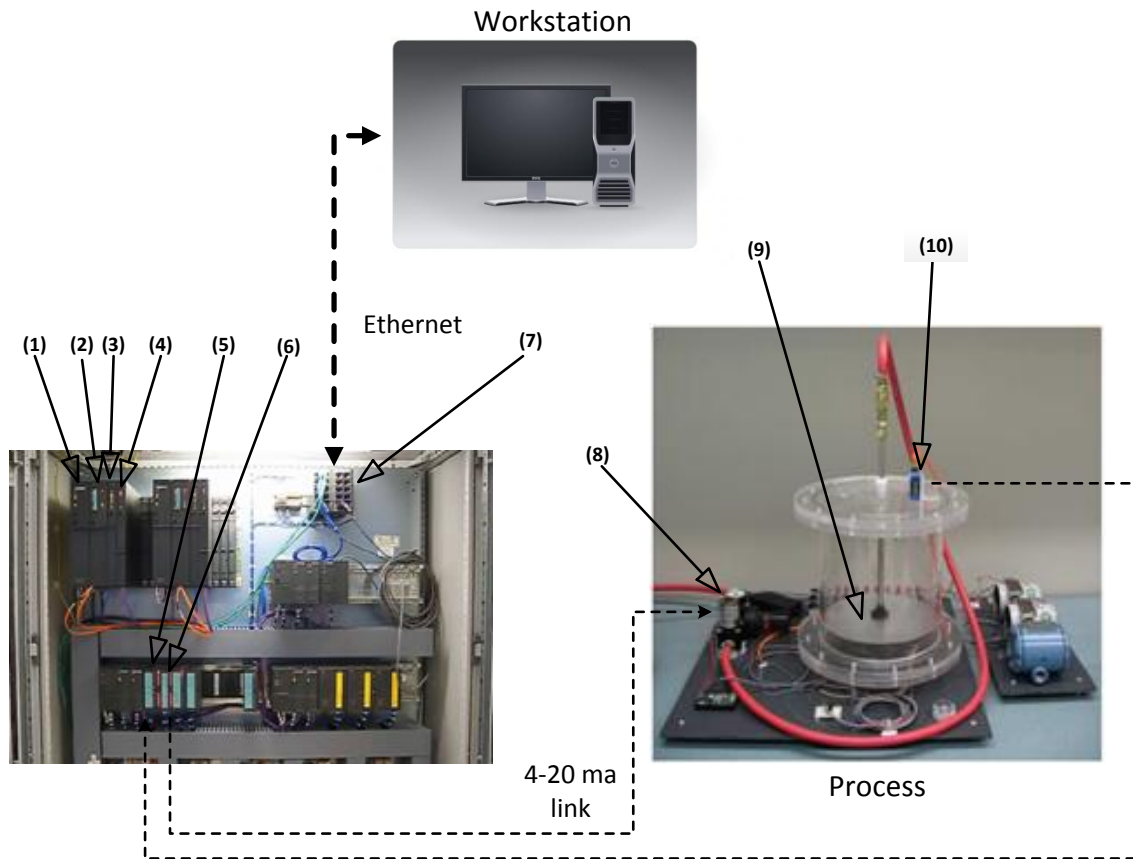


Fig 4.3: Physical setup containing Siemens PCS 7 DCS, process and workstation

Table 4.2: Components of the experimental setup

1.	Power supply (Redundant)
2.	CPU (Redundant)
3.	PROFINET communication processor card
4.	PROFIBUS communication processor card
5.	AI card with input range 4-20mA
6.	AO card with output range 4-20mA
7.	Ethernet switch
8.	Proportional valve (Actuator)
9.	Level process (range of movement from 0 to 75 cm)
10.	Laser level gauge (Sensor)



The experimental setup is a hardware version of the closed-loop system as depicted in Fig 4.3. Table 4.2 shows the hardware component list used in the experiment. The power supply (1) provides DC power to all the components from (2) to (7) listed on Table 4.2. The controller (2) is connected with Profinet communication processor card (3) for Ethernet communication through Ethernet switch (7) with the workstation. The controller (2) is also connected with the Profibus DP communication processor card (4) for communicating with the distributed peripherals (DP), such as AI (5) and AO (6) cards. The power supply, controller, communication processor and AI and AO cards are DIN rail mounted on PCS 7 control system rack. The plate level process components are Proportional valve (8), Level process (9) and Laser level gauge (10) respectively.

The Siemens PCS 7 DCS is connected to a workstation containing SIMATIC Manager software suit for programming and configuring the system. Figure 4.3 shows the physical setup used to demonstrate the dual-rate MPC scheme. One analog input of the PCS 7 is connected to the plant's plate level sensor through 4-20mA and one analog output is connected to the proportional flow control valve drive through 4-20mA link. Figure 4.4 shows the block diagram of the dual-rate Model Predictive Controller where MPC1 and MPC2 forms MPC\_DUAL and resides on the PCS 7 and are software components only. On the other hand, proportional air flow control valve and driver, plant and plate level sensor are physical hardware devices.

## 4.5 Dual-rate MPC Design Formulation and Parameters

### 4.5.1 Dual-rate MPC in MODE-1

#### MPC1 process model

$$G(z) = \frac{Y(z)}{U(z)} = \frac{0.02379z + 0.01861}{z^2 - 1.477z + 0.4771} \text{ at } T = 0.2 \text{ sec (5 Hz)}$$

#### MPC1 augmented state-space process model triplet from (3.5)

$$A = \begin{bmatrix} 0.4771 & 0 & 0 \\ 0.1413 & 1.000 & 0 \\ 0.2119 & 1.500 & 1.000 \end{bmatrix}; \quad B = \begin{bmatrix} 0.1413 \\ 0.0158 \\ 0.0237 \end{bmatrix}; \quad C = [0 \quad 0 \quad 1.000]$$

#### MPC1 gains

$$K_{mpc} = [1.8209 \quad 7.8106 \quad 0.8183]; \quad K_x = [1.8209 \quad 7.8106]; \quad K_y = 0.8183$$

#### MPC1 parameters

$$N_p = 10; \quad N_c = 4; \quad R_w = 1$$

---

#### MPC2 process model

$$G(z) = \frac{Y(z)}{U(z)} = \frac{0.2985z + 0.09684}{z^2 - 1.025z + 0.02472} \text{ at } T = 1.0 \text{ sec (1 Hz)}$$

#### MPC2 augmented state-space process model triplet from (3.5)

$$A = \begin{bmatrix} 0.0247 & 0 & 0 \\ 0.2635 & 1.000 & 0 \\ 0.3953 & 1.500 & 1.000 \end{bmatrix}; \quad B = \begin{bmatrix} 0.2635 \\ 0.1990 \\ 0.2985 \end{bmatrix}; \quad C = [0 \quad 0 \quad 1.000]$$

#### MPC2 gains

$$K_{mpc} = [0.6719 \quad 2.5118 \quad 0.6790]; \quad K_x = [0.6719 \quad 2.5118]; \quad K_y = 0.6790$$

#### MPC2 parameters

$$N_p = 10; \quad N_c = 4; \quad R_w = 1;$$

### 4.5.2 Dual-rate MPC in MODE-2

#### MPC1 process model

$$G(z) = \frac{Y(z)}{U(z)} = \frac{0.02379z + 0.01861}{z^2 - 1.477z + 0.4771} \text{ at } T = 0.2 \text{ sec (5 Hz)}$$

#### MPC1 augmented state-space process model triplet from (3.5)

$$A = \begin{bmatrix} 0.4771 & 0 & 0 \\ 0.1413 & 1.000 & 0 \\ 0.2119 & 1.500 & 1.000 \end{bmatrix}; \quad B = \begin{bmatrix} 0.1413 \\ 0.0158 \\ 0.0237 \end{bmatrix}; \quad C = [0 \quad 0 \quad 1.000]$$

#### MPC1 gains

$$K_{mpc} = [1.8209 \quad 7.8106 \quad 0.8183]; \quad K_x = [1.8209 \quad 7.8106]; \quad K_y = 0.8183$$

#### MPC1 parameters

$$N_p = 10; \quad N_c = 4; \quad R_w = 1;$$

---

#### MPC2 process model

$$G(z) = \frac{Y(z)}{U(z)} = \frac{1.107z + 0.1095}{z^2 - z + (1.511e-005)} \text{ at } T = 3.0 \text{ sec (0.333 Hz)}$$

#### MPC2 augmented state-space process model triplet from (3.5)

$$A = \begin{bmatrix} 0.0000151 & 0 & 0 \\ 0.2702661 & 1.000 & 0 \\ 0.4053992 & 1.500 & 1.000 \end{bmatrix}; \quad B = \begin{bmatrix} 0.2702661 \\ 0.7377658 \\ 1.1066488 \end{bmatrix}; \quad C = [0 \quad 0 \quad 1.000]$$

#### MPC2 gains

$$K_{mpc} = [0.2839 \quad 1.05076 \quad 0.4517]; \quad K_x = [0.2839 \quad 1.05076]; \quad K_y = 0.4517$$

#### MPC2 parameters

$$N_p = 10; \quad N_c = 4; \quad R_w = 1;$$

## 4.6 Implementing Dual-rate MPC Algorithm in SIMATIC Manager

SIMATIC manager is the development environment or IDE used for programming and configuring the Siemens PCS 7 DCS. First, a dual-rate MPC algorithm has been developed in the SCL Editor named as MPC\_DUAL function block. Then, the developed MPC\_DUAL function block is imported into CFC and connected with AI and AO library modules. The steps of designing MPC\_DUAL function block and creating dual-rate MPC project through SIMATIC Manager is described in some details in APPENDIX A. The AI function block is used for reading values from the sensor and an AO module (function block) for controlling the actuator. The CFC looks as follows in Fig 4.4 after connecting with other modules:

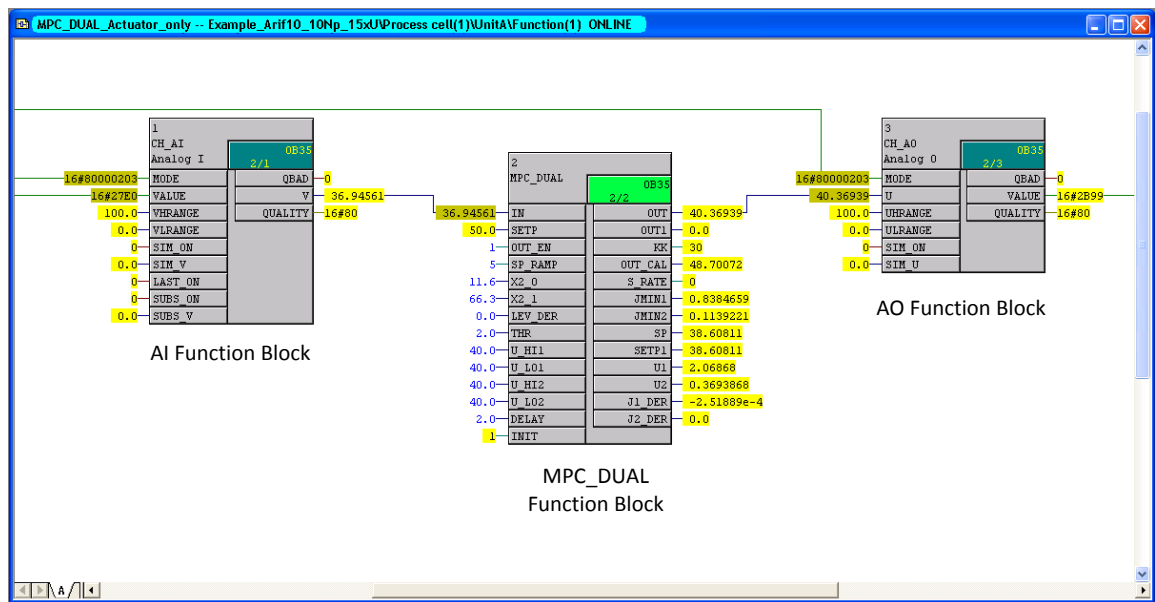


Fig 4.4: Dual-rate MPC on CFC editor

After connecting the AI, AO and MPC\_DUAL modules appropriately, the CFC file is compiled and then downloaded to the PCS 7 DCS CPU. One can then execute the program by choosing RUN mode.

A functional block diagram of overall system containing software components on the DCS and the physical plant is shown in Fig 4.5.

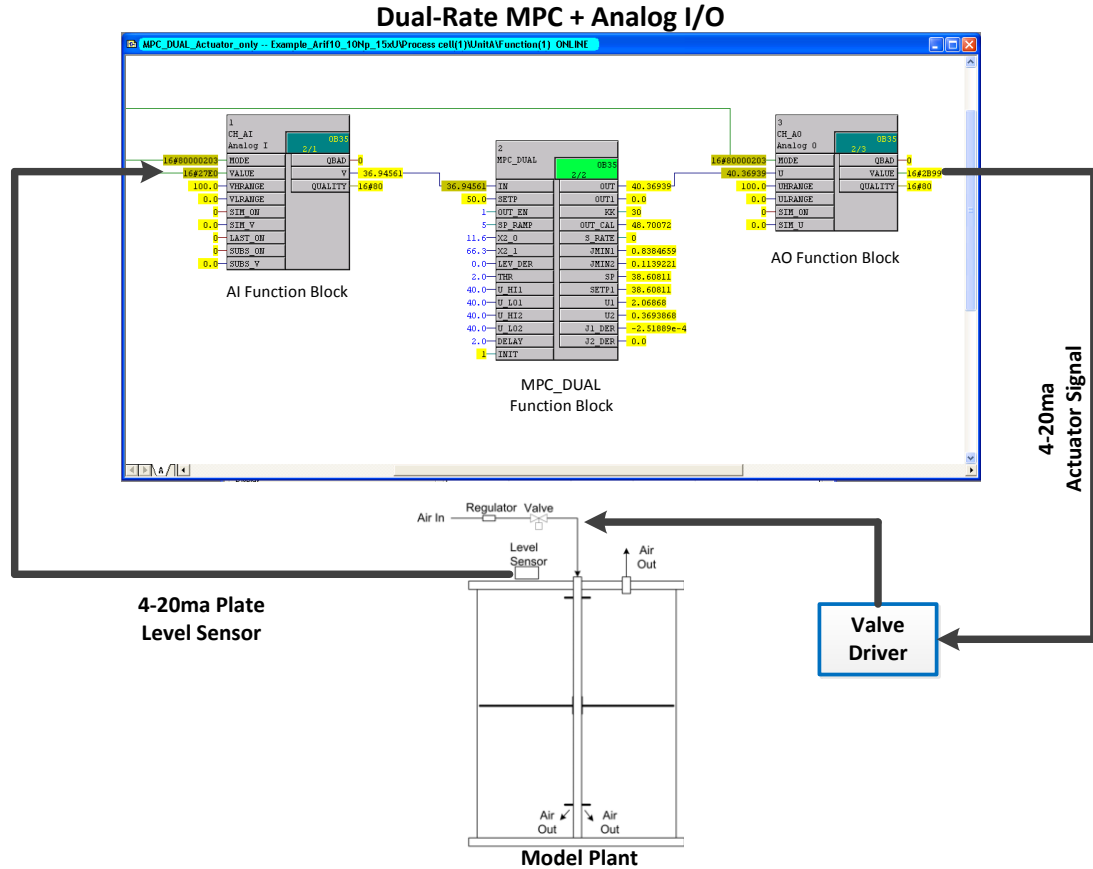


Fig 4.5: Dual-rate MPC Function Block and model plant block diagram

When the system is running, different variables e.g. plate level, MPC2 cost, set-point, actuator signal etc. can be viewed online in a software component of the SIMATIC CFC, known as Trend Display. The Trend Display in PCS 7's CFC has a minimum resolution of 1 sec. Figure 4.6 shows an example screen shot of a Trend Display, while the system is running.

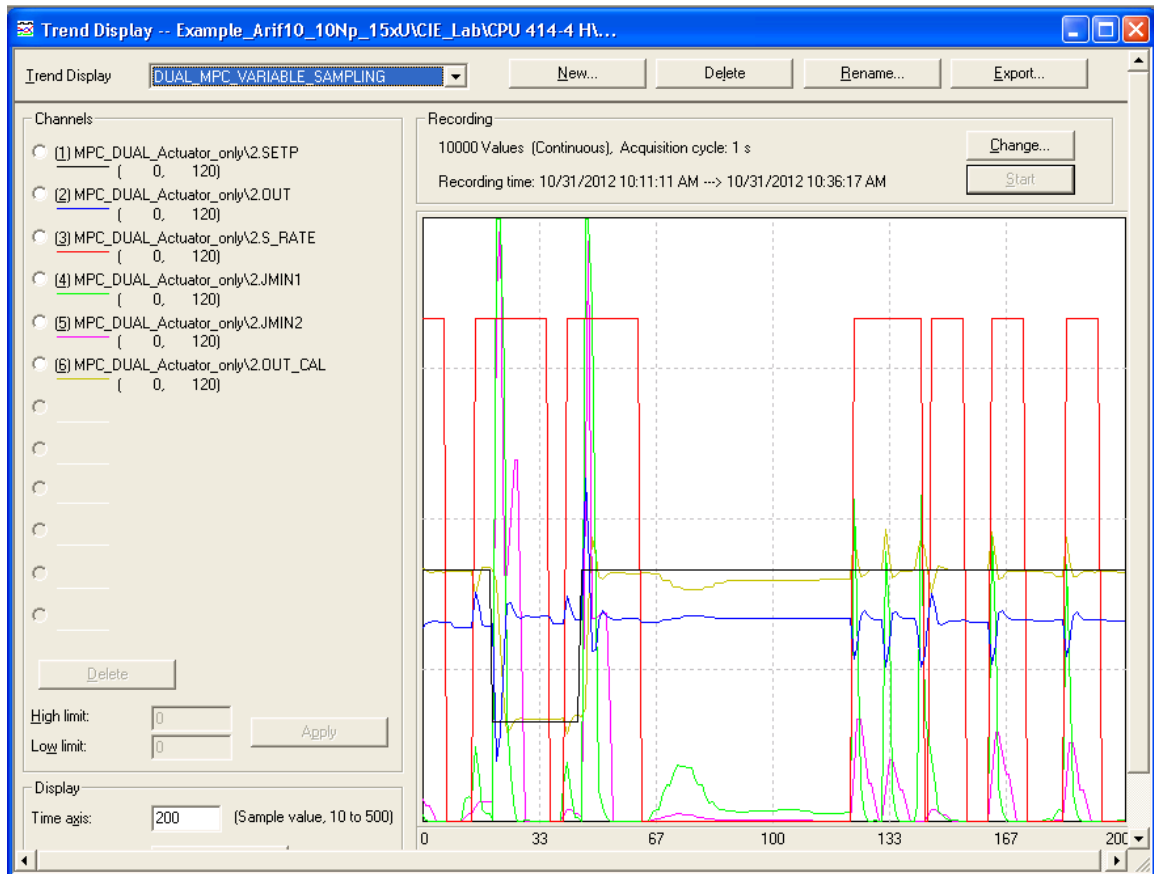


Fig 4.6: Trend Display showing various signals, resolution 1 sec (minimum)

## Chapter 5

# EXPERIMENTAL RESULTS AND DISCUSSION

In the proposed scheme, for achieving increased bandwidth through fieldbus, a dual-rate MPC has been designed and implemented on Siemens PCS 7 DCS. The lack of current fieldbus capabilities for supporting multiple sampling-rates at runtime forced us to implement the 4-20mA analog communication technology to demonstrate the concept of dual-rate sampling technique in a process control application. In a real digital fieldbus based control application, one can modify the fieldbus scheduling mechanism in conjunction with the proposed dual-rate MPC.

In this chapter, various experimental results are discussed. Most of the results are recorded online through a data capture software in the CFC Editor, known as Trend Display, which resides on the workstation connected to the Siemens PCS 7 DCS. Some data have also been collected through direct measurement of signal voltages through an oscilloscope. Various experiments have been performed. Results are simplified for easier demonstration of the concept. To achieve that and to study the effects of different transient conditions, such as step-change, external disturbance injection etc. different experiments have been performed. Step-change experiments have been performed by changing the set-point variable on the dual-rate MPC function block MPC\_DUAL manually. The disturbance injection experiments have been performed in two ways (1) by obstructing the air exit passage of the level process manually, (2) by directly imposing some voltage pulses on the actuator signal.

It is to be noted here that although in Fig 5.1, Fig 5.3, Fig 5.4 and Fig 5.7 it is seen to be a step-change with sharp rise and fall of the set-point curve, in practical system, sharp step-change is not used and in this experiment a slight ramp has been included to the set-point

signal by using a ramp algorithm within the controller. That is why the plate level is showing slight slope in Fig 5.1, Fig 5.3, Fig5.4 and Fig 5.7.

