VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV VÝKONOVÉ ELEKTROTECHNIKY A ELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF POWER ELECTRICAL AND ELECTRONIC ENGINEERING

DATA ACQUISITION AND CONTROL SYSTEM OF HYDROELECTRIC POWER PLANT USING INTERNET TECHNIQUES

SYSTÉM SNÍMÁNÍ DAT A OVLÁDÁNÍ VODNÍ ELEKTRÁRNY PROSTŘEDNICTVÍM INTERNETOVÉ TECHNIKY

DIZERTAČNÍ PRÁCE
DOCTORAL THESIS

AUTOR PRÁCE — Ing. MOUSA SATTOUF
AUTHOR

VEDOUCÍ PRÁCE — Prof. Ing. JIŘÍ SKALICKÝ, CSc.
SUPERVISOR

BRNO 2015

ABSTRACT

Hydropower has now become the best source of electricity on earth. It is produced due to the energy provided by moving or falling water. History proves that the cost of this electricity remains constant over the year. Because of the many advantages, most of the countries now have hydropower as the source of major electricity producer. The most important advantage of hydropower is that it is green energy, which mean that no air or water pollutants are produced, also no greenhouse gases like carbon dioxide are produced which makes this source of energy environment-friendly. It prevents us from the danger of global warming.

Using internet techniques to control several hydroelectric plants has very important advantages, as reducing operating costs and the flexibility of meeting changes of energy demand occurred in consumption side. Also it is very effective to confront large disturbances of electrical grid, such as adding or removing large loads, and faults. In the other hand, data acquisition systems provides very useful information for both typical and scientific analysis, such as economical costs reducing, fault prediction systems, demand prediction, maintenance schedules, decision support systems and many other benefits.

This thesis describes a generalized model which can be used to simulate a data acquisition and control system of hydroelectric power plant using MATLAB/SIMULINK and TrueTime simulink library. The plant considered consists of hydropower turbine connected to synchronous generator with excitation system, and the generator is connected to public grid.

Simulation of hydropower turbine and synchronous generator can be done using various simulation tools, In this work, SIMULINK/MATLAB is favored over other tools in modeling the dynamics of a hydropower turbine and synchronous machine. The SIMULINK program in MATLAB is used to obtain a schematic model of the hydropower plant by means of basic function blocks. This approach is pedagogically better than using a compilation of program code as in other software programs .The library of SIMULINK software programs includes function blocks which can be linked and edited to model. Either Simulink Real-Time library or TrueTime library can be used to build and simulate internet and real time systems, in this thesis the TrueTime library was used.

KEYWORDS

Data acquisition systems, Networked control systems, Hydroelectric plant, Synchronous generator, Excitation system, Three-phase fault,PID controller, Hydraulic turbine, Power system stabilizer, Internet, Switched Ethernet, Analog to digital converters, Digital to analog converters, Discrete time control systems, Transducers, Steady state model, Transient model, Automatic voltage regulator, governors, Matlab, Simulink, TrueTime.

ABSTRAKT

Vodní energie se nyní stala nejlepším zdrojem elektrické energie na zemi. Vyrábí se pomocí energie poskytované pohybem nebo pádem vody. Historie dokazuje, že náklady na tuto elektrickou energii zůstávají konstantní v průběhu celého roku. Vzhledem k mnoha výhodám, většina zemí nyní využívá vodní energie jako hlavní zdroj pro výrobu elektrické energie.Nejdůležitější výhodou je, že vodní energie je zelená energie, což znamená, že žádné vzdušné nebo vodní znečišťující látky nejsou vyráběny, také žádné skleníkové plyny jako oxid uhličitý nejsou vyráběny, což činí tento zdroj energie šetrný k životnímu prostředí. A tak brání nebezpečí globálního oteplování.

Použití internetové techniky k ovladání několika vodních elektráren má velmi významné výhody, jako snížení provozních nákladů a flexibilitu uspokojení změny poptávky po energii na straně spotřeby. Také velmi efektivně čelí velkým narušením elektrické sítě, jako je například přidání nebo odebrání velké zátěže, a poruch. Na druhou stranu, systém získávání dat poskytuje velmi užitečné informace pro typické i vědecké analýzy, jako jsou ekonomické náklady, predikce poruchy systémů, predikce poptávky, plány údržby, systémů pro podporu rozhodování a mnoho dalších výhod.

Tato práce popisuje všeobecný model, který může být použit k simulaci pro sběr dat a kontrolní systémy pro vodní elektrárny v prostředí Matlab / Simulink a TrueTime Simulink knihovnu. Uvažovaná elektrárna sestává z vodní turbíny připojené k synchronnímu generátoru s budicí soustavou, generátor je připojen k veřejné elektrické síti.