## 5.1 Test Scenarios

Seven test scenarios have been carried out. The details are listed in Table 5.1.

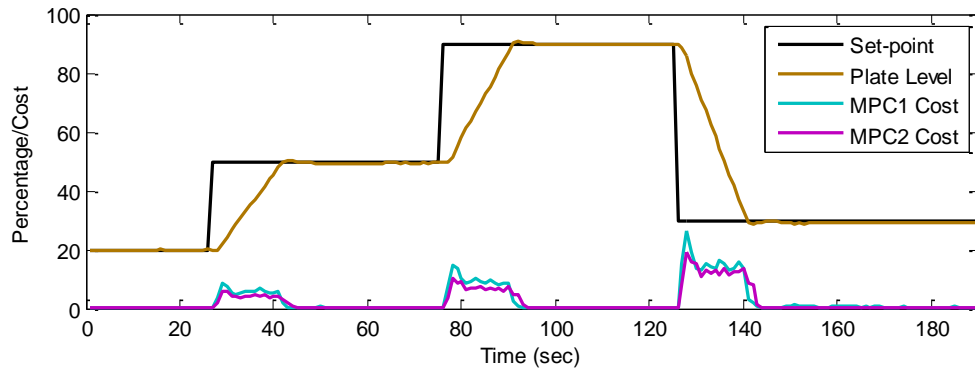
Table 5.1: Test scenarios

No.	Criteria	Explanation
1.	Cost-change due to set-point change	Study the effect of transient step-change on the cost of each MPC of the dual-rate MPC both in MODE-1 and MODE-2
2.	Sampling-rate switching on cost change	Study the effect of cost change on switching sampling-rate. Cost is compared with a threshold cost and sampling-rate is switched when the cost becomes greater than the threshold in both MODE-1 and MODE-2
3.	Sampling-rate switching on step-change	Study the effect of transient step-change on switching sampling-rate which happens due to the cost change on transients both in MODE-1 and MODE-2.
4.	Performance comparison on single-rate and dual-rate settings	Study the performance improvement of a dual-rate controller to compare it with a single slow-rate controller and a single fast-rate controller. Study the performance in both MODE-1 and MODE-2
5.	Effect of external disturbance on sampling-rate switching	Study the effect of external disturbance and observe the switching of the sampling-rate to cope with the transient situation in both MODE-1 and MODE-2. Disturbance is introduced by the obstruction of the air flow exit
6.	Effects of transients on dual-rated actuator signal	Study the actuator signal for confirming dual-rated control taking place in a real system where Siemens PCS 7 is controlling the physical plant with the dual-rate MPC

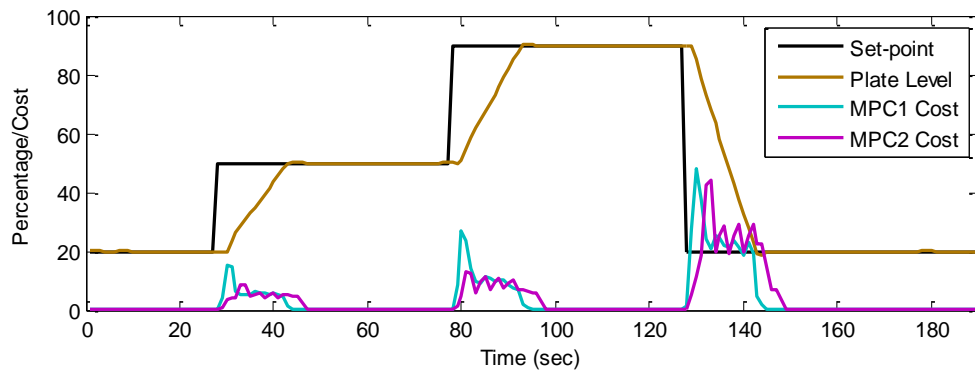


7.	Effect of disturbance in between two consecutive samples	Study the effect when disturbance arises in between two consecutive samples in MODE-2. Disturbance within 3000ms is injected through a signal generator
----	--	---

## 5.2 Cost-change due to Set-point Change



(a)

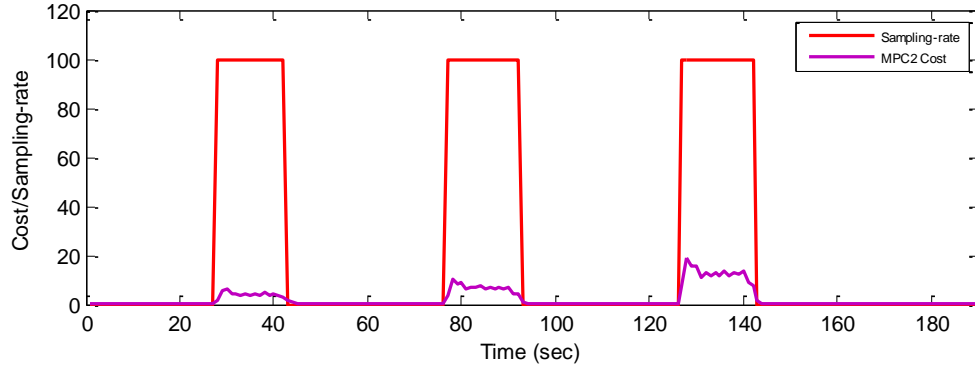


(b)

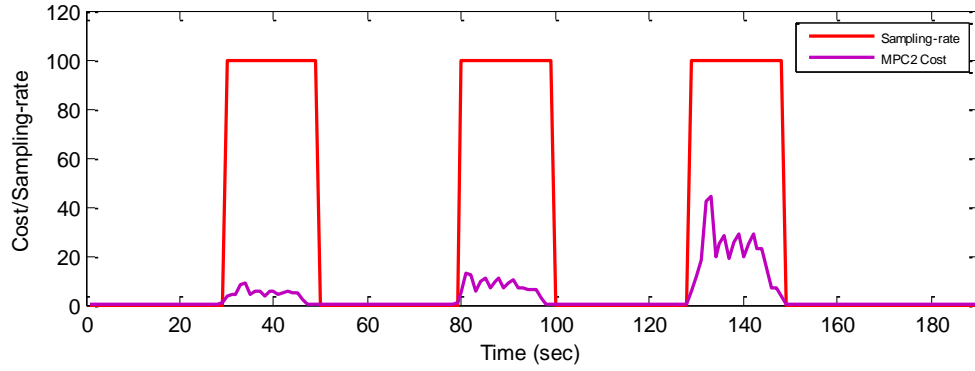
Fig 5.1: Step-change transient causing MPC1 and MPC2 cost change on (a) MODE-1  
(b) MODE-2

Figure 5.1 shows that a step-change can cause changes in costs associated with optimization cost of the MPC1 and MPC2. Figure 5.1(a) shows the case of MODE-1 and Fig 5.1(b) shows the case of MODE-2. In both cases, the step-change indicates cost change. This rise of cost over the threshold is utilized in switching the sampling-rate of the controller, which has been discussed in the next section.

### 5.3 Sampling-rate Switching on Cost Change



(a)

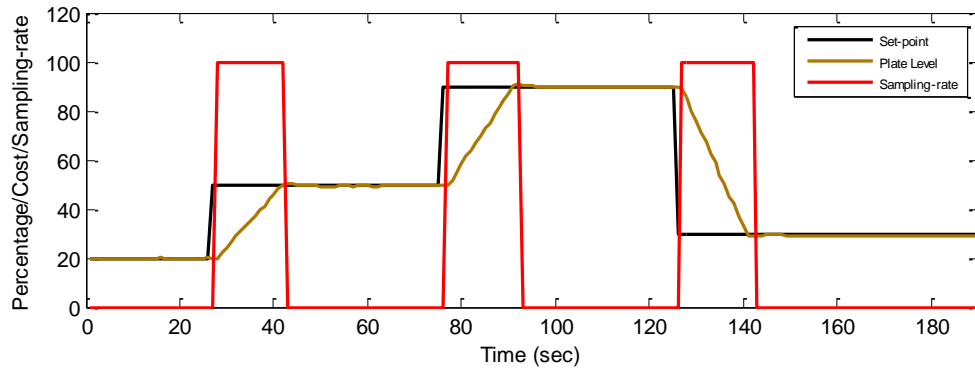


(b)

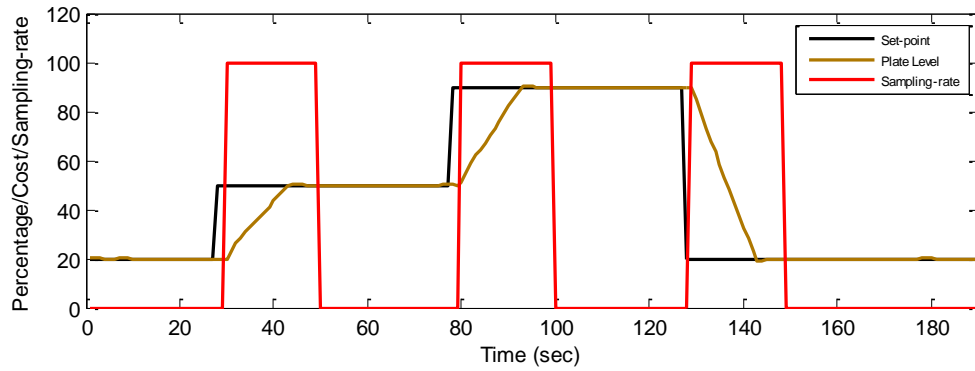
Fig 5.2: Change of MPC cost initiating sampling-rate switching (a) MODE-1 (b) MODE-2, (threshold cost 2.0 in both cases)

Figure 5.2 shows that the cost change that occurred due to the earlier step-changes causing initiation of switching sampling-rates. This initiation is done by comparing the MPC2 cost with a threshold when ( $J_{mpc2} > \text{Threshold}$ ). The threshold is decided by trial and error and not optimized in the current investigation. A threshold cost of 2.0 has been used in both cases for MODE-1 and MODE-2.

## 5.4 Sampling-rate Switching on Step-change



(a)



(b)

Fig 5.3: Step-change transient causing sampling-rate change (a) MODE-1, threshold cost 2.0  
(b) MODE-2, threshold cost 2.0

Figure 5.3(a) shows the effect of step-change causing sampling-rate to switch from slower (1 Hz) to faster (5 Hz) in MODE-1. The system is running at 20% set-point until at 27<sup>th</sup> sec. The set-point has been changed manually at 27<sup>th</sup> sec from 20% to 50%, which demanded higher sampling-rate to be initiated. From 27<sup>th</sup> sec to 45<sup>th</sup> sec the plant (plate level in this case) slowly followed a ramp to track the set-point. After the output reaches the set-point, the controller is switched back to slower sampling-rate again without losing control performance as it closely follows the set-point. Similar types of step-change have been tested on 74<sup>th</sup> sec and 126<sup>th</sup> sec and in both cases, where the sampling-rate has changed automatically.

Figure 5.3(b) shows similar behavior for MODE-2. In this case, the slower sampling-rate is 0.333 Hz. At this 15 times slower rate than optimum 200ms rate, system acts similarly, but the response time of the step-change is slower as seen in Fig 5.3(a) due to the slower sampling-rate.

## 5.5 Performance Comparison between Single-rate and Dual-rate Settings

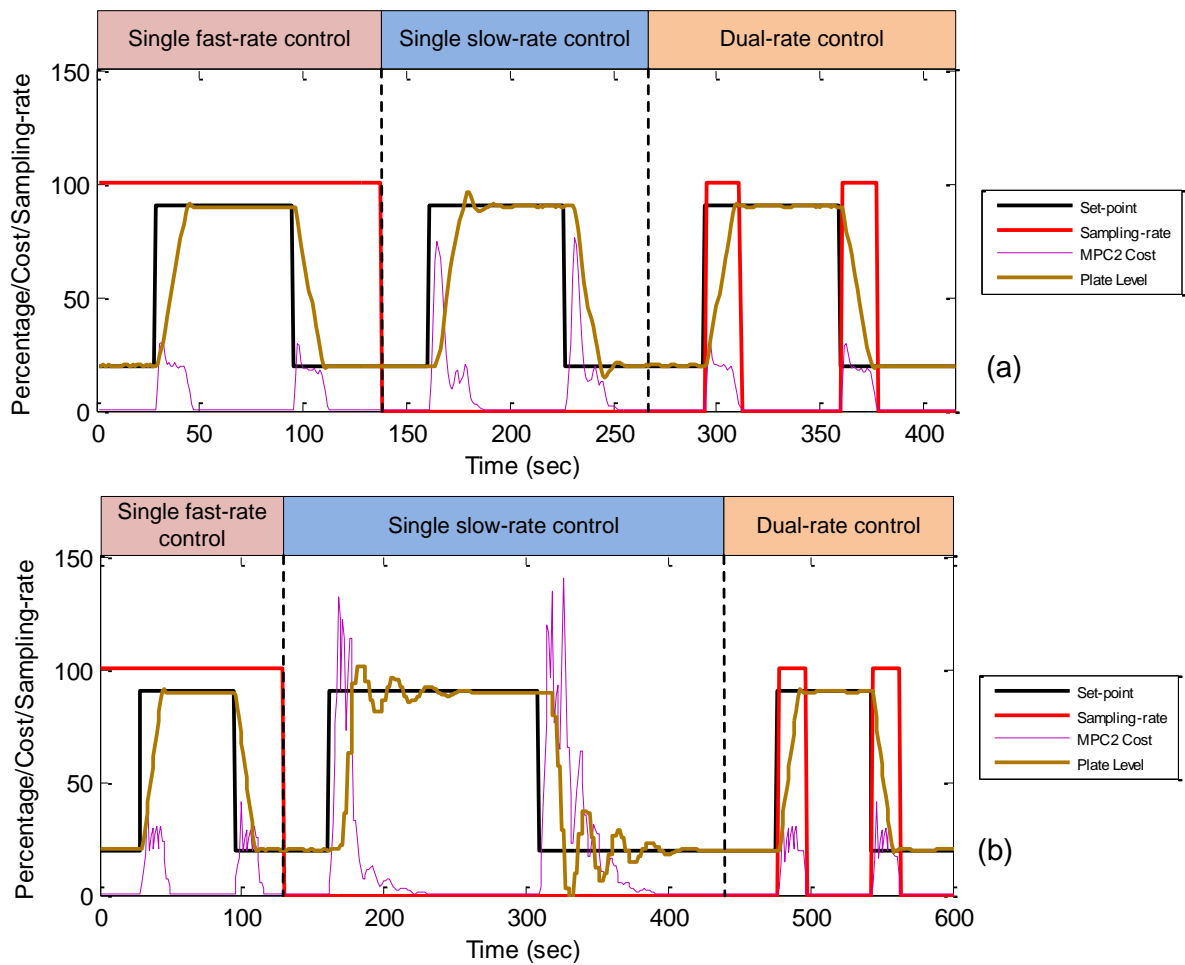


Fig 5.4: Performance comparison between single-rate controllers vs. a dual-rate controller  
(a) MODE-1, (b) MODE-2

The performance comparison between single a fast-rate controller vs. a single slow-rate controller vs. a dual-rate controller has been examined in this experiment.

Figure 5.4(a) shows the results for MODE-1. At the start of the experiment, the dual-rate feature of the controller is turned off and the system is running on 5 Hz as a single fast-rate controller up to 140<sup>th</sup> sec. Within this period, two step-change tests have been performed from 20% to 90% and 90% to 20%. At 140<sup>th</sup> sec, the system switches to slower sampling-rate of 1 Hz and is running as a single slow-rate controller and similar step-change tests have been done from 20% to 90% and 90% to 20%. At 270<sup>th</sup> sec, the system is switched to dual-rate mode and is running as a dual-rate controller. Identical step-change tests have been performed from 20% to 90% and 90% to 20% in dual-rate mode as well. The performances of these three modes have been compared in Table 5.2.

Figure 5.4(b) shows similar experiments in MODE-2. In MODE-2, due to very slow sampling-rate, the single-slow rate control mode takes long time to settle down.

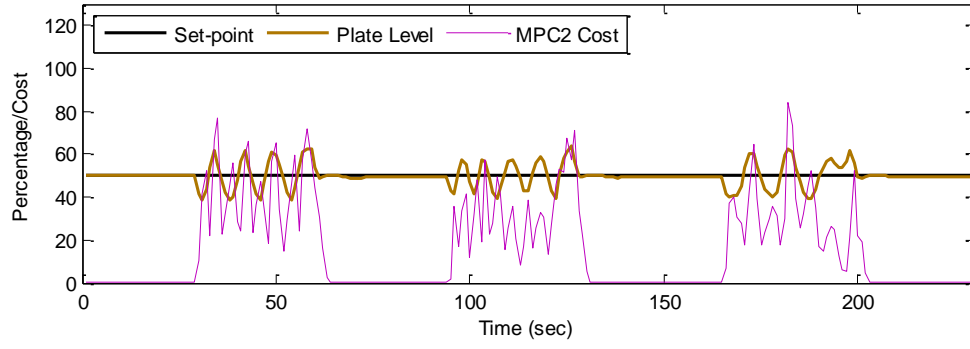
For quantitative comparison among the three scenarios in both MODE-1 and MODE-2, the dynamic properties (overshoot and settling time) and a defined index  $e_{perf} = (\frac{1}{N})\sqrt{\sum_{k=1}^N (|r_k - y_k| + \Delta u_k^2)}$  are used.

Table 5.2: Quantitative comparison of single-rate and dual-rate controllers

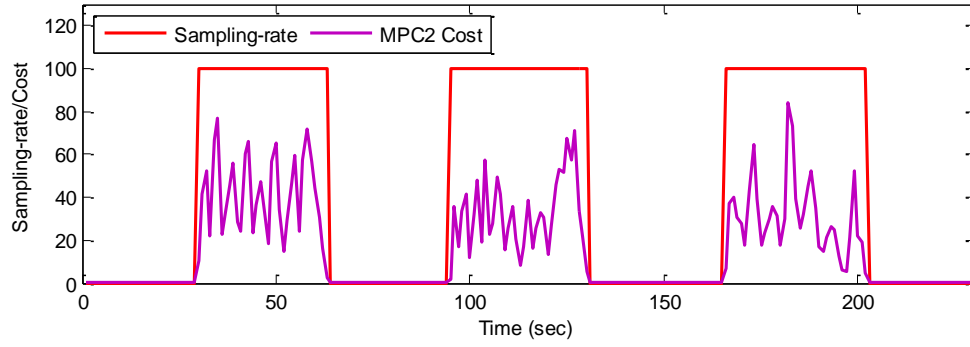
		<b>Single fast-rate controller</b>	<b>Single slow- rate controller</b>	<b>Dual-rate controller</b>
MODE-1	Overshoot (%)	1.02	6.78	1.02
	Settling time (sec)	4	26	4
	$e_{perf}$	1.34	2.34	1.34
MODE-2	Overshoot (%)	1.02	12.42	1.06
	Settling time (sec)	4	111	45
	$e_{perf}$	1.34	4.27	1.84

## 5.6 Effect of External Disturbance on Sampling-rate Switching

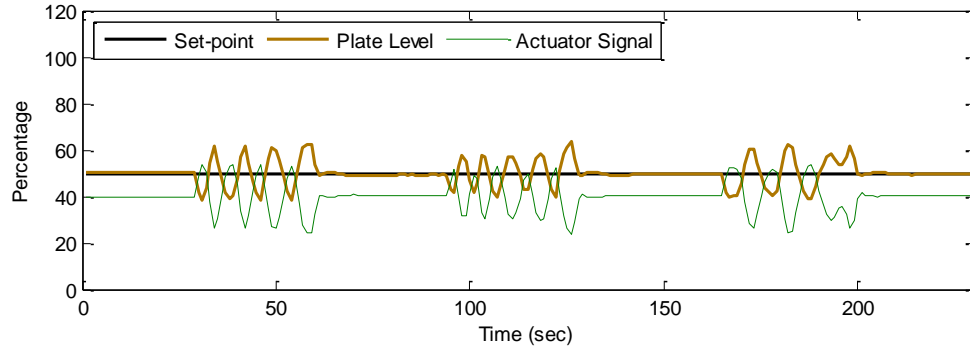
In Figure 5.5(a), the plant is in steady-state and has been running at 1 Hz in MODE-1, i.e. at slower sampling-rate in MODE-1. External disturbances are injected into the system by obstructing the air flow exit manually. By the manual oscillatory obstruction of the air exit path, an oscillatory disturbance pattern has been created on the plate level. Figure 5.5(a) shows the external disturbance causing plate level to oscillate, which, in turn, makes the MPC2 cost to exceed over the threshold limit. It is noticeable that the MPC2 cost goes over the threshold limit (2.0) at the time of disturbance injection. This condition triggers a switching of sampling-rate from 1 Hz to 5 Hz as shown in Fig 5.5(b). Also the Figure 5.5(b) shows that the system returns to the slower sampling-rate when the disturbances subside.



(a)



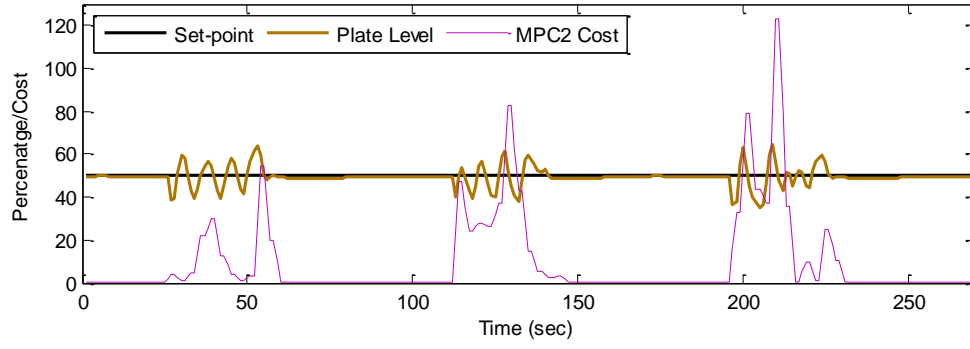
(b)



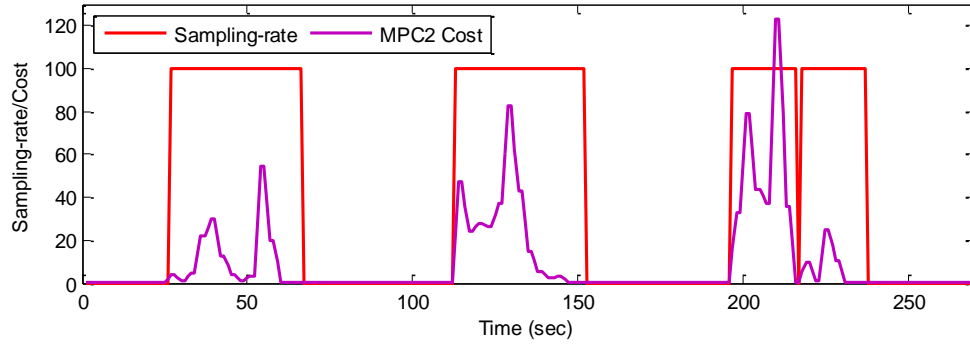
(c)

Fig 5.5: Effect of external disturbance in MODE-1, (a) External disturbance induced cost change, (b) Cost change initiated sampling-rate change (c) External disturbance on plate level induced actuator response in opposite phase

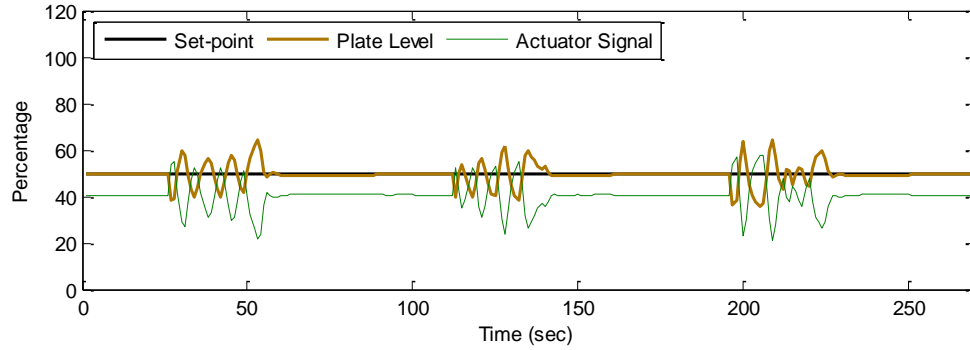
Figure 5.5(c) shows the actuator signal i.e. the control signal of the controller at higher sampling-rate (5 Hz) which is at opposite phase of the plate level as predicted.



(a)



(b)



(c)

Fig 5.6: Effect of external disturbance in MODE-2, (a) External disturbance induced cost change, (b) Cost change initiated sampling-rate change (c) External disturbance on plate level induced actuator response in opposite phase

Figure 5.6 shows the similar situations in MODE-2 i.e. when the controller uses 0.333 Hz as the slower sampling-rate and 5 Hz as the faster sampling-rate.



## 5.7 Effects of Transients on Dual-rated Actuator Signal

This experiment uses both the Trend Display and an oscilloscope to capture the data in MODE-1. The lowest resolution of the Trend Display is 1 sec which is inadequate for getting the 200ms steps of the actuator signal. An oscilloscope is used in parallel for capturing the actuator signal in Fig 5.7.

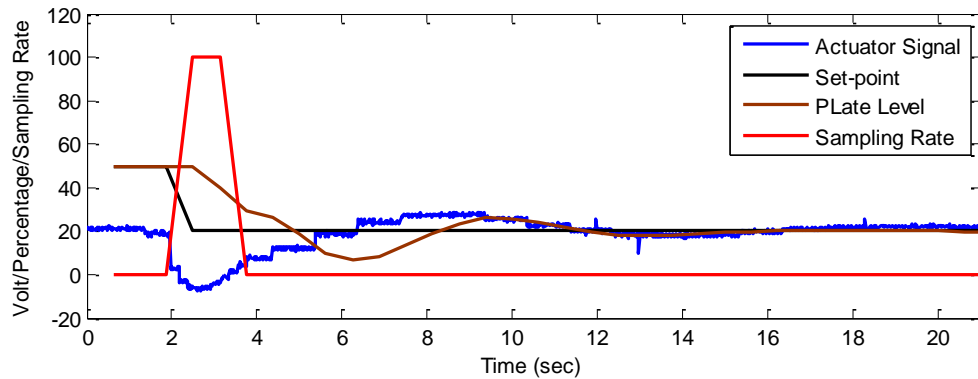
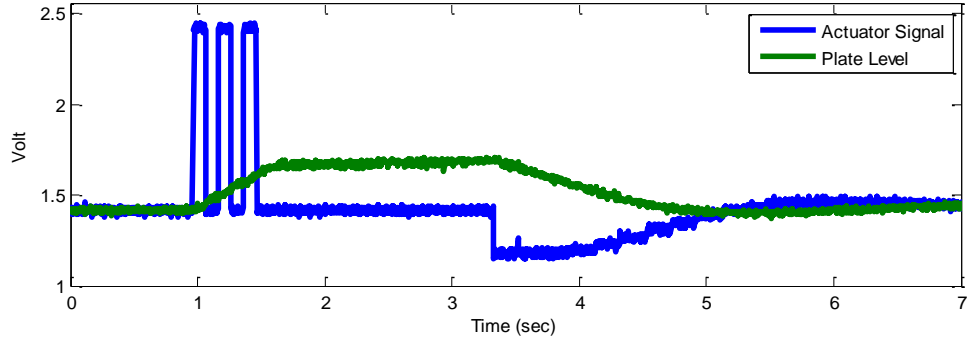


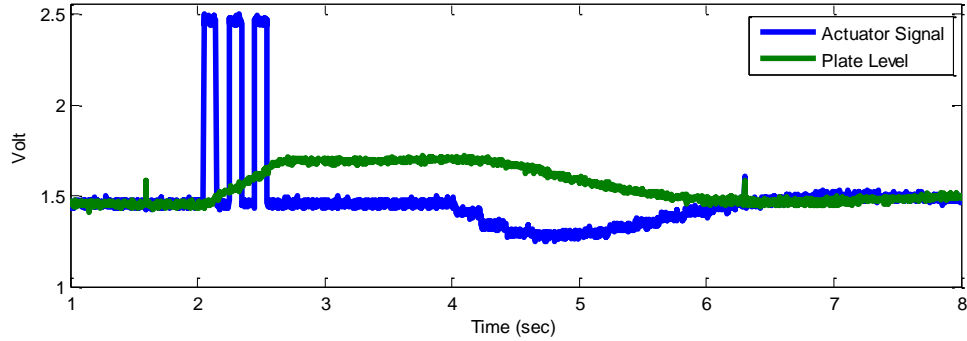
Fig 5.7: Step-change causing dual-rated actuator signal

This experiment shows dual-rate MPC in action. The faster sampling-rate in this controller is 5 Hz and the slower sampling-rate is 1 Hz, i.e.  $n=1/5$ . The plant is running at a set-point of 50% at slower sampling-rate of 1 Hz. At 2<sup>nd</sup> sec a step-change from 50% to 20% has occurred. This transient condition initiates the higher sampling-rate as shown by the red line on Fig 5.7. Also, the blue line obtained from the oscilloscope denoting the actuator signal confirms the sampling-rate change from 1 Hz (1 sec) to 5 Hz (200ms) by the finer steps. In this experiment, the cost threshold is deliberately chosen to be 70.0 instead of 2.0 to cause an early sampling-rate switching from higher to lower value (1 Hz) with sufficient errors for viewing the slower sampling-rate in action. This actually makes the system performance very poor. It is done deliberately just for the verification of the dual sampling-rate in action which is not possible to view clearly when threshold value of 2.0 is used.

## 5.8 Effect of Disturbance in between Samples in MODE-2



(a)



(b)

Fig 5.8: Plate level change on external disturbance pulses on actuator signal causing actuator signal change (a) constant sensor sampling, variable actuator sampling, (b) both sensor and actuator sampling in variable mode

The dual-rate MPC designed automatically switches the sampling-rate with the intention of conserving fieldbus network bandwidth under a steady-state plant condition. This is achieved by switching both sensor and actuator sampling-rates simultaneously to conserve network bandwidth due to reduced sampling-rate with a larger macrocycle. But this conservation of bandwidth can be achieved in a different way by keeping the sensor sampling-rate constant at the faster sampling-rate and by varying only the actuator sampling-rate between the two rates.

The effect of constant sensor sampling-rate (with higher rate, 5 Hz) and variable actuator sampling-rate (0.333 Hz and 5 Hz) based dual-rate controller and variable sensor and actuator sampling-rate (changed simultaneously between 0.333 Hz and 5 Hz ) based dual-rate controller on actuator signal has been studied in this experiment in MODE-2. The switching of the sampling-rate is initiated, in this case, by injecting three short 1000mV pulses added with the actuator signal which ultimately pushed the plate upwards and acted as external disturbance. It can be concluded that, the former case (constant sensor sampling at the faster rate) causes sharp jump as shown in Fig 5.8(b) and the later (both sensor and actuator sampling simultaneously variable in between the two values) produces smoother steps in actuator signal as shown in Fig 5.8(a).

## **5.9 DISCUSSION**

The above experimental results convincingly show that in a process control loop, sampling-rate can be reduced than the optimum rate when the system reaches steady-state condition. And the control action can be restored automatically at the time of transients due to step-changes and external disturbances by switching to a faster sampling-rate. When the transient efforts are over, the system can be switched back to a slower sampling-rate for conserving network bandwidth. As most modern distributed control systems are using fieldbus technology, this scheme of dual (or multiple) sampling-rate can facilitate three important aspects:

- 1) Bandwidth conservation in slow speed fieldbus network
- 2) Energy conservation of smart devices that are connected to the fieldbus
- 3) Reduction of wear in mechanical actuators

### 5.9.1 Network Bandwidth Conservation Mechanism using Dual-rate MPC

In Chapter 2, the working principle of Foundation Fieldbus is discussed where it is noticed that FF has a Link Active Scheduler (LAS) which is in charge of data transfer through fieldbus network. Figure 5.9 shows a simple Foundation Fieldbus based control loop where the smart sensor device is FF capable and connected to the DCS's fieldbus AI module through FF. Similarly, the FF capable smart actuator device is connected to the DCS's fieldbus AO module through fieldbus. In FF, there are two types of data communications, (1) scheduled data transfer for periodic or cyclic data transfer with strict time constraints for sensor and actuator sampling mainly for process control purpose and (2) acyclic and unscheduled data transfer mainly for operation, monitoring, configuration and maintenance purposes. The scheduled and unscheduled data frames together are known as a macrocycle.

Currently, Foundation Fieldbus supports only one fixed macrocycle table within its LAS which is created at the network configuration stage. If this feature of FF can be modified to support more than one macrocycle, which is switchable at runtime through a dual-rate MPC by using a slower sampling-rate, the macrocycle can be increased as shown in Fig 5.10(a) when necessary. As the communication speed remains the same, interval taken by the scheduled data communication also remains the same as the number of devices connected remains the same. Hence, with the increase of the macrocycle made by the dual-rate MPC, more time can be added for acyclic or unscheduled data transfer which is shown in Fig 5.10(b) Thus, more data can be transferred through fieldbus within this timeframe which can be used for operation, maintenance, configuration and diagnostics purposes.

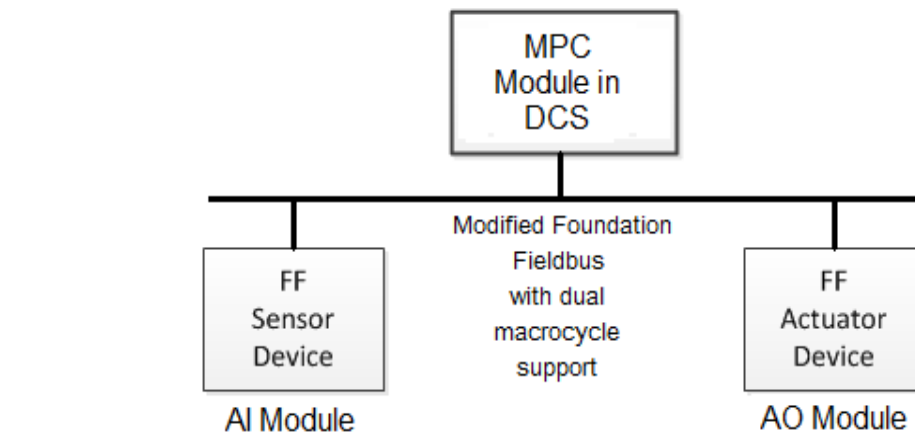


Fig 5.9: Simple Foundation Fieldbus control loop

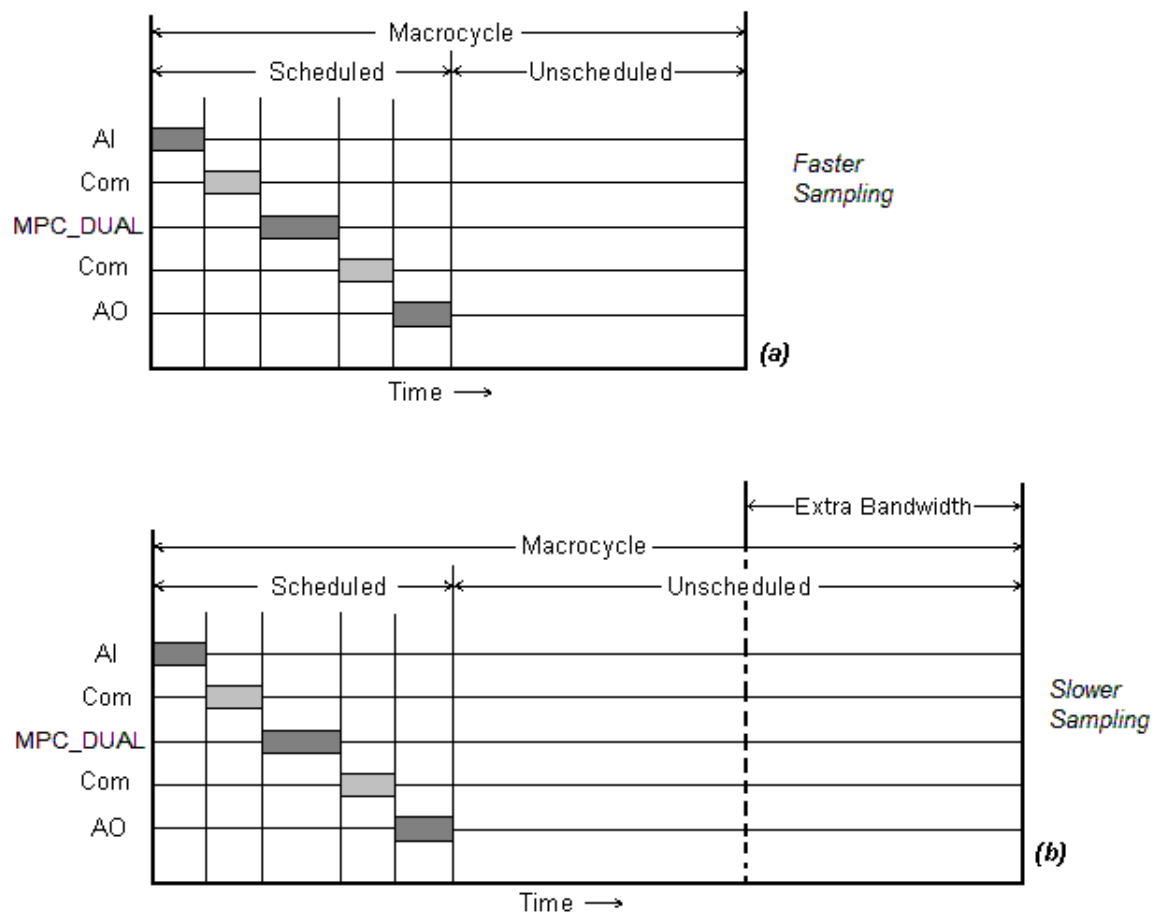


Fig 5.10: Faster and slower sampling-rate effecting bandwidth (a) Macrocycle at fast rate, (b) Macrocycle at slow rate conserving network bandwidth

### 5.9.2 Energy Conservation Mechanism using Dual-rate MPC

Energy is an important issue nowadays. In all new product designs, energy saving capability is strongly considered along with cost and performance. Making a product Green makes it easier to market as people's perception regarding energy conservation has been changing drastically in recent years. Consumers prefer Greener products for more and more cost cutting on electricity bills. Dual-rate MPC also creates opportunity for energy conservation.

Slower sampling-rate means data sent by sensor and received by actuator from controller less frequently. As the macrocycle gets larger, in case of slower sampling-rate, there is an opportunity for energy conservation by sending sensors and actuators in the sleep mode just after the transfer of the required data and re-awaken them when another set of data needs to be transferred. This process then goes on throughout the entire operating range. As these smart sensors and actuators have microcontrollers, putting them to a sleep mode after the required data transfer can save considerable amount of energy in the long run [43].

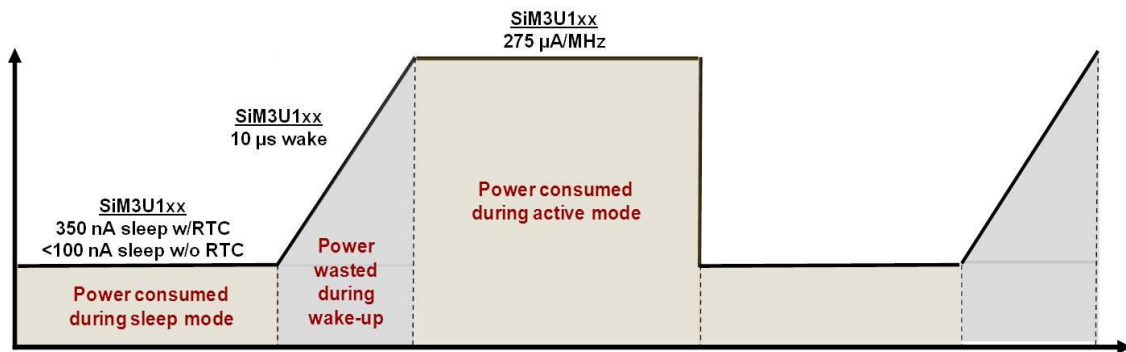


Fig 5.11: Silicon Labs Precision32 SiM3U1xx microcontroller power consumption

Figure 5.11 shows the power requirement of a Silicon Labs Precision32 SiM3U1xx microcontroller. It is seen that the current consumption in the sleep mode with RTC is 350nA and in active mode it is 275μA/MHz of the CPU execution frequency. A very conservative and rough estimation of energy conservation based on the sleep mode based algorithm in a 3.3V based design can be summarized in Table 5.3:

Table 5.3: Power consumption by SiM3U1xx microcontroller

Mode	Current consumption	Power consumption @ 3.3V
Active mode @ 40 M Hz	275 $\mu$ A	36.3mW
Sleep mode	350nA	1.155 $\mu$ W

Table 5.4: Energy requirement in active and sleep mode

Mode of operation	Macrocycle duration	Active duration	Sleeping duration	Energy consumption / hour	Fast sampling-rate activity	Total energy / hour
Active all the time	NA	NA		36.3 mW.h	100%	36.3 mW.h
Sleep mode activated	200ms	100ms	100ms	18.150567 mW.h	20%	6.535 mW.h
	1000ms (5 times slower)	100ms	900ms	3.631mW.h	80%	

It is assumed that the faster sampling-rate can be in action for 20% of the system running time for handling transients and the slower sampling-rate for 80% of the running time of the process in steady-state conditions in a dual-rate MPC based control system. From Table 5.4 it can be seen that in the dual-rate MPC based process control scheme, where the faster sampling-rate is 200ms, and the slower sampling-rate is 1000ms, the energy consumed is 6.535 mW.h instead of 36.3 mW.h when it does not use the sleep mode. Therefore, the energy conserved by utilizing the sleep mode in smart microcontroller based devices can roughly be shown as follows:

$$\text{Microcontroller energy conservation} = 1 - \frac{6.535}{36.3} \times 100\% = 82\%$$

It should be mentioned that in a fieldbus device, there are different components other than the microcontroller which consume energy. The device's energy requirement is not determined only by the microcontroller's energy consumption. In case of sensor devices, the energy consumed by the microcontroller may be 40% to 80% of the energy requirement of the whole device. Hence, by saving microcontroller's energy requirement in sensor devices, considerable amount of energy can be saved.