Simulací vodní turbíny a synchronního generátoru lze provést pomocí různých simulačních nástrojů. V této práci je upřednostňován SIMULINK / MATLAB před jinými nástroji k modelování dynamik vodní turbíny a synchronního stroje. Program s prostředím MATLAB SIMULINK využívá k řešení schematický model vodní elektrárny sestavený ze základních funkčních bloků. Tento přístup je pedagogicky lepší než komplikované kódy jiných softwarových programů. Knihovna programu Simulink obsahuje funkční bloky, které mohou být spojovány, upravovány a modelovány. K vytvoření a simulování internetových a Real Time systémů je možné použít bud‘ knihovnu simulinku Real-Time nebo TRUETIME, v práci byla použita knihovna TRUETIME.

KLÍČOVÁ SLOVA

Systémy snímání dat, Systémy síťové ovládání, Vodní elektrárna, Synchronní generátor, Excitační systém, Třífázové poruchy, PID regulátor, Vodní turbiny, stabilizátor elektrické soustavy, Systémový stabilizátor, Internet, Switched Ethernet, Analogově digitální převodník, Digitálně analogový převodník, Ovládání diskrétní soustavy, Převodníky, model v rovnovážném stavu, Přechodový model, Automatická regulace napětí, řídicí systém, Matlab, Simulink, TrueTime.

SATTOUF, M. *Data Acquisition and Control System of Hydroelectric Power Plant Using Internet Techniques*. Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, 2015. 119 p. Supervised by Prof. Ing. Jiří Skalický, CSc.

DECLARATION

I declare that I have elaborated my doctoral thesis on the theme of "Data Acquisition and Control System of Hydroelectric Power Plant Using Internet Techniques" independently, under the supervision of the doctoral thesis supervisor and with the use of technical literature and other sources of information which are all quoted in the thesis and detailed in the list of literature at the end of the thesis.

Brno ....

Mousa SATTOUF

PROHLÁŠENÍ

Prohlašuji, že dizertační práci na téma "Data Acquisition and Control System of Hydroelectric Power Plant Using Internet Techniques" vypracoval samostatně pod vedením školitele a s využitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury.

Jako autor uvedené dizertační práce dále prohlašuji, že v souvislosti s vytvořením této dizertační práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V brně dne..... .....

Mousa SATTOUF

ACKNOWLEGMENT

I would like to express my deepest appreciation and sincere gratitude to my supervisor prof. Ing. Jiří Skalický, CSc for his continuous support, encouragement, and valuable advice throughout the progress of study. I would like also to thank all staff of Department of Power Electrical and Electronic Engineering.

I would also like to express my sincerest gratitude to my parents, Saleh and Wardah, my wife, Dania, and my sisters, Sahar, Rasha and Amani for their infinite patience, providing continuous guidance and encouragement throughout my study.

I am also very grateful for my brother Nezar and I hope that his soul is resting peacefully.

Mousa Sattouf

Brno, February 2015.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ACRONYMS

Acronyms

DAQ	Data acquisition
ADC	Analog to digital converter
DAC	Digitel to analog converter
LVDT	Linear voltage differential transformer
MEMS	Microelectromechanical system
MSPS	Million samples per second
ECL	Emitter-coupled logic
TTL	Transistor–transistor logic
RES	Renewable energy systems
RF	Radio frequency
RoMain	Railway Open Maintenance tool
TCP	Transmission control protocol
HTTP	Hypertext transfer protocol
PPP	Point-to-Point Protocol
HTML	Hypertext markup language
XML	Extensible markup language
NCS	Netwoked control system
OSI	Open-systems interconnection
RA	Random access network
CD	Collision detection network
CA	Collision avoidance network
TDM	Time-division multiplexed network
TDM/TP	Token passing TDM
TDM/MS	Master slave TDM
MAC	Medium access control
CSMA	Carrier sense multiple access
BEB	Binary exponential backoff algorithm
CRC	Cyclic redundancy check
FIFO	first-in-first-out
S/H	Sample and hold
PID	Proportional-integral-derivative controller
AVR	Automatic voltage regulator
PSS	Power system stabilizer