### **5.9.3 Communication Protocol and Hardware Requirement for Energy Conservation in Field Devices**

To use the sleep mode in fieldbus based devices, in control applications, some special communication frames or commands should be used by the controller to make a device active or awaken before reading from the device and for putting it in the sleep mode after the data has been transferred over the fieldbus. To do that, a specific piece of hardware is required along with the common bus communication controller. Figure 5.12 shows such a device named Device 2 containing a bus controller that can recognize a wake-up frame and can interrupt the microcontroller to wake-up from the sleep mode. Controller Area Networking or CAN a communication technology that probably is the only network technology that is going to support this mechanism with its The ISO 11898-6 specification of "wake-up frame" which is expected to be ratified at the end of 2012 [39].

A dedicated "wake-up frame" is to be transmitted through the network and that can make the device wake-up from the sleep mode which takes typically 10-15 $\mu$ s [39]. While reading from a device or requesting a device for sending data to the controller through fieldbus, every request must be preceded by a wake-up frame to activate the device first. Figure 5.13 shows that the controller is communicating with Device 2 after waking up and the rest of the devices are still in the sleep mode [42]. This sort of strategy is has been used in CAN and LIN for energy conservation [46].



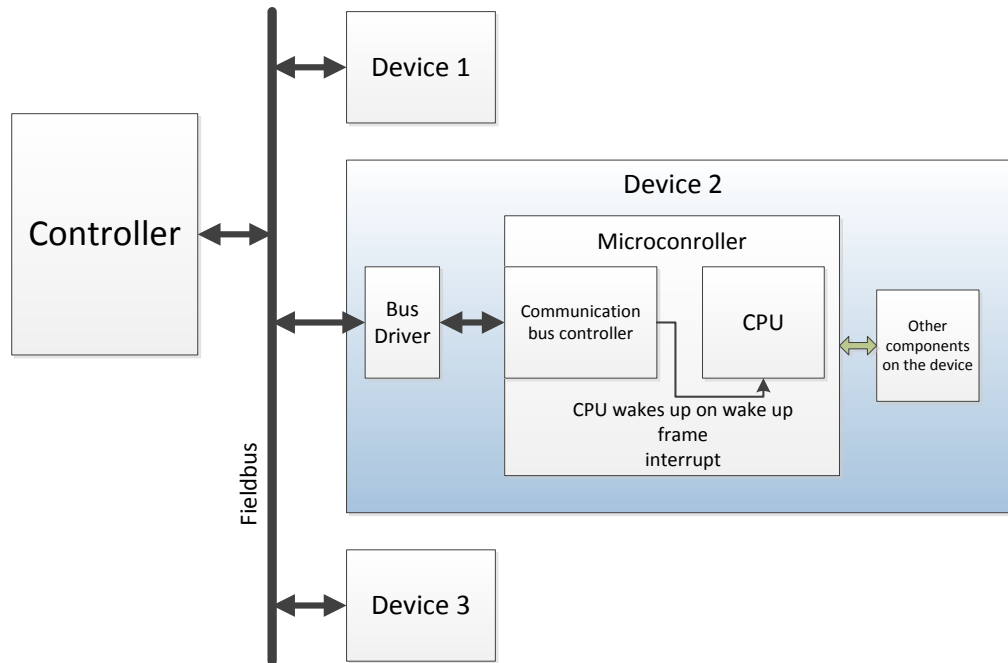


Fig 5.12: Devices having hardware for recognizing wake-up frames

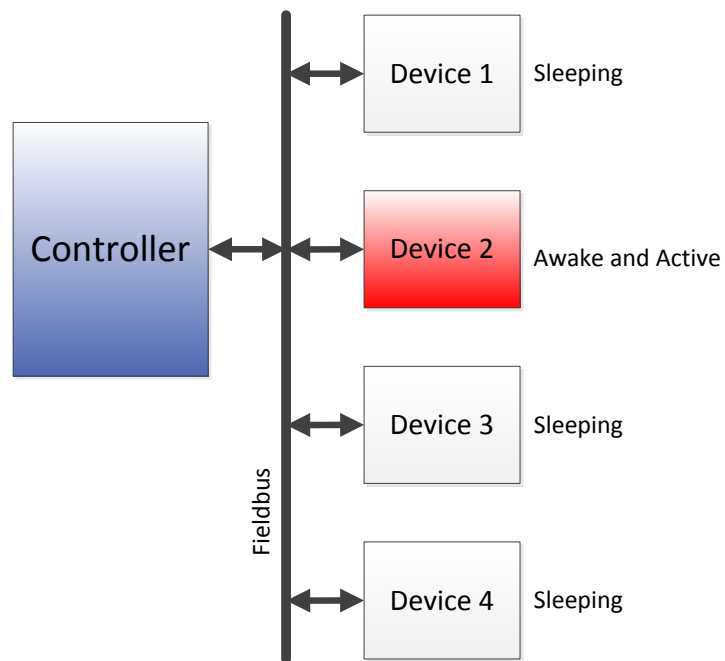


Fig 5.13: Controller communicating with device after waking it up

## Chapter 6

# CONCLUSIONS

### 6.1 Conclusions

In this research, the design and implementation of a dual-rate model predictive controller has been investigated for conservation of network bandwidth in fieldbus based distributed control systems. The main focus of the work is to derive a control scheme that can automatically switch between the two sampling-rates depending on the dynamic states of the system. A model predictive control algorithm with state observer based on Kalman filter has been adopted. For getting a dual-rate MPC, two MPCs are used in parallel with different sampling-rates where one sampling-rate is integer multiple of the other. By continuously monitoring the costs of the two parallel MPCs online and comparing them against a threshold cost, the controller decides which sampling-rate to use by switching to the appropriate MPC output to the actuator.

To demonstrate the effectiveness of the proposed scheme for conserving network bandwidth, Siemens PCS 7 DCS platform with 4-20mA analog communication is used. It is found from the design and implementation of the dual-rate MPC that transients cause the MPC optimization cost to be changed and that can be directly used as an index for automatically switching sampling-rates between faster to slower for network bandwidth conservation in a digital fieldbus network. The switching mechanism works based on comparing with the proper threshold cost.

The dual-rate controller can act in three modes: a) single fast-rate controller, b) single slow-rate controller and c) dual-rate controller. From the quantitative comparisons of the three modes on Table 5.2 it can be seen that in MODE-1, single fast-rate controller and

dual-rate controller has the same overshoot, settling time and  $e_{perf}$  and those are 1.02%, 4 sec and 1.34 respectively. Thus, a dual-rate controller running in MODE-1, having the fast sampling-rate of 5 Hz and the slow sampling-rate of 1 Hz has the same performance as a single fast-rate controller running at 5 Hz all the time. On the other hand, a dual-rate controller running in MODE-2, having the fast sampling-rate of 5 Hz and the slow sampling-rate of 0.333 Hz has slightly poorer performance compared to the single fast-rate controller running at 5 Hz. In MODE-2, the fast-rate controller has the overshoot, setting time and  $e_{perf}$  of 1.02%, 4 sec and 1.34 respectively, similar to that in MODE-1. In dual-rate mode, the overshoot, setting time and  $e_{perf}$  are 1.06, 45 sec, 1.84 respectively and this is not as good as in MODE-1. Therefore it can be concluded that within a dual-rate MPC, the slower the MPC2 sampling-rate becomes the more the performance deteriorates. Thus, the MPC2 sampling-rate should not be too slow to maintain good performance of a dual-rate MPC.

## 6.2 Contributions

The contributions of this thesis can be summarized as follows:

1. This thesis discusses about the use of a dual-rate controller for the bandwidth conservation scheme that can be achieved in a fieldbus based distributed control system. In a dual-rate controller, fast sampling-rate can be used during transients e.g. set-point change or external disturbance to cope the higher control demand but in a fieldbus based DCS, slower sensor and actuator sampling-rate than usual can be used when the system reaches steady-state. This slow sampling-rate in steady-state condition can contribute to some distinctive advantages. One of the major advantages can be network bandwidth conservation which can contribute to additional data transportation capability through the same shared network medium. This additional bandwidth can be used for system operation, diagnostics, configuration, maintenance and other non-real-time data transfer applications. Slower sampling-rate can also

contribute to energy conservation if smart devices with sleep mode in between samplings are adopted in addition with the fieldbus technology that can work with this smart strategy. Slow sampling-rate can also reduce wear of mechanical actuators.

2. The proposed dual-rate controller is designed using Model Predictive Control algorithm for automatically switching between two sample rates is a unique contribution of this thesis. The cost functions of the MPCs calculated online work as an index of system dynamics and comparing with a proper threshold cost initiates sampling-rate switching action according to the demand of the system.
3. To implement the proposed control scheme for achieving desired objectives, Siemens PCS 7 DCS is used which is a widely used DCS in industries throughout the world. The main purpose of this implementation is to demonstrate the ease of implementation of the proposed control scheme in an industrial setting. The reason for selecting Siemens PCS 7 is having some powerful programming tools that support matrix and arrays along with fast sampling-rates amongst prevailing DCSs in the market.

### **6.3 Future Works**

Based on this research, following suggestions are made for future investigations:

- New fieldbus protocol development supporting multi-rate scheduling in the Link Active Scheduler:

The proposed dual-rate model predictive controller is primarily designed for reduction of bandwidth requirement of fieldbus networks. But, unfortunately, no current fieldbus technology supports multi-rate scheduling mechanism in their LAS. Thus, the proposed controller could not be demonstrated in a real fieldbus based

control environment. To use this type of dual-rate or multi-rate controller with bandwidth conservation capability, existing industrial networks should be modified to support such schemes by the implementation of new modifiable scheduling features at runtime.

- Design of a N-rate (multi-rate) controller supporting more than 2 sampling-rates:

This thesis concentrated only on a dual sampling-rate based controller. Similar approach for a multi-rate controller (more than two sampling-rates) can be developed with optimized MPC algorithms and online model modification capability to accommodate different sampling-rates.

- Development of an efficient multi-rate MPC algorithm instead of running single MPCs in parallel:

This research develops a dual-rate MPC based on two single MPC with different sampling-rates (one is N times slower than the other) running in parallel which simplifies design procedure but consumes more CPU resources. For a true multi-rate MPC controller, this approach to design a multi-rate controller is not feasible. An efficient MPC algorithm must be developed which can transform the MPC for different sampling-rate online and can free up CPU resources. This can be done by including sampling-rate as constraints of the MPC equations and calculate the effective and optimized sampling-rate.

- Threshold optimization for switching sampling-rates:

In this thesis, switching between sampling-rates has been performed by comparing the MPC cost with a fixed constant threshold. This threshold value has not been optimized. Additional optimization algorithm can be used for calculating the threshold which may improve the system performance.

- Modification of communication network protocol for supporting energy conservation:

Current communication technologies do not support “wake-up frames” to wake-up devices from the sleep mode prior to the establishment of any communication with that device for conservation of energy. CAN technology has introduced wake-up frames for energy conservation according to ISO 11898-6 [39]. This scheme can be expanded to other industrial network protocols for energy conservation purposes.

- Development of smart devices for energy conservation using slower sampling-rates:

Energy conservation is an important topic of research. Use of slower sampling-rate with appropriate fieldbus technology can conserve energy. By activating the device only when reading measurements and transferring data through the fieldbus and putting them to the sleep mode for rest of the time in accordance with the ISO 11898-6 specification can save energy [44]. Battery operated wireless sensor networks already use this type of method for energy conservation which can also be adopted in the current application [40], [41].

- Reduction of wear and tear on the mechanical actuators

Using reduced sampling-rate when the system reaches steady-state, can reduce mechanical wear of the actuators thus contributing to plant maintenance cost saving. More research can be done in this field as well to see the effect of slower sampling-rate on the actuator wear and tear.

- Study the observer response in external disturbances

In observer based control systems, during external disturbances, the controller tries to cope with the transient situation. While doing so, the observer provides the estimated states of the system to the controller. Therefore, its characteristics affect the

controller. More research can be done based on different characteristics of observers to see their effects on a dual rate MPC in transient disturbances.

## REFERENCES

- [1] “Fieldbus,”  
<http://en.wikipedia.org/wiki/Fieldbus>, date of access: Nov. 26, 2012.
- [2] “Advantages of Fieldbus,”  
<http://kernow.curtin.edu.au/www/Fieldbus/ads.htm>, date of access: Nov. 26, 2012.
- [3] “FOUNDATION HSE,”  
[http://www.fieldbus.org/index.php?Itemid=314&id=138&option=com\\_content&task=view](http://www.fieldbus.org/index.php?Itemid=314&id=138&option=com_content&task=view), date of access: Nov. 26, 2012.
- [4] “FOUNDATION H1,”  
[http://www.fieldbus.org/index.php?option=com\\_content&task=view&id=137&Itemid=313](http://www.fieldbus.org/index.php?option=com_content&task=view&id=137&Itemid=313), date of access: Nov. 26, 2012.
- [5] “Industrial Networks for Communication and Control,”  
<http://anp.tu-sofia.bg/djiev/PDF%20files/Industrial%20Networks.pdf>, date of access: Nov. 26, 2012.
- [6] “Three decades of DCS technology,”  
<http://www.controlglobal.com/articles/2005/227.html>, date of access: Nov. 26, 2012.
- [7] “Distributed Control System Architecture,”  
<http://www.ucos.com/dcs2.htm>, date of access: Nov. 26, 2012.
- [8] L. Q. Li, K. Hanasaki, X. Y. We, Y. B. Peng, Z. Lin and Y. H. Wu, “Integration of Fieldbus into DCS”, in *Proceedings of the 38th Annual Conference SICE*, Aug. 1999, pp.1043-1049
- [9] “Tutorial – Introduction to Fieldbus,”  
<http://verwertraining.com/tutorials/tutorial-introduction-to-fieldbus-and-profibus/>,  
date of access: Nov. 26, 2012.
- [10] “HART Communications,”  
[http://www.samson.de/pdf\\_en/l452en.pdf](http://www.samson.de/pdf_en/l452en.pdf), date of access: Nov. 26, 2012.
- [11] “HART Fieldbus Communication Protocol,”



- <http://www.pacontrol.com/download/Hart-Application-Guide.pdf>, date of access: Nov. 26, 2012.
- [12] “Fieldbus Control Architecture,”  
<http://www.hlg-ltd.com/HLO/FRSI/FRSArch.htm#diags>, date of access: Nov. 26, 2012.
- [13] B. D. O. Anderson and J. B. Moore, “Time-varying feedback laws for decentralized control,” *IEEE Trans. Automat. Control*, vol. 26, pp. 1133-1139, 1981.
- [14] P. P. Khargonekar, K. Poolla, and A. Tannenbaum, “Robust control of linear timeinvariant plants using periodic compensation,” *IEEE Trans. Automat. Control*, vol. 30, pp. 1088-1096, 1985.
- [15] B. A. Francis and T. T. Georgiou, “Stability theory for linear time-invariant plants with periodic digital controllers,” *IEEE Trans. Automat. Control*, vol. 33, pp. 820-832, 1988.
- [16] A. W. Olbrot, “Robust stabilization of uncertain systems by periodic feedback,” *Int. J. Control*, vol. 45, pp. 747-758, 1987.
- [17] S. H. Wang, “Stabilization of decentralized control systems via time-varying controllers,” *IEEE Trans. Automat. Control*, vol. 27, pp. 741-744, 1982.
- [18] M. E. Sezer and D. D. Siljak, “Decentralized multi-rate control,” *IEEE Trans. Automat. Control*, vol. 35, pp. 60-65, 1990.
- [19] E. F. Camacho and C. Bordons. Model Predictive Control in the Process Industry. Springer-Verlag, Berlin, Germany, 1995.
- [20] S. Qin and T. A. Badgwell, “A survey of industrial model predictive control technology,” *Control Engineering Practice*, vol. 11, no. 7, pp. 733 –764, 2003.
- [21] G. C. Goodwin, S. F. Graebe, and M. E. Salgado, Control system design, Prentice Hall, Upper Saddle River, New Jersey 07458, 2001.
- [22] C. E. Garca, D. M. Prett, and M. Morari, “Model predictive control: Theory and practicea survey,” *Automatica*, vol. 25, no. 3, pp. 335 – 348, 1989.
- [23] J. Richalet, A. Rault, J. Testud, and J. Papon, “Model predictive heuristic control: Applications to industrial processes,” *Automatica*, vol. 14, no. 5, pp. 413 – 428, 1978.
- [24] C. Cutler and B. Ramaker, “Dynamic Matrix Control - A Computer Control Algorithm,” *Automatic Control Conference*, San Francisco, CA, 1980.

- [25] K. Natarajan and R. Balasubramanian, "On the Design of LQR Multi-rate Controllers," *American Control Conference*, Seattle, WA, 1986.
- [26] L.P. Wang, *Model Predictive Control System Design and Implementation using MATLAB*, Springer-Verlag, London, UK, 2009.
- [27] N.P. Mahalik, *Fieldbus Technology*, Springer-Verlag, Berlin, Germany, 2003.
- [28] S. Matsutani, T. Zanma, Y. Sumiyoshi, M. Ishida, A. Imura, M. Fujitsuna, "Optimal control of PMSMs using model predictive control with integrator", in *Proc. of the ICROS-SICE International Joint Conference*, Fukuoka (Japan), pp. 4847-4852, Aug. 18-21, 2009.
- [29] N. Cardoso and J. M. Lemos, "Model Predictive control of Depth of Anaesthesia: Guidelines for controller configuration," in *Engineering in Medicine and Biology Society*, pp. 5822-5825, *30th Annual International Conference of the IEEE*, 2008.
- [30] M. C. Berg, N. Amit, and J. Powell, "Multi-rate digital control system design," *IEEE Trans. Automat. Contr.*, vol. 33, pp. 1139–1150, 1988.
- [31] M. Araki and K. Yamamoto, "Multivariable multi-rate sampled-data systems: State-space description, transfer characteristics, and Nyquist criterion," *IEEE Trans. Automat. Contr.*, vol. 30, pp. 145–154, 1986.
- [32] "DC Servo Motor Controller,"  
[http://elm-chan.org/works/smc/report\\_e.html](http://elm-chan.org/works/smc/report_e.html), date of access: Nov. 26, 2012.
- [33] C. L. Phillips, *Digital Control System Analysis and Design*, Prentice Hall, Inc., Englewood Cliffs, NJ, 1995.
- [34] P. N. Paraskevopoulos, *Modern Control Engineering*, Marcel Dekker, Inc., NY, 2002.
- [35] "Model Predictive Control,"  
[http://en.wikipedia.org/wiki/Model\\_predictive\\_control](http://en.wikipedia.org/wiki/Model_predictive_control), date of access: Nov. 26, 2012.
- [36] Holkar K. S., Waghmare L. M., An Overview of Model Predictive Control, *International Journal of Control and Automation*, Vol. 3, Nr. 4, 2010.
- [37] J. Tornero, Y. Gu, and M. Tomizuka, "Analysis of multi-rate discrete equivalent of continuous controller," *The 1999 American Control Conference*, San Diego, CA, July 1999.

- [38] W.W. Chiang, "Multi-rate state-space digital controller for sector servo system," *in proceedings of the 29<sup>th</sup> IEEE Conference on Decision and Control*, pp. 1902 - 1907, Honolulu, HI, December 1990.
- [39] "CAN Bus: Taking a Larger Role in EVs and PHEVs"  
<http://www.digikey.ca/ca/en/techzone/energy-harvesting/resources/articles/can-bus-taking-a-larger-role.html>, date of access: Nov. 26, 2012.
- [40] K. Arshak and E. Jafer, "A wireless sensor network system for pressure and temperature signals monitoring," *in IEEE International Symposium on Industrial Electronics*, pp. 1496 –1501, 2007.
- [41] D. Murali and N. Ida, "A Sampling Method for Reduction of Power in Battery Operated Receivers," *in 11<sup>th</sup> International Conference on Optimization of Electrical and Electronic Equipment*, pp. 47-50, 2008.
- [42] "Internet of Things Could Use Wake Up Benchmark,"  
[http://www.microcontrollercentral.com/author.asp?section\\_id=1741&doc\\_id=242506](http://www.microcontrollercentral.com/author.asp?section_id=1741&doc_id=242506), date of access: Nov. 26, 2012.
- [43] "Choosing The Optimal Low Power MCU,"  
[http://ca.mouser.com/low\\_power\\_choosing\\_mcu/](http://ca.mouser.com/low_power_choosing_mcu/), date of access: Nov. 26, 2012.
- [44] "Standard in development: BS ISO 11898-6 Road vehicles - Controller area network (CAN) - Part 6: High-speed medium access unit with selective wake-up functionality,"  
<http://standardsdevelopment.bsigroup.com/Home/Project/201200714>, date of access: Nov. 26, 2012.
- [45] Y. Wang, S. Boyd, "Fast model predictive control using online optimization," *in proceedings of IFAC World Congress*, pp. 6974-6997, Seoul, South Korea, July 2008.
- [46] "LIN (Local Interconnect Network) Solutions,"  
<http://www.datasheetcatalog.org/datasheet/SGSThompsonMicroelectronics/mXyurtw.pdf>, date of access: Nov. 26, 2012.
- [47] Siemens AG, "SIMATIC, Working with STEP 7, Getting Started," Edition 03/2006.
- [48] Siemens AG, "SIMATIC, S7 SLC V5.3 for S7-300/400, Manual," Edition 01/2005.

- [49] Siemens AG, "SIMATIC NET, S7 CPs for PROFIBUS, CP 443-5 Extend, Manual," Edition 07/2011.
- [50] "Loop Scheduling,"  
<http://www2.emersonprocess.com/siteadmincenter/PM%20Central%20Web%20Documents/Eng%20Sch%20-%20Fieldbus%20103.pdf>, date of access: Nov. 26, 2012.
- [51] Q. Li, D. J. Rankin, and J. Jiang, "Evaluation of delays induced by Foundation Fieldbus H1 networks," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 10, pp. 3684–3692, Oct. 2009.
- [52] N. Cardoso and J. M. Lemos, "Model Predictive control of Depth of Anaesthesia: Guidelines for controller configuration," in *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*, 2008, pp. 5822-5825.
- [53] W. Wroblewski, "Implementation of a model predictive control algorithm for a 6dof Manipulator-simulation results", *Fourth Int. Workshop on Robot Motion and Control*, Puzczykowo, Poland, 2004.
- [54] S Cavalieri. A. Di Stefano, and O. Mirabella. "Optimization of acyclic bandwidth allocation exploiting the priority mechanism in the FieldBus data link layer," *IEEE Trans. on Industrial Electronics*, vol. 40. no 3. pp. 297-306. June 1993.

## APPENDIX A

# HARDWARE COMPONENTS OF SIEMENS PCS 7 DCS AND MODEL PLANT

### A.1 Siemens PCS 7 DCS Hardware Components

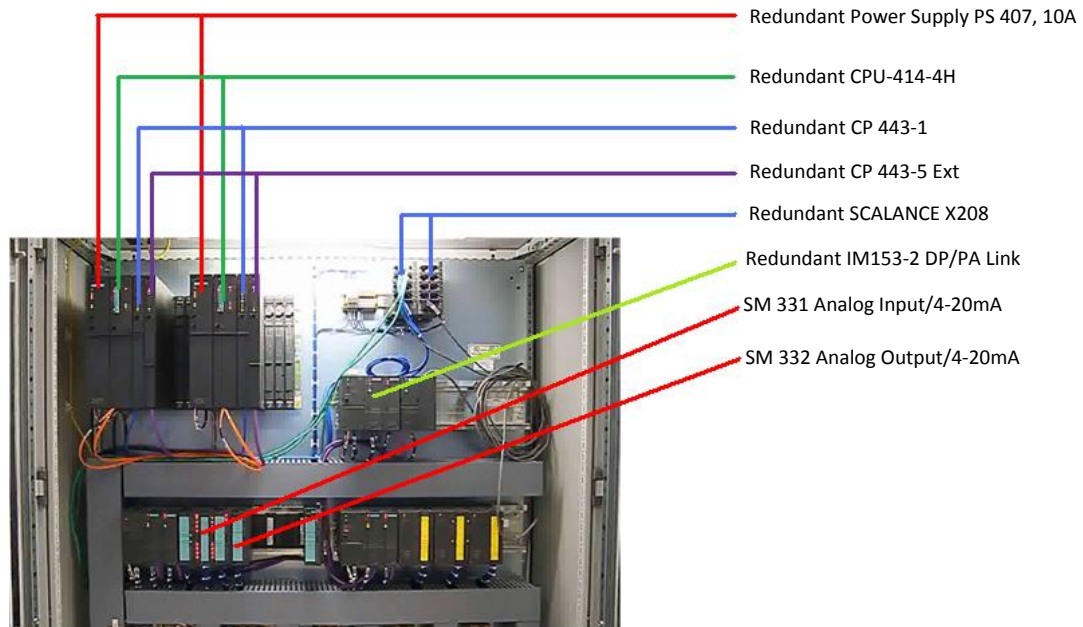


Fig A.1: Siemens PCS 7 DCS components in panel board

Figure A.2 shows various physical components of Siemens PCS 7 DCS that is used in this thesis. The important components and their features are described as follows:

### PS 407 Power Supply:

The PS 407 is a 10 Amp power supply module share the following characteristics in addition to their individual specifications. Its encapsulated design is specially made for use in racks of the PCS 7 system. It maintains cooling via natural convection. Plug-in connection of the supply voltage with AC - DC coding following protection class I (with protective conductor) in accordance with IEC 61140; VDE 0140, Part 1. It has limitation of the inrush current in accordance with NAMUR Recommendation NE 21 and short-circuit-proof outputs as well. Capability of monitoring of both output voltages in case one of these fails, the power supply module signals a fault to the CPU. Both output voltages (5 VDC and 24 VDC) share a common ground. PS 407 also have battery backup as option. The parameters set and the memory contents (RAM) are backed up via the backplane bus in the CPUs and programmable modules. In addition, the backup battery enables you to carry out a restart of the CPU. Both the power supply module and the backed up modules monitor the battery voltage. In PS 407, operating and fault/error LEDs reside on the front plate for monitoring.

### CPU-414-4H:

A high performance CPU for automation & process control having following features:

**Table A.1: CPU-414-4H features**

<b>User memory</b> <ul style="list-style-type: none"><li>• integrated</li><li>• for program</li><li>• for data</li></ul> 1.4 MB 700 KB 700 KB		<b>Memory bits, timers, counters</b> <ul style="list-style-type: none"><li>• Memory bits</li><li>• S7 timers / S7 counters</li><li>• IEC timers / IEC counters</li></ul> 8 KB 2048/2048 SFB/SFB	
<b>Loading memory</b> <ul style="list-style-type: none"><li>• integrated</li><li>• expandable to</li></ul> 256 KB RAM 64 MB		Structure <ul style="list-style-type: none"><li>• Number</li></ul> 21	
		<b>Expansion devices</b> <ul style="list-style-type: none"><li>• Number</li><li>DP master via CP</li><li>• Number FM</li><li>• Number CP 441</li></ul> maximum 10 limited maximum 30	
<b>Number of blocks</b>		MPI/DP interface	

• FC	2048	• DP slaves	maximum 32
• FB	2048	• Transmission rate	up to 12 Mbps
• DB	4095		
DB0 reserved			
<b>Program execution</b>		<b>DP interface</b>	
• Free cycle	1	• Number	1
• Time interrupts	4	• DP slaves	96 each
• Delay interrupts	4	• Transmission rate	up to 12 Mbps
• Watchdog interrupts	4		
• Hardware interrupts	4		
• Startup	2		
<b>Execution times</b>		<b>I/O address area</b>	
• Bit operations	0.06 $\mu$ s	• Total address area	8 KB / 8 KB
• Word operations	0.06 $\mu$ s	• Process image	8 KB / 8 KB
• Fixed-point arithmetic	0.06 $\mu$ s	• Digital channels	65536/65536
• Floating-point arithmetic	0.18 $\mu$ s	• Analog channels	4096/4096

#### CP 443-1:

The CP 443-1 Advanced communications processor is designed for operation in a PCS 7 DCS based on CPU-414-4H. It allows the CPU-414-4H to be attached to an Industrial Ethernet network. To set up small local area networks or to connect several Ethernet devices, a 4-port switch with autocrossing and autosensing has been integrated in the CP 443-1 Advanced Communication Processor [49].

The CP 443-1 Advanced supports the different communication services, such as PROFINET IO Controller which direct access to IO devices over industrial Ethernet. PROFINET CBA is also supported. Use of a SIMATIC PCS 7–414-4H for Component based Automation on the basis of the new PROFINET standard of the PNO. This standard allows component technology in automation, graphic configuration of communication between smart devices instead of laborious programming, Vendor-independent plant-wide engineering etc. It also allows PCS 7 communication with PG functions, operator monitoring and control functions, data exchange over S7 connections. It also has S5-compatible communication features with SEND/RECEIVE

interface over ISO transport connections, SEND/RECEIVE interface over TCP connections, ISO-on-TCP and UDP connections; With the SEND/RECEIVE interface via TCP connections, the CP 443-1 IT, Advanced supports the socket interface to TCP/IP available on practically every end system. Support for Multicast over UDP connection where the multicast mode is made possible by selecting a suitable IP address when configuring connections. Open TCP/IP communication to allow the user program to establish connections with other TCP/IP-compliant communication partners and to exchange data, SIMATIC Manager provides a UDT for the connection parameter assignment and four FBs for high-performance data exchange.

#### **CP 443-5 Ext:**

The CP 443-5 Extended communications processor is designed for use in a PCS 7 DCS and SIMATIC S7-400 (standard) and S7400H (fault tolerant system) automation system. The CP 443-5 Extended allows the S7400 / S7400H to be connected to a PROFIBUS fieldbus system. CP can be used as a router for data records intended for field devices (for example DP slaves). It supports various services as follows:

- PROFIBUS DP with the following characteristics:
  - DP master (class 1) (redundant operation in fault tolerant system also possible)
  - Direct data exchange (DP slave to DP slave)
  - SYNC / FREEZE
  - Constant bus cycle time (only in the standard system)
  - Selectable DP modes
- CiR (Configuration in RUN) - in the standard system
  - By making a change to the configuration with CiR (Configuration in RUN), it is possible to put a DP slave / DP slot extension into operation or take it out of operation when necessary while the system is running.
- Enabling /disabling DP slave



- In the standard system, DP slaves can be activated and deactivated by the user program using system function SFC12.
- Diagnostics requests
  - As a DP master (class 1), the CP 4435 supports diagnostics requests of a DP master (class 2).

### **SCALANCE X208:**

The SCALANCE X-200 Industrial Ethernet switches permit cost effective configuration of Industrial Ethernet line, star or ring topologies with switching functionality where high network availability or remote diagnostics options are required. The devices with IP30 degree of protection have been designed for use in the control cabinet. The SCALANCE X208PRO has the IP65 degree of protection for installation outside the control cabinet.

X208 provides an ideal solution for configuring Industrial Ethernet line, star and ring topologies. Featuring reliable data communication, due to rugged device connection using PROFINET-compatible plug-in cables that offer additional strain relief and bending strain relief thanks to latching on the housing. Its design promises high network availability through configuration of redundant ring topologies with SCALANCE X-400, SCALANCE X-200IRT or OSM/ESM as redundancy managers and fast and easy diagnosis with LEDs on the device, through the integral Web server and through signaling contacts. The integration of the SCALANCE X-208 switches in the existing network management infrastructure through SNMP access point is easy as it is in the process diagnosis and system diagnosis with PROFINET. Configuration and diagnostics integrated into SIMATIC Manager provide significant benefits during the engineering, start-up and operating phases of a plant. Uncrossed connecting cables can be used due to the integrated Autocrossover function with X208. Module replacement can be done without the need for a programming device, using the C-PLUG swap media for backing up the configuration data.

**Analog Input Module, SM 331:**

The analog input module SM 331 has the following features:

- 8 inputs:
  - 2-wire transducer: 4-20 mA
  - 4-wire transducer: 0-20 mA, 4-20 mA, or  $\pm 20$  mA
- Measured values can be smoothed by digital filters in 4 levels (none, weak, medium, or strong)
- Resolution: 15 bit + sign
- Short-circuit-proof
- Wire breakage monitoring

**Analog Output Module, SM 332:**

The analog output module SM 332 used with the PCS 7 DCS has the following features:

- 8 outputs: 0/4-20 mA
- Resolution: 15 bit + sign
- Scan time (all channels) at AO mode: 10 ms
- Settling time at AO mode and resistive load: 0.1 ms
- Settling time at AO mode and inductive load (10 mH): 0.5 ms
- Wire breakage monitoring

**A.2 Siemens PCS 7 Software Components**

The main tool for programming and configuring Siemens PCS 7 DCS is SIMATIC Manager. This is an IDE that handles all the software components for programming and configuring PCS 7 DCS. This tool contains various software components and some of the important ones that have been used in this thesis are as follows:

Table A.2: PCS 7 software development tools

Name of the tool	Description
SCL Editor & Compiler	Structured Control Language editor and compiler. Creates Function Blocks, Data Blocks etc. written in SCL high level language
SIMATIC CFC	Uses readymade function blocks from the library or custom designed SCL, Ladder Logic or other languages
Hardware Configuration	Used for configuring all the hardware components of the DCS e.g. I/O modules, communication modules, protocol converters, CPU features, fieldbus control loop characteristics, macrocycle duration etc.

### A.3 The model plant

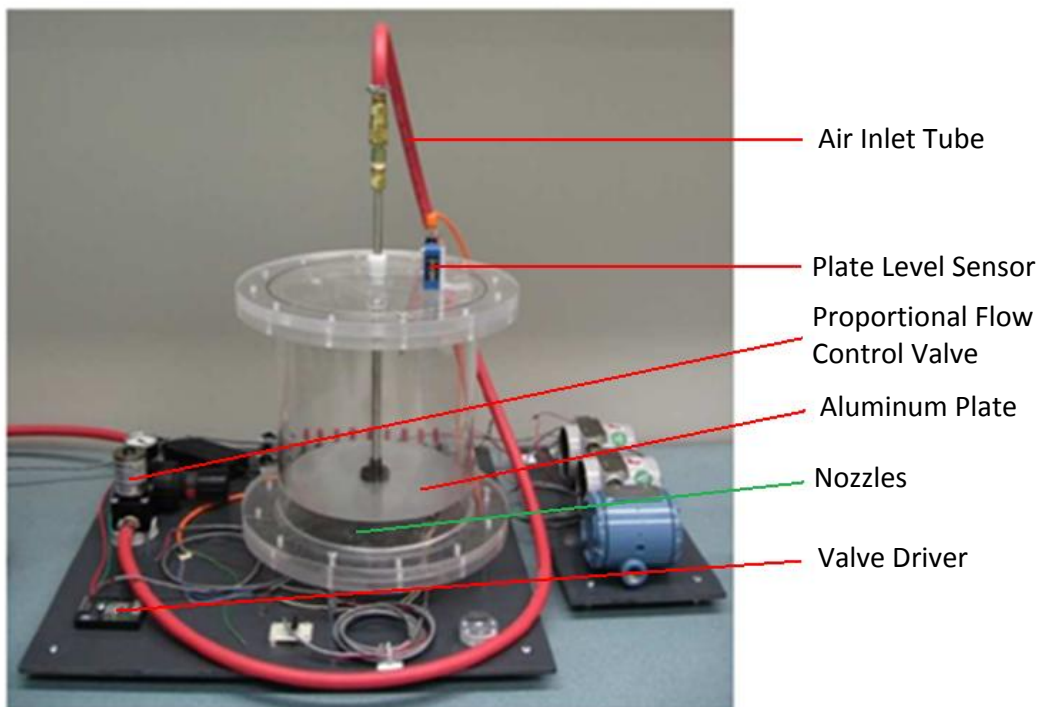


Fig A.2: Model Plant Components

The level process shown in Fig A.2 consists of a cylinder and an aluminum plate that can move vertically to allow air to pass through the gap between the plate and the cylinder wall. Air nozzles are distributed evenly under the plate and compressed air can pass through the nozzles to lift the plate upwards. A proportional flow control valve regulates the air supply through the nozzles. The plate level is measured by a LASER based level sensor located on top of the cylinder. A hole resides on top of the cylinder for exhausting the air to maintain equilibrium.

**Air inlet tube:** The inlet tube, a stainless steel tube, is connected with the proportional air flow control valve with a rubber tube. This tube also works as a sliding guide of the aluminum plate. At the end, it is connected to some nozzles facing upwards which are distributed evenly for pushing the aluminum plate upwards.

**Plate level sensor:** Wenglor CP35MHT80 is a LASER displacement sensor which uses a high-resolution CMOS line array and DSP technology, virtually eliminating material, color and brightness related measurement value differences. The features of the sensor are as follows:



(a)



(b)

Fig A.3: (a) Wenglor CP35MHT80 is a LASER displacement sensor (b) Teknecraft 203314 Proportional flow control valve

- Range: 300mm (from 50-350mm)
- Two modes: speed mode or resolution mode
- Resolution < 50  $\mu\text{m}$ ; Resolution (Speed-Mode): < 80  $\mu\text{m}$
- Linearity: 0.15 %; Linearity (Speed-Mode): 0.2 %
- Supply Voltage: 18-30 V DC
- Current Consumption ( $U_b = 24\text{V}$ ): < 80 mA
- Cut-Off Frequency: 400 Hz; Cut-Off Frequency (Resolution-Mode): 200 Hz
- Response Time: < 1.25ms; Response Time (Resolution Mode): < 2.5ms
- Temperature Drift: < 25  $\mu\text{m}/^\circ\text{C}$
- Temperature Range: (-25)-(-50)  $^\circ\text{C}$
- Analog Output: 0-10 V; Current Load Voltage Output: < 1 mA
- Analog Output: 4-20 mA; Current Output Load Resistance: < 500 Ohm
- Interface: RS-232 at the Baud Rate of 38400

### **Proportional Flow Control Valve:**

Teknocrat 203314 Proportional flow control valve features:

- 0.375 inch orifice, 24 volt, Aluminum Material
- Size: inlet 3/8 NPT, outlet 1/2 NPT
- Time response: <35ms
- Inlet pressure 0 to 120psi, outlet pressure is 25% of inlet pressure
- Teknocrat's solenoid is truly proportional that is, the position of the solenoid armature (which also acts as the poppet) is extremely linear relative to the input current. As can be seen in Fig A.4, linearity is maintained even at the high and low end of the input signal.

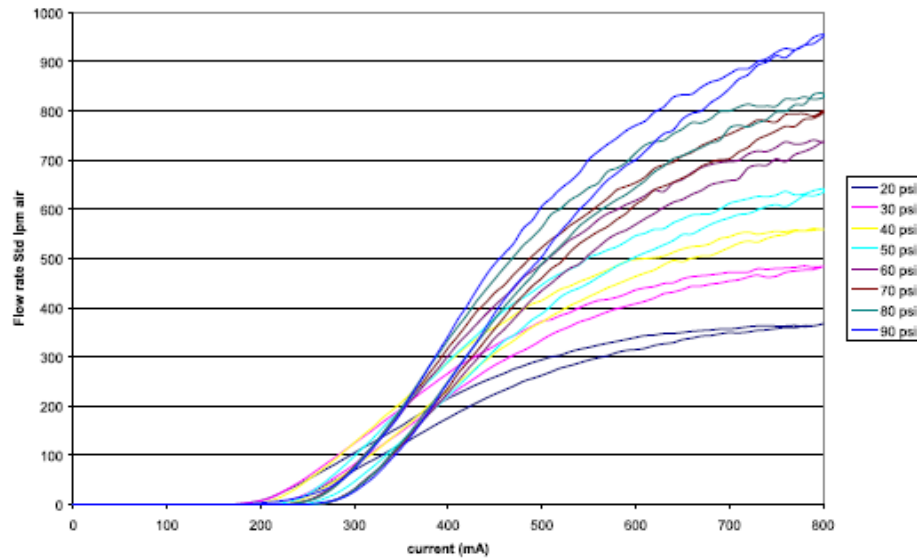


Fig A.4: Flow curve for 0.375 balanced 24V coil proportional solenoid valve

**Aluminum Plate:** The aluminum plate slides over the stainless steel tube on a nylon bush. The plate is enclosed within a transparent cylinder and there is a minute gap (few mm) between the aluminum plate and the cylinder wall so that some air can flow in between the plate and the cylinder.