Symbols

t	Time
T	samplimg time
s	Laplace factor
z	Z-transform function variable

v_a, v_b, v_c	Phases a, b and c stator voltages of synchronous machine
i_a, i_b, i_c	Phases a, b and c stator currents of synchronous machine
v_f	d-axis field winding voltage
i_f	d-axis field winding current
r_f	d-axis field winding resistance
λ_f	d-axis field winding flux linkage
v_g	q-axis field winding voltage
i_g	q-axis field winding current
r_g	q-axis field winding resistance
λ_g	q-axis field winding flux linkage
v_{kd}	d-axis damper winding voltage
i_{kd}	d-axis damper winding current
r_{kd}	d-axis damper winding resistance
λ_{kd}	d-axis damper winding flux linkage
v_{kq}	q-axis damper winding voltage
i_{kq}	q-axis damper winding current
r_{kq}	q-axis damper winding resistance
λ_{kq}	q-axis damper winding flux linkage
θ_r	Electrical angle
i_q	q-axis stator current
i_d	d- axis stator current
v_q	q-axis stator voltage
v_d	d- axis stator voltage
λ_q	q-axis stator windings flux linkage
λ_d	d- axis stator windings flux linkage
ψ_q	q-axis stator windings flux linkage per second
ψ_d	d- axis stator windings flux linkage per second
ω_r	Rotor angular speed rad/s
ω_e	Electrical angular speed rad/s
ω_b	Base angular speed
ω_{rm}	Rotor angular speed in mechanical radians per second
ω_{bm}	Base mechanical angular speed
ψ_{mq}	q-axis mutual flux linkage per second
ψ_{md}	d-axis mutual flux linkage per second
L_{mq}	q-axis mutual inductance
L_{md}	d-axis mutual inductance
x_{ls}	Stator winding per phase reactance
r_s	Stator winding per phase resistance
ψ'_{kd}	d-axis damper winding flux linkage referred to stator
i'_{kd}	d-axis damper winding current referred to stator
r'_{kd}	d-axis damper winding resistance referred to stator

x'_{kd}	d-axis damper winding reactance referred to stator
ψ'_{kq}	q-axis damper winding flux linkage referred to stator
i'_{kq}	q-axis damper winding current referred to stator
r'_{kq}	q-axis damper winding resistance referred to stator
x'_{kq}	q-axis damper winding reactance referred to stator
ψ'_{f}	d-axis exciting winding flux linkage referred to stator
i'_{f}	d-axis exciting winding current referred to stator
r'_{f}	d-axis exciting winding resistance referred to stator
x'_{f}	d-axis exciting winding reactance referred to stator
i_q^s	Stationary q-axis stator current
i_d^s	Stationary d-axis stator current
p	Poles number
P_{em}	Electromechanical power
T_{em}	Electromechanical torque
T_{mech}	Externally-applied mechanical torque
T_{damp}	Friction torque
J	Moment of inertia
H	Inertia constant
V_b	Base voltage
I_b	Base current
Z_b	Base impedance
T_b	Base torque
x'_{d}	d-axis transient reactance
x'_{q}	q-axis transient reactance
E'_{d}	d-axis transient electromotive force
E'_{q}	q-axis transient electromotive force
T'_{do}	d-axis open circuit transient time constant
T'_{qo}	q-axis open circuit transient time constant
L'_{d}	d-axis transient stator windings inductance
L'_{q}	q-axis transient stator windings inductance
λ'_{d}	d-axis transient stator windings flux linkage
λ'_{q}	q-axis transient stator windings flux linkage
λ'_{kd}	d-axis transient damper windings flux linkage
λ'_{kq}	q-axis transient damper windings flux linkage
E''_{d}	d-axis sub-transient electromotive force
E''_{q}	q-axis sub-transient electromotive force
T''_{do}	d-axis open circuit sub-transient time constant
T''_{qo}	q-axis open circuit sub-transient time constant
L''_{d}	d-axis sub-transient stator windings inductance
L''_{q}	q-axis sub-transient stator windings inductance
λ''_{d}	d-axis sub-transient stator windings flux linkage

λ''_q	q-axis sub-transient stator windings flux linkage
F_{net}	Net force on the water in penstock
Q	Volumetric flow rate
ρ	Mass density of water
H_s	Static head
H	Head
H_l	Head loss
A	Penstock cross-sectional area
g	Acceleration of gravity
h_{base}	Base head
q_{base}	Base Volumetric flow rate
q	Volumetric flow rate per unit
h	Head per unit
T_w	Water starting time
G	Gate position
D	Damping cofficient
P_m	Turbine output power
A_t	Factor introduced by difference in bases values
q_o	Initial volumetric flow rate value
G_o	Initial gate position value
Z_C	Compensation impedance of excitation system
V_C	Compensation voltage
I_g	Generator current
V_g	Generator voltage
i_{exf}	Exciter field current
R_{exf}	Exciter field resistance
L_{exf}	Exciter field inductance
K_F	Gain of feedback first order filter of excitation system
T_F	Time constant of feedback first order filter of excitation system
K_A	AVR gain
T_A	AVR time constant
K_E	Exciter gain
T_E	Exciter time constant
A_{ex}, B_{ex}	Parameters of saturation exponential function
T_w	High pass filter time constant of PSS
T_1, T_2	Phase compensation time constants of PSS
K_{STAB}	Gain of PSS

1 INTRODUCTION

The widespread application of Renewable Energy Sources (RES) requires the use of internet techniques both for monitoring system operation and control of its operation. In this paper, the development of a data acquisition system for remote monitoring and control of hydroelectric power plant plants is presented. It is based on switched Ethernet network.