**Nozzles:** Nozzles are spread evenly in a star shape and have a common inlet at the center which is connected to the air inlet tube which also guides the sliding aluminum plate. When the proportional air flow control valve opens and air flows through the nozzles which create sufficient pressure to lift the aluminum plate.

**Proportional Flow Control Valve Driver:** Teknocraft 9-B5950-007 signal amplifier features are as follows:

- Power supply: 9-32 VDC
- Input control signal: 4-20mA or 1-5V DC.
- Output current: 2A
- PWM based dither control function makes the proportional valve not jam into a fixed position

## APPENDIX B

### MODEL PLANT TRANSFER FUNCTION

#### B.1 Mathematical model of the plant

To determine the dynamic relationship between the process input and output in the form of transfer function, the following procedure has been used:

Table B.1: Plant response at different frequency

Frequency ( Hz)	0.05	0.075	0.1	0.125	0.15	0.175	0.2	0.225
Amplitude (mA)	0.616	0.409	0.301	0.239	0.197	0.167	0.147	0.130

The plant has been put in an oscillatory function with some fixed frequencies as on the **Table B.1** and the amplitude is measured in terms of mA through an oscilloscope. This current signal is proportional to the plate displacement [51].

$$\frac{X(s)}{U(s)} = \frac{k}{ms^2 + bs} = \frac{\frac{k}{m}}{s\left(s + \frac{b}{m}\right)}$$

By using the above formula the plant model can be found as follows:

$$\frac{X(s)}{U(s)} = \frac{1.5}{s^2 + 3.7s}$$

## APPENDIX C

# INTRODUCTION TO SIEMENS SIMATIC MANAGER FOR DEVELOPING DUAL-RATE MPC USING PCS 7

### C.1 Creating a New Project on PCS 7 SIMATIC Manager

A new PCS 7 multi-project can be created using PCS 7 wizard's "New Project" with a few mouse clicks. The prerequisite to do that is to have installation of PCS 7 as of V6.0 [47]. The procedures are as follows:

Table C.1: New project creation procedure

Step	Procedure
1.	Starting the SIMATIC Manager. Either by double-clicking on the corresponding icon on the desktop or by using the command <b>Start SIMATIC Manager</b>
2.	Closing any projects which may be opened by using the menu command <b>File &gt; Close</b>
3.	Starting the PCS 7 Wizard by using the menu command <b>File &gt; "New Project" Wizard</b>
4.	Leaving the presettings in the initial dialog box (1/4) and click on the "Next" command button.
5.	Selecting the CPU414-4H in the wizard dialog box (2/4) and click on the "Next" command button.
6.	<ul style="list-style-type: none"><li>▪ In the wizard dialog box, (3/4) selecting "3" layers for the Plant Hierarchy.</li><li>▪ Under PLC Objects activate the "CFC Chart" and "SFC Chart" check boxes.</li><li>▪ Under OS Objects activate the "PCS 7 OS" check box and select the "Single workstation system" radio button. Then click on the "Next" command button.</li></ul>



7.	In the wizard dialog box (4/4) entering "fast" under the directory name, specify the storage site or leave the default setting and click on the "Complete" command button.
8.	<p>The "Message Number Assignment Selection" dialog box is only displayed if not have changed the default setting "Always Prompt Settings" for new projects in the "Select" dialog box.</p> <p>Selecting the "Unique message number assignment for the CPU" radio button and terminate the dialog box by clicking on the "OK" dialog box.</p> <p>Furthermore it can be specified default settings for future projects/libraries in this dialog box. If specified a default differently from the standard default setting, the prompt for the message number assignment selection in the PCS 7 Wizard is no longer displayed</p>
9.	Closing the Plant View and open the Process Object View by using the <b>View &gt; Process Object View</b> menu command.
10.	In the SIMATIC Manager using the <b>Window &gt; Arrange &gt; Horizontally</b> menu command to position the two windows "Process Object View" and "Component View" underneath each other.

As an example, a project named "fast\_MP" (as an example) multiproject with a SIMATIC 400 station, a SIMATIC PC station and a master data library "fast\_Lib" can be created by following the above procedure. After the above-mentioned steps have been completed, the created multiproject is displayed in the Project Object View and in the Component View in the SIMATIC Manager. The SIMATIC Manager window is shown after following such a procedure for creating project named "fast\_MP" in Fig C.1. After creating a new project system hardware must be configured, which is discussed in section C.2 of this chapter.

In the dual-rate MPC application, similar procedure has been followed and the project named Example\_Arif10\_10Np\_15xU (this is the dual-rate MPC project) has been created and the Component view is shown as in Fig C.2(a), where the MPC\_DUAL on the right hand side box is a SCL source containing the dual-rate MPC code. If the MPC\_DUAL is clicked, the SLC editor and compiler window opens as shown in Fig C.2(b). After writing the code for MPC\_DUAL function block and other related function blocks required for

the project, clicking the **compile** button shown in Fig C.2(b), the SCL code will compile and the compiler generates error and warning messages on the bottom window of Fig C.2(b).

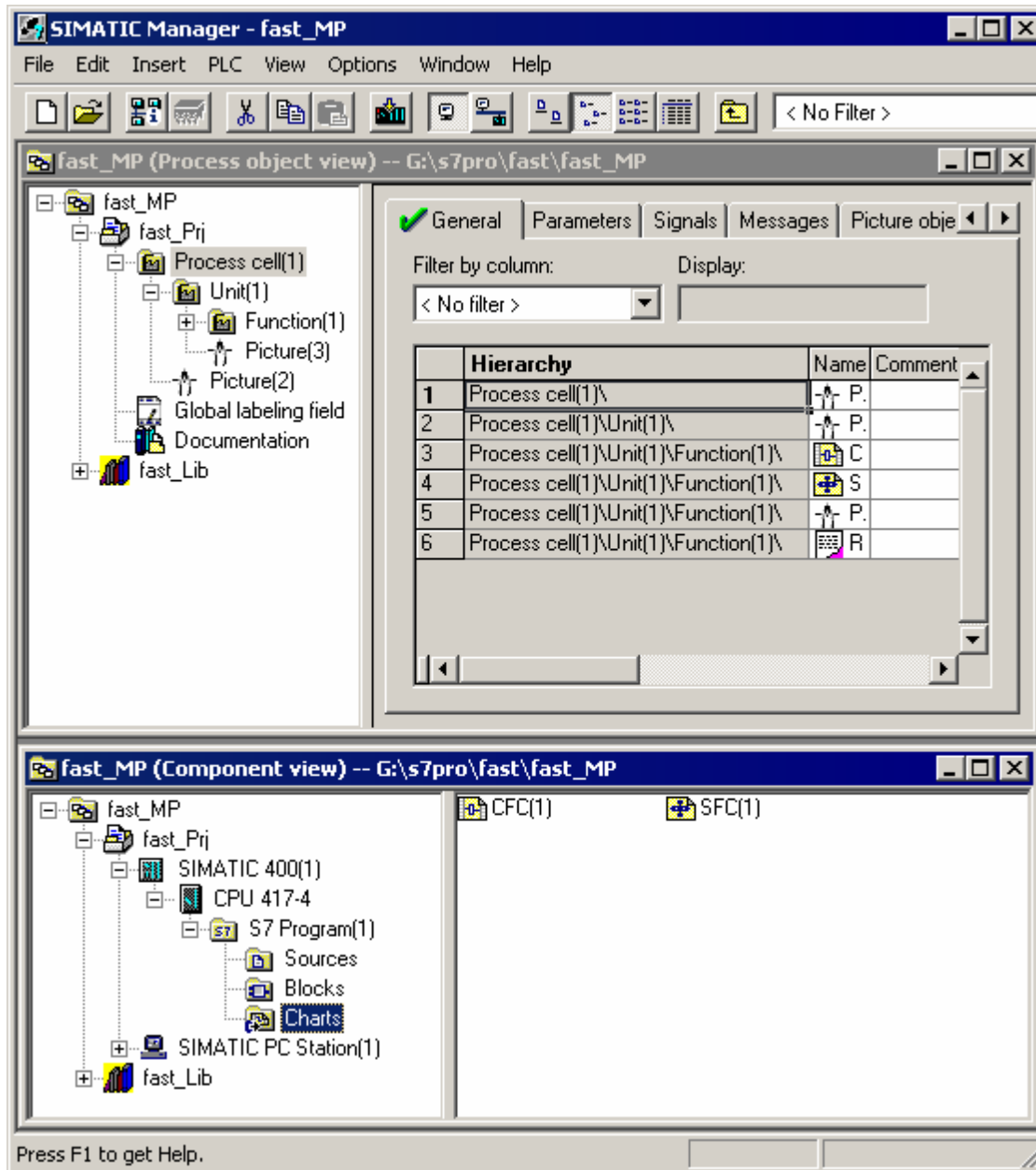
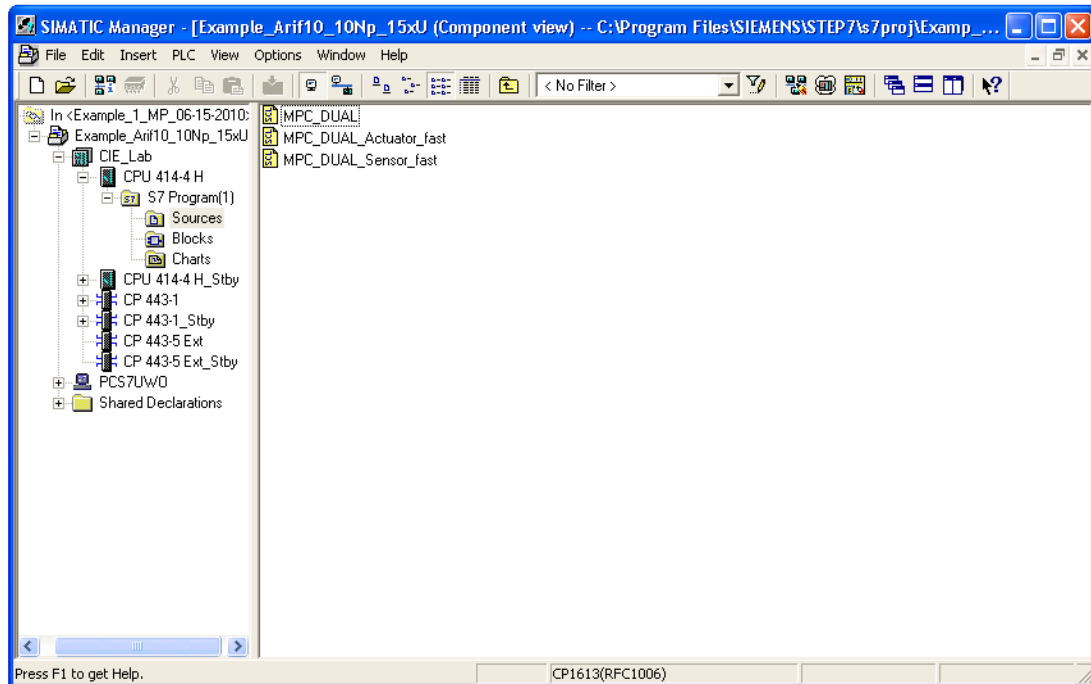
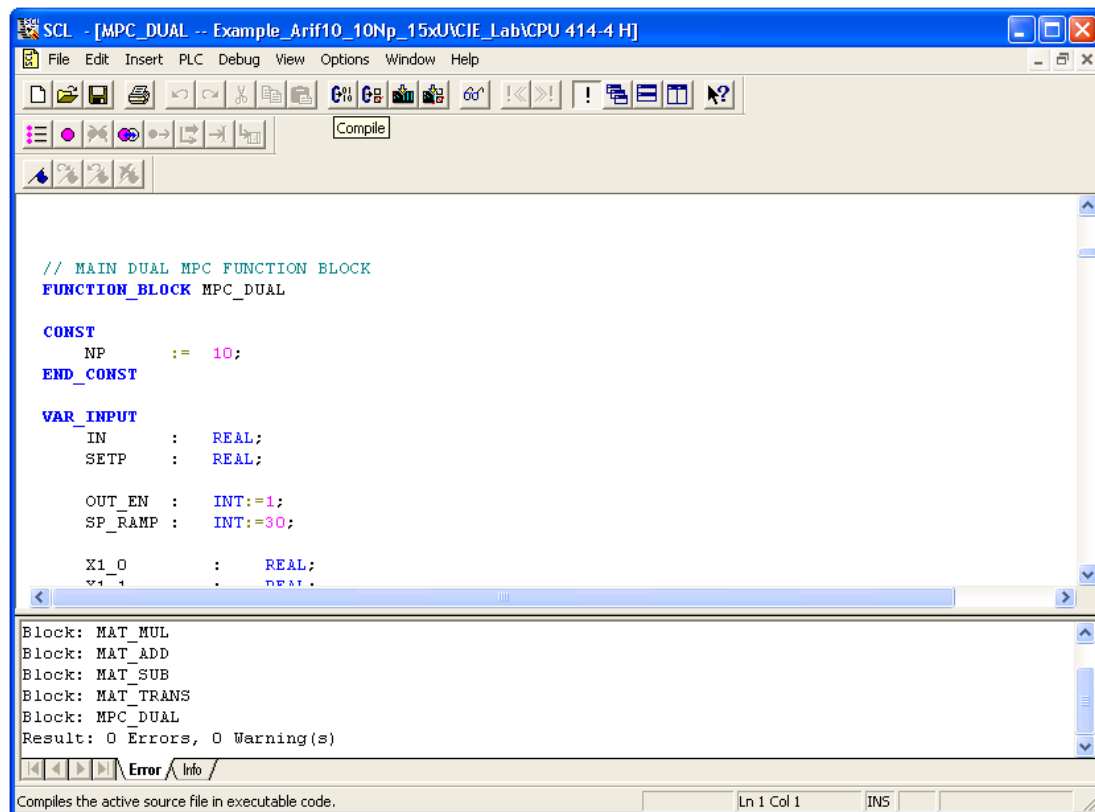


Fig C.1: Creating a new project in SIMATIC Manager



(a)



(b)

Fig C.2: (a) Dual-rate MPC project on SIMATIC Manager (b) SCL editor and compiler

The MPC\_DUAL SCL file contains the main MPC\_DUAL function block and the supporting data and function blocks are shown as follows:

Table C.2: MPC\_DUAL SCL source file modules

Name	Type	Visibility	Explanation
MAT	DB	Internal	Common matrix for data transfer between other function blocks
MAT_MUL	FB	Internal	Function block for 10x10 matrix multiplication
MAT_ADD	FB	Internal	Function block for 10x10 matrix addition
MAT_SUB	FB	Internal	Function block for 10x10 matrix subtraction
MAT_TRANS	FB	Internal	Function block for 10x10 matrix transpose
MPC_DUAL	FB	External	Dual-rate MPC function block exportable to CFC for use with other modules

The successful and error free compilation and generation of function blocks also demand that proper symbols for function blocks, data blocks, organization blocks etc. are correctly defined on the **Symbol Editor** by clicking **Options>Symbol Table** on the SCL Editor as in Fig C.3 where it shows that function block FB27 has the symbol named MPC\_DUAL, DB 27 has DUAL\_MPC\_DATA, FB21 has MAT\_ADD and so on.

After proper compilation of the SCL code using the appropriate symbols through Symbol Editor, the generated function blocks and other blocks are visible on the right side window of SIMATIC Manager by clicking its **'Blocks'** item under the **CPU 414-4H>S7 program(1)** which is shown in Fig C.3 as FB27 which is the MPC\_DUAL function block generated previously on the SCL compiler.

After developing the MPC\_DUAL function block and its supporting function blocks and data blocks, clicking **Charts** item under the **CPU 414-4H>S7 program(1)** of SIMATIC Manager window opens the CFC Editor shown in Fig C.4.

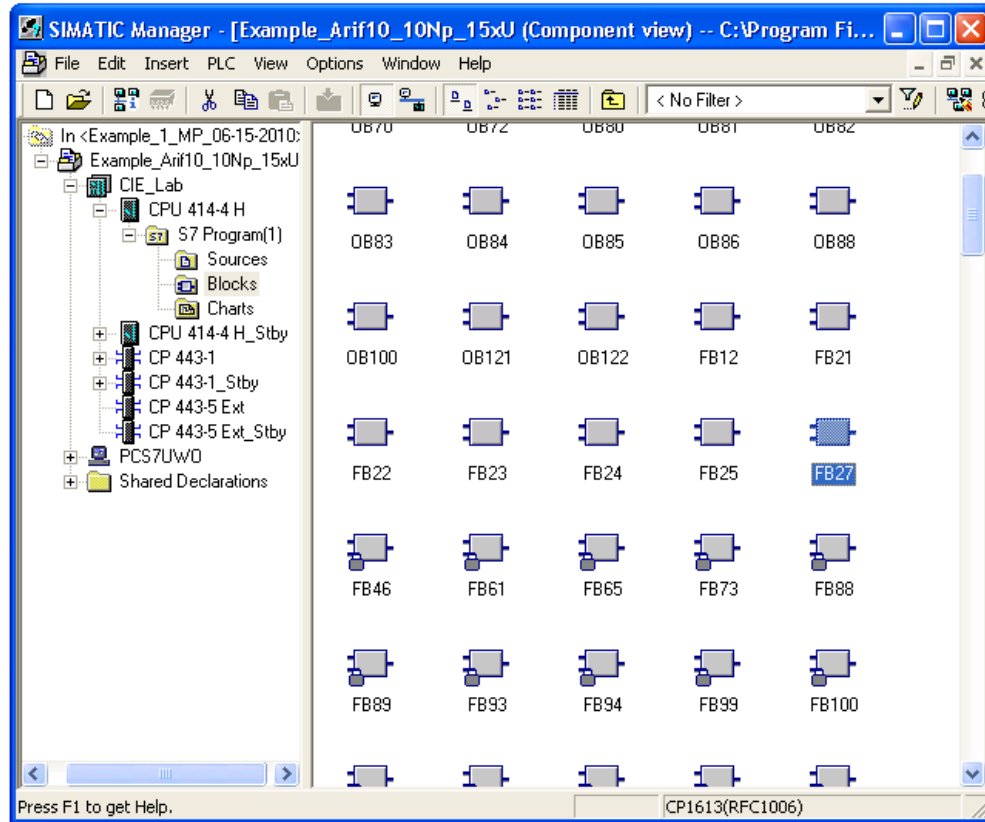


Fig C.3: Newly generated MPC\_DUAL as FB27

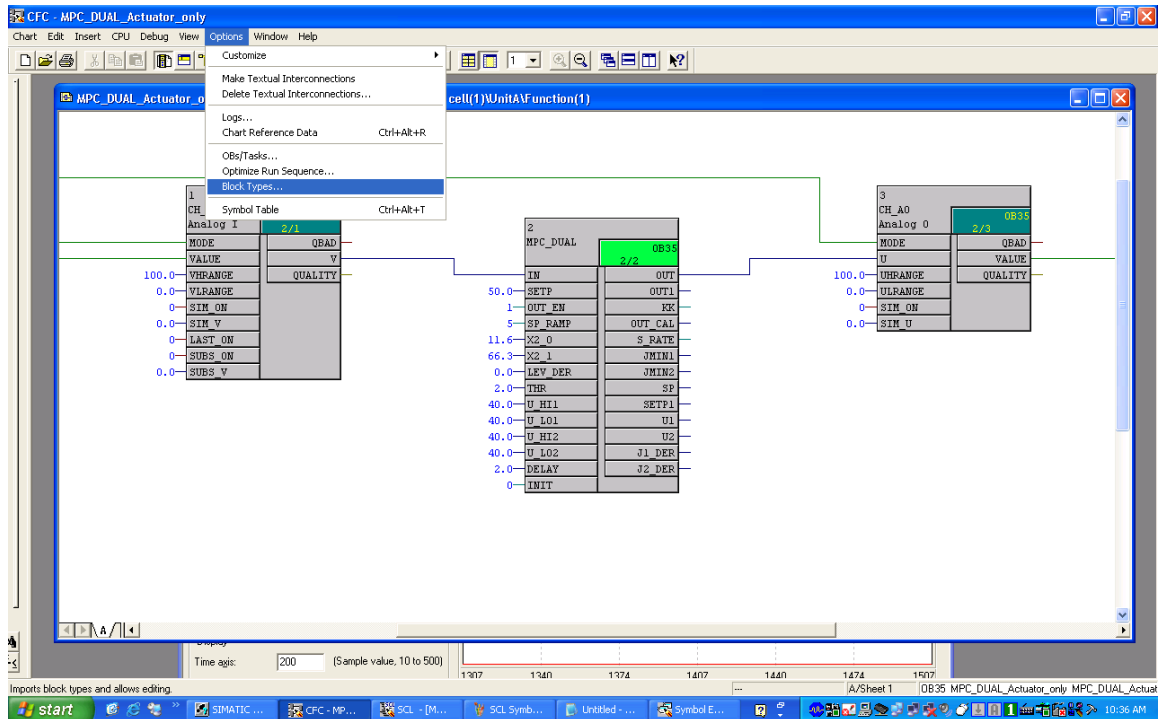


Fig C.4: SIMATIC CFC showing dual-rate MPC application

First, after entering CFC, the custom function blocks e.g. MPC\_DUAL and its associated function blocks are imported in the CFC by clicking menu item “**Options>Block Types ...**” which brings the Block Types window as follows in Fig C.5 and importing MPC\_DUAL function block from left ‘Block folder offline’ to ‘Chart folder’. After doing so, required function blocks e.g. CH\_AI, MPC\_DUAL and CH\_AO are imported by clicking on the ‘Insert’ menu item and selecting appropriate items for inserting them into the chart. After inserting the function blocks, connections are made by clicking on the appropriate block parameters of the inserted function blocks and setting values, modes and visibility options etc. as shown in Fig C.4.

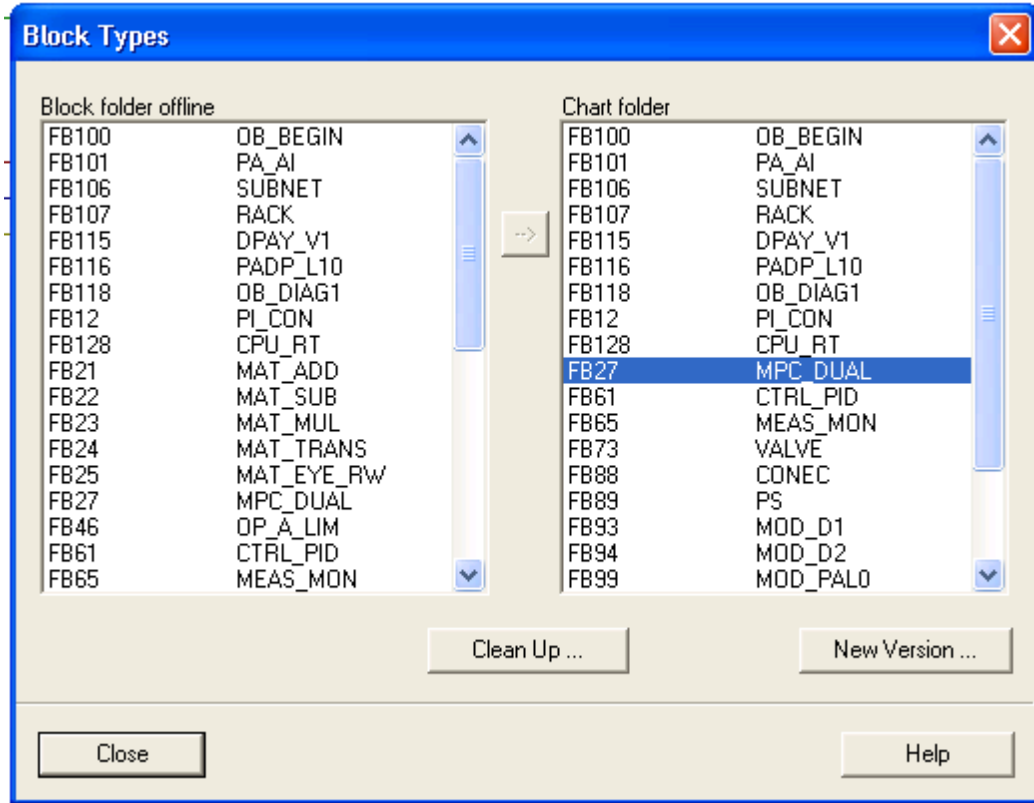


Fig C.5: Block Types child window for importing function blocks to CFC

When connections are properly made between the function blocks and parameters set appropriately, the Continuous Function Chart (CFC) just created is ready for compiling the chart. Successful compilation can then allow the code to be downloaded to the CPU for execution.

## C.2 Hardware Configuration

Hardware configuration is generally done after the creation of the new project. It is discussed later for the convenience of describing the application development process by maintaining a constant flow of description started from the previous section. Without proper hardware configuration, the application might not get downloaded at all and end up resulting some error messages.

In addition to the objects, the PCS 7 Wizard has created, further hardware modules from hardware catalog may be required. These configuration steps are carried out in the **HW Config** item on the SIMATIC Manager and shown in Fig C.1. The prerequisite of doing so is to have a project opened e.g. “fast\_MP” in Fig C.1.

### C.2.1 Example Procedure of Hardware Configuration

Table C.3: Steps for hardware configuration

Step	Procedure
1.	In the Component view, select the SIMATIC 400 station using the right-and mouse button and click on "Open Object" in the pop-up menu.
2.	<ul style="list-style-type: none"> <li>• In <b>HW Config</b> open the "PROFIBUS-DP" folder on the right-hand side in the hardware catalog.</li> <li>• Then opening the "ET 200M" folder and select "IM 153-2" to drag and drop this module onto the line of the PROFIBUS DP Master system while keeping the left-hand mouse button pressed.</li> <li>• Acknowledging the "Properties - PROFIBUS Interface IM 153-2" dialog box by clicking on the "OK" command button.</li> </ul>
3.	<ul style="list-style-type: none"> <li>• Opening the "IM 153-2" folder in the hardware catalog by clicking on the plus icon before this folder.</li> <li>• Opening the "AI-300" folder contained in it and drag the module "SM 331" AI8x12Bit" (6ES7 331-7KF01-0AB0) to Slot 4 at the bottom of the configuration table of the IM 153-2 while keeping the left-hand mouse button pressed. "AI-300" folder" is then closed</li> <li>• Opening the "AO-300" folder contained in it and drag the module "SM 332 AO2x12Bit" (6ES7 332-5HB01-0AB0) to Slot 5 at the bottom of the configuration table of the IM 153-2 while keeping the left-hand mouse button pressed. Closing the "AO-300" folder".</li> <li>• Opening the "DI-300" folder contained in it and drag the module "SM 321</li> </ul>



	<p>DI16xDC24V" (6ES7 321-1BH01-0AA0) to Slot 6 at the bottom of the configuration table of the IM 153-2 while keeping the left-hand mouse button pressed. Closing the "DI-300" folder".</p> <ul style="list-style-type: none"> <li>• Opening the "DO-300" folder contained in it and drag the module "SM 322 DO8xDC24V/2A" (6ES7 322-1BF01-0AA0) to Slot 7 at the bottom of the configuration table of the IM 153-2 while keeping the left-hand mouse button pressed. Closing the "DO-300" folder".</li> </ul>
--	--

In HW Config, the redundant interface IM 153-2 has been inserted for the ET 200 M peripheral devices as a distributed peripheral device. The ET 200M has been extended with analog/digital input/output modules. Figure C.6 shows the example hardware configuration done above.

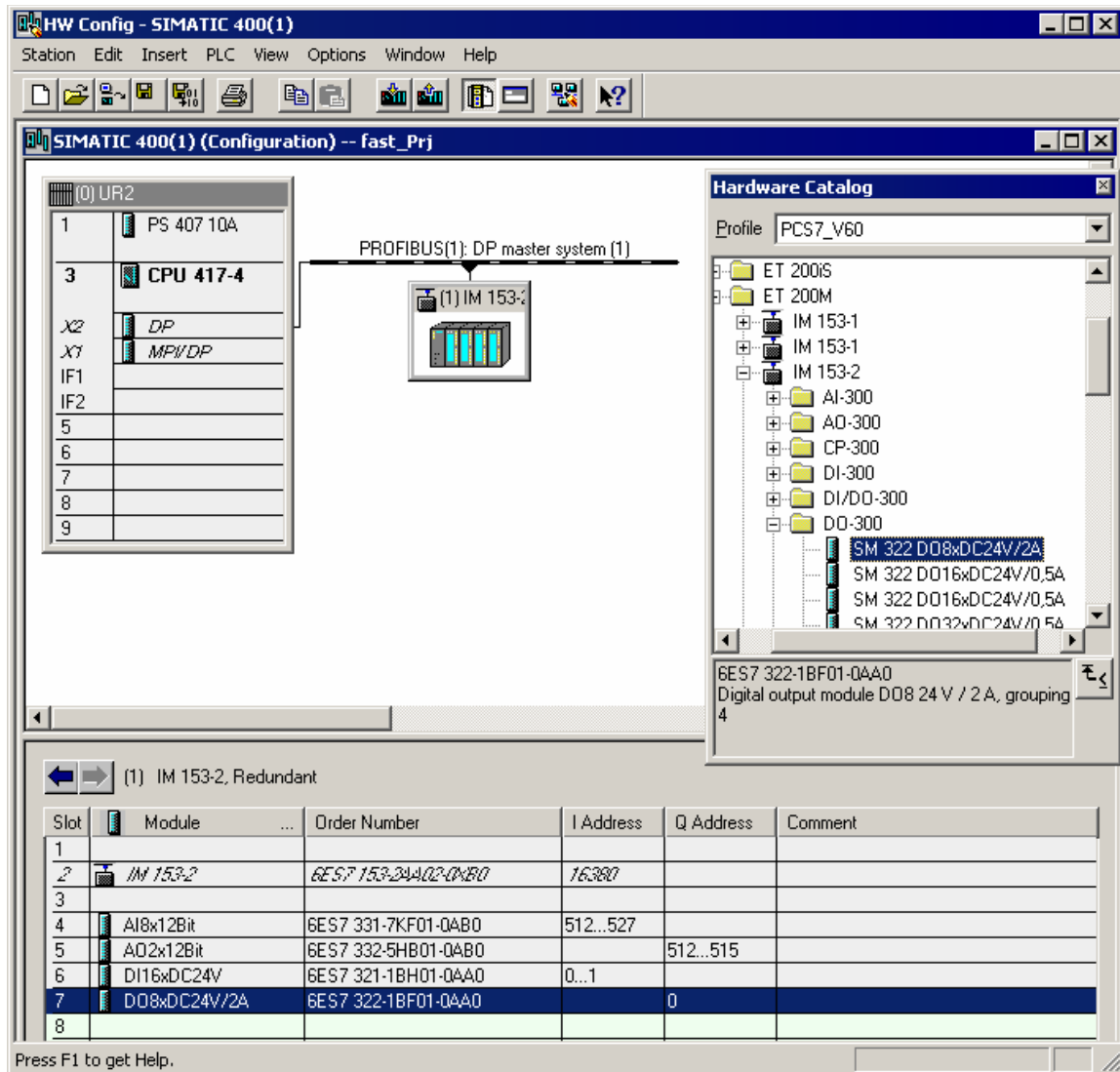


Fig C.6: General hardware configuration procedure in HW Config

In this thesis, a dual redundant CPU with dual redundant IM-532 have been used as shown in Fig A.1. The HW Config used in dual-rate MPC project is shown in Fig C.7. The CPU has been used is a CPU 414-4 H.

One of the most important hardware configuration features has been used in the dual-rate MPC project is setting the macro cycle time of the control loop.

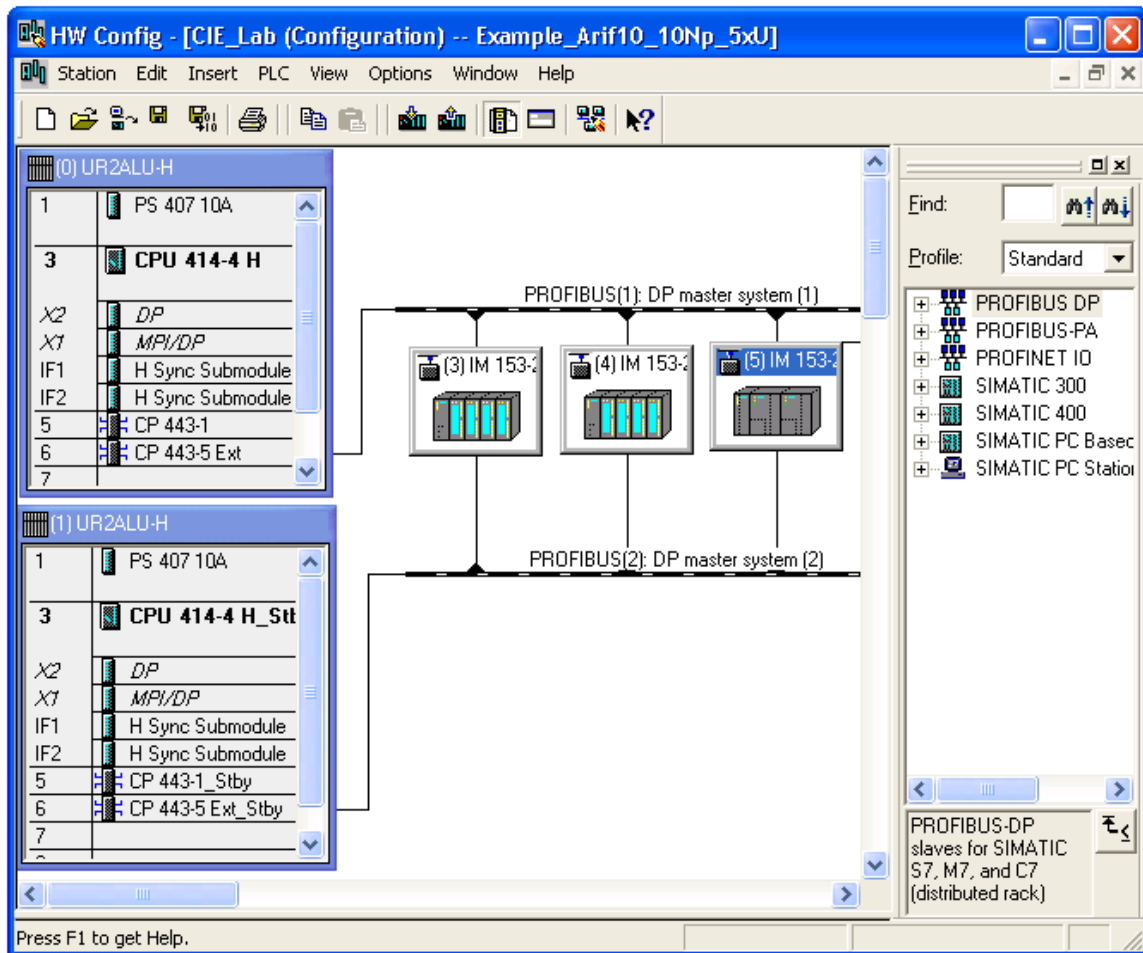


Fig C.7: Hardware Configuration for dual-rate MPC project

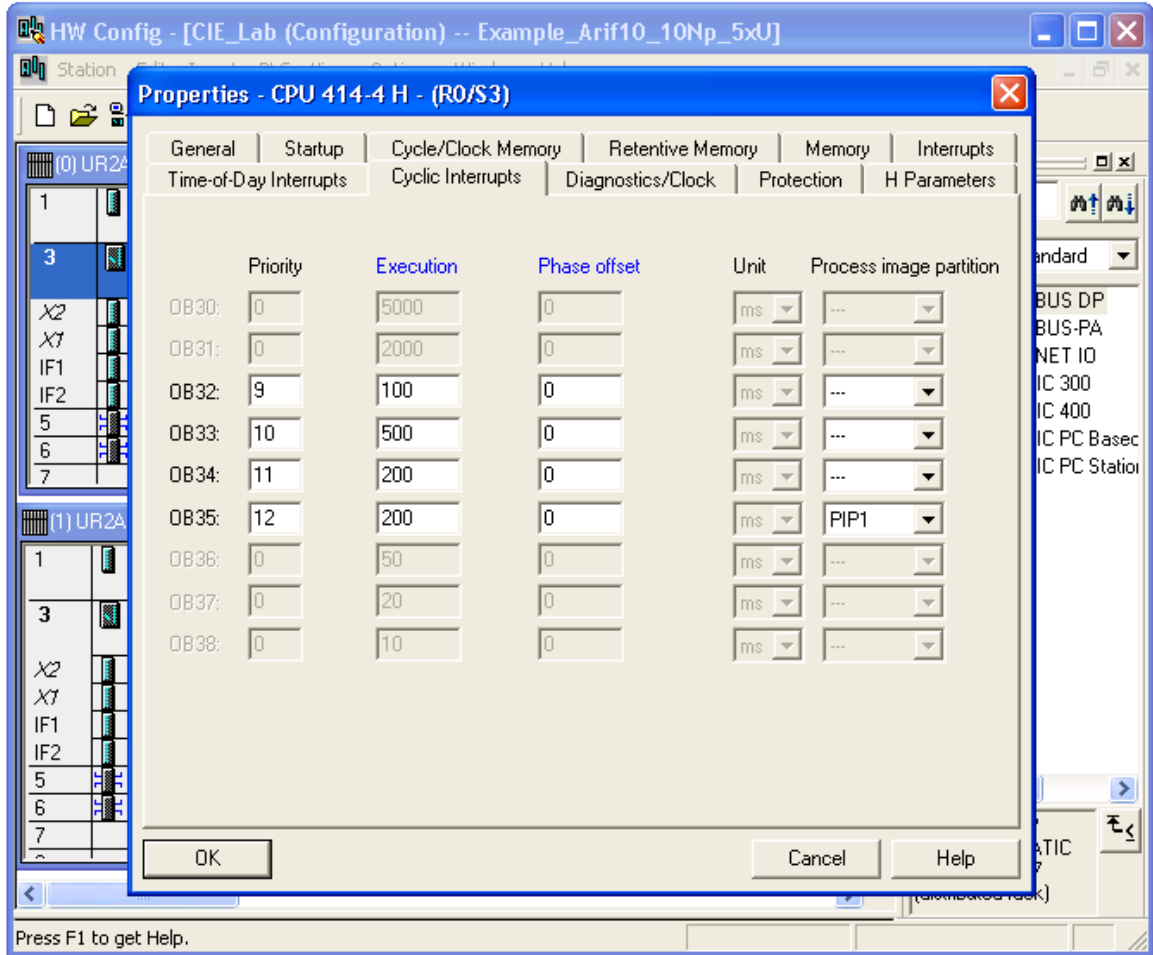


Fig C.8: Macrocycle property setting on CPU properties

### C.3 Running the Application

The following steps have to be completed prior to the execution of the dual-rate MPC application:

- 1) A new project created according to Fig C.1
- 2) Hardware configuration is done for the new project according to the Fig C.7
- 3) For maintaining proper 200ms macrocycle, the 'interrupt cycle' of the CPU properties tab is set to 200ms for OB35 block [Fig C.8]
- 4) Downloading the new hardware configuration to the DCS

- 5) Custom function blocks e.g. MPC\_DUAL and other functions and data blocks called by it are compiled in SCL compiler [Fig C.2(b)]
- 6) New custom function block MPC\_DUAL is imported to the CFC and connected them with AI and AO modules according to the design [Fig C.5]
- 7) Compiling the CFC chart containing all the components of dual-rate MPC project
- 8) Downloading the program to the DCS.

After all the above steps are complete error free, the program is RUN by clicking CFC's **CPU>Operating mode** item shown in Fig C.9 which brings the Operating mode window as Fig C.10.

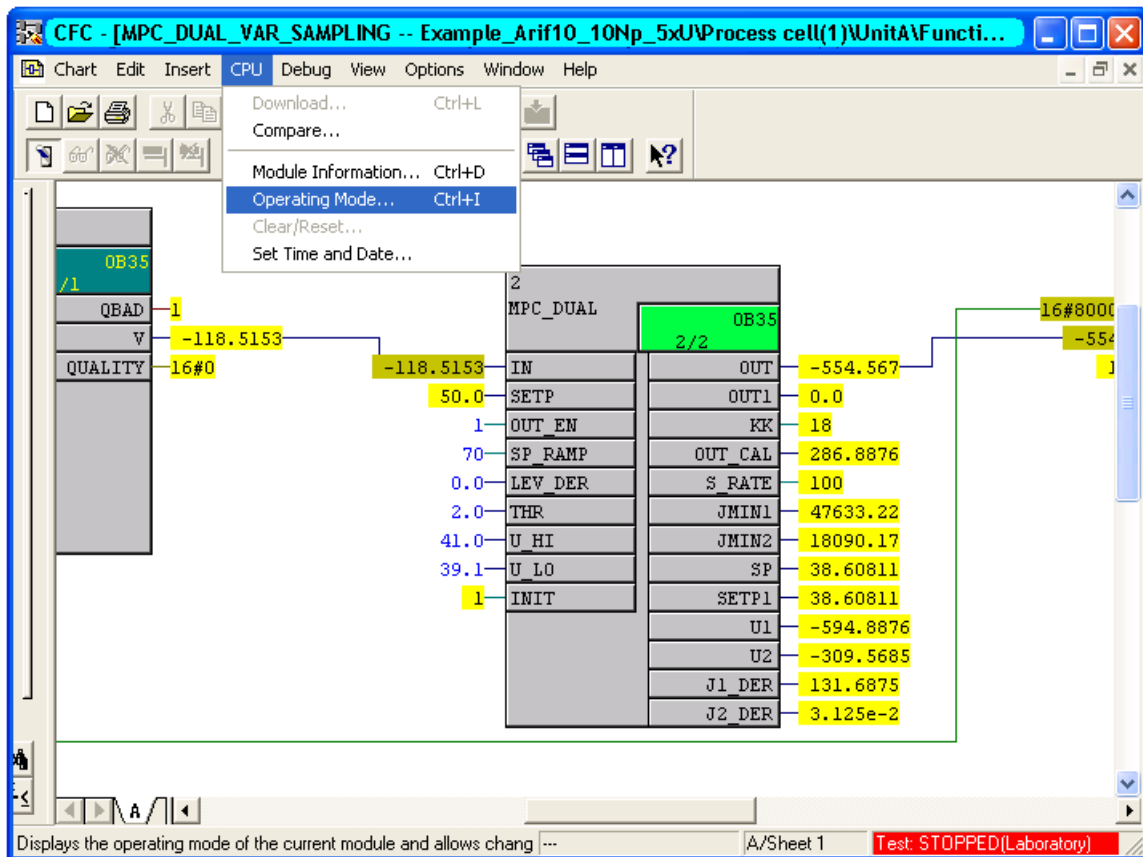


Fig C.9: Operating mode setting

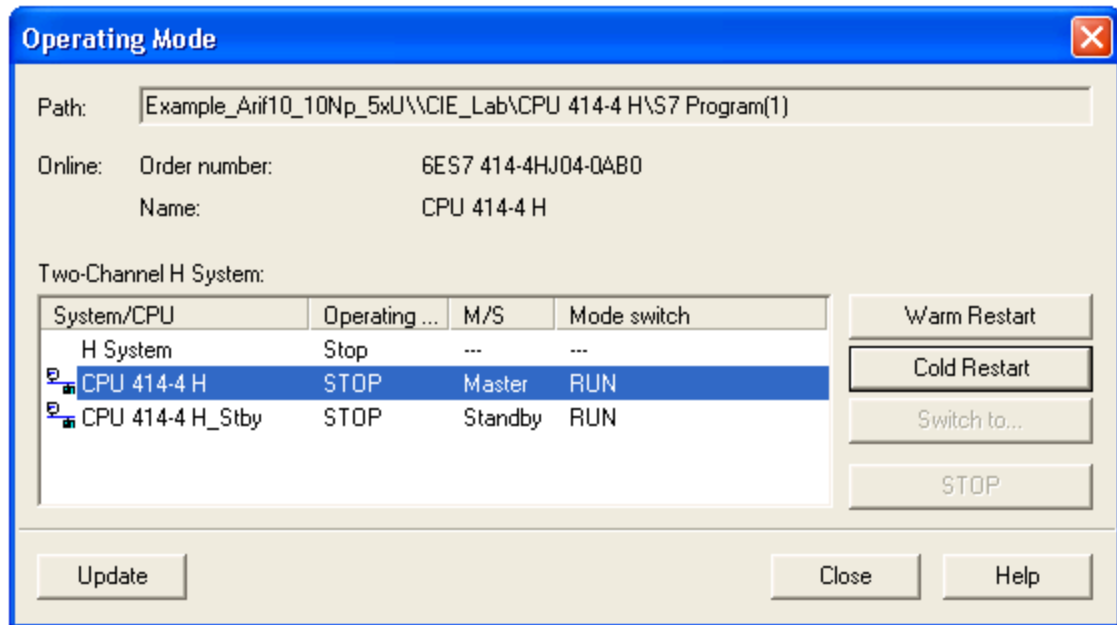


Fig C.10: Cold starting the dual-rate MPC program

The system then starts running and **Trend Display** can be turned on by **View>Trend Display** for viewing important variables online as shown in Fig C.11.

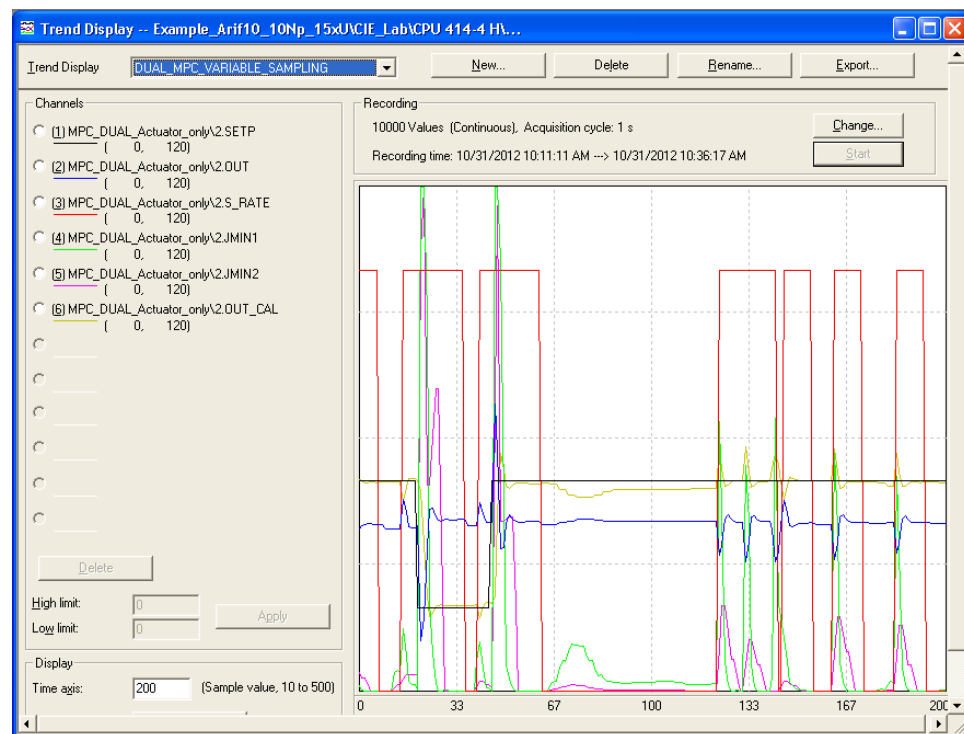


Fig C.11: Trend display for viewing various parameters online

## **APPENDIX D**

# **SIEMENS STRUCTURED CONTROL LANGUAGE (SCL) BASICS**

### **D.1 SLC Basics**

SCL (Structured Control Language) is a PASCAL oriented high-level-language for programming PLCs and DCS. SCL is also a part of PCS 7 and optimized for programming PCS 7 DCS applications. SCL fulfills the textual high-level language ST (Structured Text) defined in IEC 61131-3 and is prepared for certifying for the Reusability Level [48]. Features of SCL are as follows:

- Complex algorithm programming
- Programming of mathematical functions
- Data and recipe management
- Optimization of process

In this chapter, the discussion has been concentrated on the SCL features has mainly been used in the dual-rate MPC project.

### **D.2 SCL Functionality**

SCL supports the STEP 7 block concept is used in most Siemens PLC and DCS products. Following blocks are supported by SCL as shown in Fig D.1:

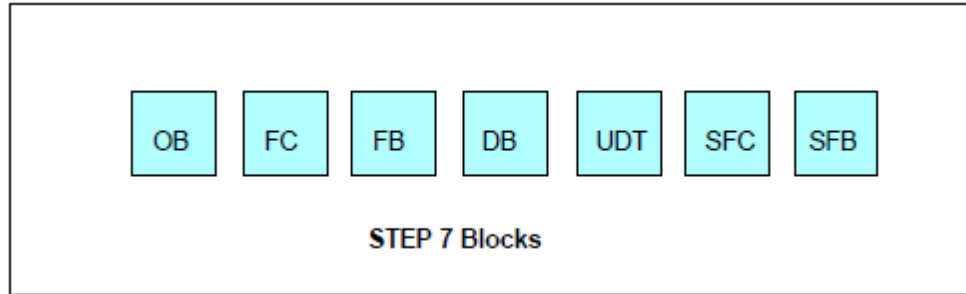


Fig D.1: Supported blocks

where, OB-Organization Block, FC- Function, FB-Function Block, DB-Data Block, UDT-User Defined Data Types, SFC-System Function, SFB-System Function Block.

### D.3 SCL Development Environment

SCL offers a powerful development environment for practical use which is tuned to both specific properties of SCL as well as to STEP 7 part of SIMATIC Manager. The following components are included in the development environment:

An editor for programing programs consists of different function blocks shown in Fig D.1. The programmer is supported in the process by powerful functions by means of library components or custom blocks designed by the programmer himself.

A batch compiler is used for editing program into MC7 machine code. The generated MC7 code executes on all CPUs of the S7-300/400 programmable controller which is based on CPU 314 (and CPU 414 having more advanced features).

A debugger is also a strong feature of the SCL software package for searching for logical program errors in an error-free compilation. Error searching is carried out in the source language. Figure D.2 provides an overview of the components of the SCL development environment.



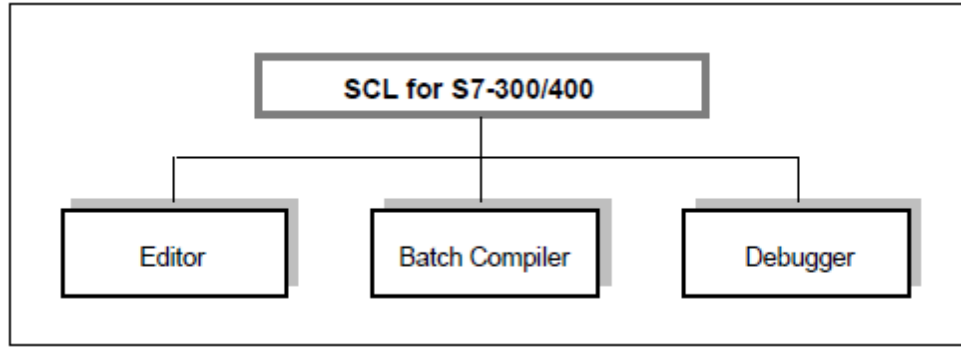


Fig D.2: Development environment of SCL

## D.4 Blocks in S7-SCL Source Files

Any number blocks can be created in SCL source file. Step 7 blocks & PCS 7 blocks are subunits of a user program which can be distinguished according to their function, structure and intended use in an application. There are different types of blocks that can be generated by SCL for example OB, DB, FC, FB and UDT.

Many predefined blocks can be found in the library of SIMATIC Step 7. Commonly used functions and control algorithms and communication blocks are already available in Step 7 standard package. Only special control algorithms and functions need to be developed.

## D.5 Calling Order of the Blocks in SCL Source File

In an SCL source file, for calling blocks some rules has to be followed as called block are located above the calling blocks. Figure D.3 shows the relative location of the blocks that should be followed in the SCL source file for error-free compilation.

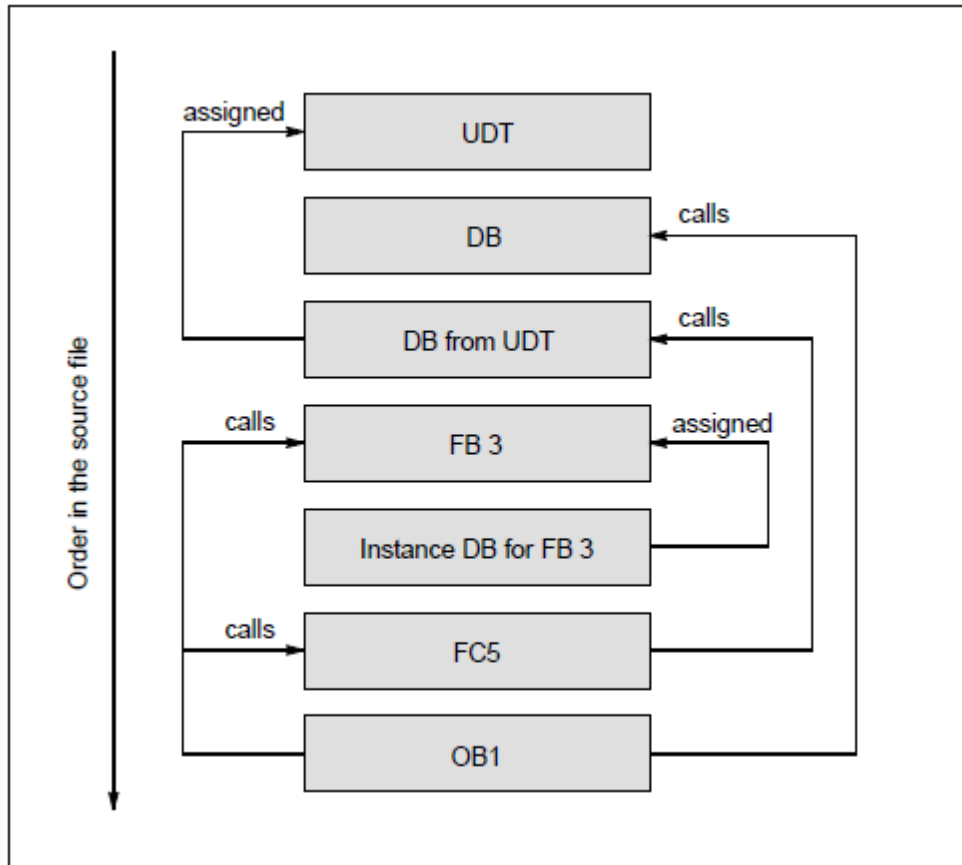


Fig D.3: Proper sequence of definition of blocks in SCL file

## D.6 General Structure of a Block

In SCL, a block can have the following items:

- A keyword identifies the start of a block and a block number or a symbolic block name, for example, "ORGANIZATION\_BLOCK OB1" for an organization block. For functions, the function type is also specified. This determines the data type of the return value. If you want no value returned, specify the keyword VOID.
- Optional block title is preceded by the keyword "TITLE =".
- Optional block comment. Each line beginning with "//", the block comment can be extended over several lines.

- Entry of the block attributes (optional)
- Entry of the system attributes for blocks (optional)
- Declaration section (depending on the block type)
- Statement section in logic blocks or assignment of actual values in data blocks (optional)
- In logic blocks: Statements
- Block end indicated by END\_ORGANIZATION\_BLOCK, END\_FUNCTION\_BLOCK or END\_FUNCTION

## D.7 Block Start and End Identifiers

The start and the end identifiers of a block vary depending on the block types. In the following Table D.1, the allowed syntax for various types of blocks is shown:

Table D.1: Start and end syntax for various blocks

Function block	FB	FUNCTION_BLOCK fb_name ... END_FUNCTION_BLOCK
Function	FC	FUNCTION fc_name : function type ... END_FUNCTION
Organization block	OB	ORGANIZATION_BLOCK ob_name ... END_ORGANIZATION_BLOCK
Data block	DB	DATA_BLOCK db_name ... END_DATA_BLOCK
Shared data type	UDT	TYPE udt_name ... END_TYPE

## D.8 Declarations of Local Variables, Parameters, Constants, and Labels

Table D.2 describes the syntax of declaration for different data types:

Table D.2: Syntax of declarations of different types of data

<b>Data</b>	<b>Syntax</b>	<b>FB</b>	<b>FC</b>	<b>OB</b>	<b>DB</b>	<b>UDT</b>
Constants	CONST declaration list END_CONST	X	X	X		
Labels	Labels LABEL declaration list END_LABEL	X	X	X		
Temporary variables	Temporary Variables VAR_TEMP declaration list END_VAR	X	X	X		
Static variables	Static variables VAR declaration list END_VAR	X	X		X	X
Input parameters	Input parameters VAR_INPUT declaration list END_VAR	X	X			
Output parameters	Output parameters VAR_OUTPUT declaration list END_VAR	X	X			
In/Out parameters	In/out parameters VAR_IN_OUT declaration list END_VAR	X	X			

## D.9 Data Types

### Elementary Data Types:

SCL supports following elementary data types for programing blocks as shown in Table D.3. Elementary data types are data elements that cannot be subdivided into smaller units. DIN EN 1131-3 standard defines these elementary data types.

Table D.3: SCL elementary data types

Group	Data Types	Explanation
Bit Data Types	BOOL BYTE WORD DWORD	Data elements of this type occupy either 1 bit, 8 bits, 16 bits or 32 bits
Character Types	CHAR	Data elements of this type occupy exactly 1 character in the ASCII character set
Numeric Types	INT DINT REAL	Data elements of this type are available for processing numeric values.
Time Types	TIME DATE TIME_OF_DAY S5TIME	Data elements of this type represent the various time and date values in STEP 7.

### Complex Data Types:

SCL supports a set of complex data types which is a very powerful feature of SCL supporting the development of complex algorithms and functions. Table D.4 shows the set of complex data types available in SCL.

Table D.4: SCL complex data types

Data Type	Explanation
DATE_AND_TIME DT	Defines an area of 64 bits (8 bytes). This data type stores date and time (as a binary coded decimal) and is a predefined data type in S7-SCL.
STRING	Defines an area for a character string for up to 254 characters (data type CHAR).
ARRAY	Defines an array consisting of elements of one data type (either elementary or complex).
STRUCT	Defines a group of data types in any combination of types. It can be an array of structures or a structure consisting of structures and arrays.

## D.10 Example SCL Code for a Function Block

An example data block and a function of the dual-rate MPC project are shown below and its different parts from the perspective of programming in a high level language are described below for easier understanding of SCL blocks and codes in a bit details:

```

1:    //COMMON DATA BLOCK FOR MATRIX FUNCTION BLOCKS
2:    DATA_BLOCK MAT
3:
4:    STRUCT
5:        A :ARRAY[1..10,1..10] OF REAL;
6:        B :ARRAY[1..10,1..10] OF REAL;
7:        C :ARRAY[1..10,1..10] OF REAL;
8:    END_STRUCT
9:    BEGIN

10:   END_DATA_BLOCK
11:

```

```

12:  //MATRIX MULTIPLICATION FUNCTION BLOCK
13:  FUNCTION_BLOCK MAT_MUL
14:
15:  VAR_INPUT
16:      ROW1  : INT;
17:      COL1  : INT;
18:      COL2  : INT;
19:  END_VAR
20:
21:  VAR_TEMP
22:      I  : INT;
23:      J  : INT;
24:      K  : INT;
25:  END_VAR
26:
27:  BEGIN
28:      FOR I:=1 TO ROW1 BY 1 DO
29:          FOR J:=1 TO COL2 BY 1 DO
30:              MAT.C[I,J]:=0.0;
31:              FOR K:=1 TO COL1 BY 1 DO
32:                  MAT.C[I,J]:=MAT.C[I,J]+MAT.A[I,K]*MAT.B[K,J];
33:              END_FOR;
34:          END_FOR;
35:      END_FOR;
36:  END_FOR;
37:
38:  END_FUNCTION_BLOCK

```

Table D.5: Example SCL code explanation

Line Number	Explanation
1.	Commenting starts by using ‘//’ which describes the data block’s features which is declared below
2.	Starting of a data block named MAT
4-7.	Declaring a structure containing three 2 dimensional array (10x10) variables named A,B and C
8.	Ending the declaration of the structure
9.	Beginning of data block code section. In data block there is no code to be included
10.	Ending the data block named MAT
12.	Comment for describing the function declared below
13.	Starting a function block named MAT_MUL
15-19.	Declaration of three INT type INPUT variables named as ROW1, COL1 and COL2. INPUT variables have visibility outside the block and can be modified externally while the block is running
21-25.	Declaration of three INT type Temporary variables which do not have any visibility from outside of the function block.
28.	Marks the beginning of the function block coding
29-36.	Three nested loops for calculating the matrix multiplication of MAT.A and MAT.B element and puts the result in MAT.C element of the MAT data block
38.	Marks the ending of the function block



## VITAE

Name	Mohammad Arif Hossain
Place of Birth	Dhaka, Bangladesh
Year of Birth	1978
Post Secondary	Bangladesh University of Engineering & Technology
Education and	Dhaka, Bangladesh
Degrees	1996-2001, B.Sc. (Mechanical Engineering)
	The University of Western Ontario
	London, ON, Canada
	2011-2013, M.E.Sc. (Electrical & Computer Engineering)

### Publications:

Mohammad Arif Hossain, Jin Jiang and Drew J. Rankin, "Design and Implementation of a Dual-rate MPC on a Fieldbus based Distributed Control System," *IFAC Journal for Control Engineering Practice*, under preparation.