The advantages of networking in control systems are so strong that any systems above some minimal level of complexity are likely to utilize networking. There are many motivations for wanting computer networks in a control system, additional computing power and distribution of computing to match the target system‘s topology. There are also the motivation of data and information integrity and reduced cabling. Traditional control systems relied on analog information transmission, both for sensing and for actuation. Analog cables carry one signal per cable. Analog signals are also susceptible to contamination from a variety of noise sources. Operationally, digital signals can have arbitrarily strong protection against noise. Furthermore, the cost of that protection scales reasonably with the severity of the noise. Overall system cost is another major motivation for networked control systems.

Fig. 1.1 shows a block diagram of Data Acquisition (DAQ) and control system of any actual system or plant. Data acquisition stores data in database server or database node in format that can be easily retrievable engineering, scientific review, costs and economical analysis, fualt analysis, maintenance analysis and others. The system illustrated is the presented and simulated system in this thesis for hydroelectric power plant, which is consists of synchronous generetor, excitation system and hydrualic turbine.

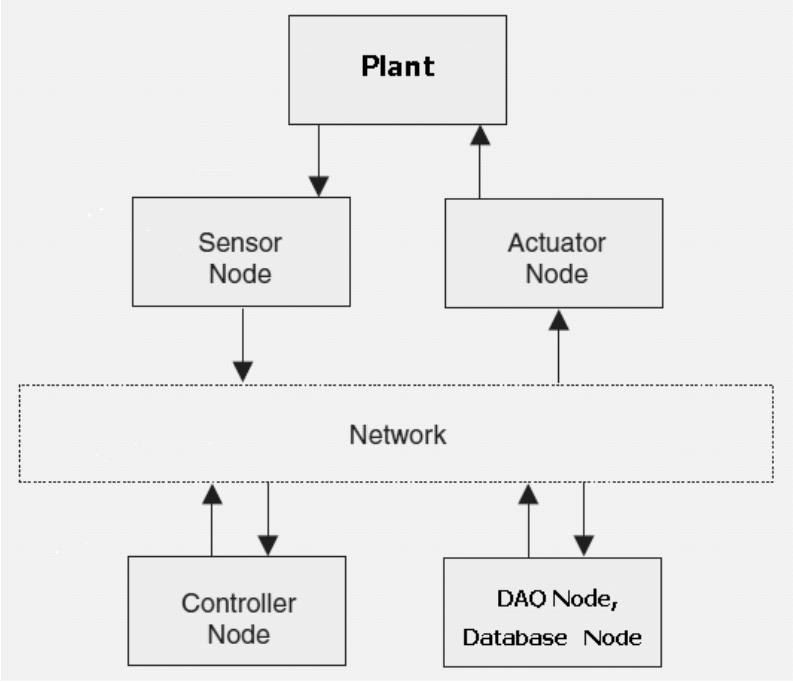


Fig. 1.1: Network control and DAQ system

1.2 Thesis layout

This thesis is organized into eight chapters. The layout of the thesis is described below.

- **Chapter 1 Introduction**: is an introduction about Data acquisition and control networked systems.
- **Chapter 2 Data Acquisition Systems**: describes data acquisition systems and its components which are transducers, signal conditioning and signal convensions. Finally it describes DAQ systems using internet.
- **Chapter 3 Networked Control Systems**: includes an introduction to NCS and its protocols, then describes Discrete-time control systems.
- **Chapter 4 Synchronous Machine Model**: describes three mathematical models of synchronoous generetors, which are steady state model, transient model and sub-transient model.
- **Chapter 5 Hydrualic Turbine:** describe hydrualic turbine systems, models, and thier governing systems.
- **Chapter 6 Excitation Systems**: describes exciters systems, Automatic Voltage Regulators AVR and Power System Stabilizers PSS.
- **Chapter 7 Internet Network Model/TrueTime**: describes the TrueTime simulink library and its components used to simulate the network.
- **Chapter 8 Simulation and Results**: presents four simulation projects for both steady state and transient models of synchronous generator.
- **Chapter 9 Conclusion**: contians conclusion and future work.

There are also three appendices to the thesis, they are:

- **Appendix A**: contains sets of parametrs values used in simulation.
- **Appendix B**: describes discrete PID method used in simulation.
- **Appendix C**: contains sets of TrueTime commands.

2 DATA ACQUISITION SYSTEMS

DAQ (Data AcQuisition) is simply the process of bringing a real-world signal, such as avoltage, into the computer, for processing, analysis, storage or other data manipulation. A Physical phenomena represent the real-world signal which is measured. In order to optimize the characteristics of a system in terms of performance, handling capacity and cost, the relevant subsystem can be combined together.

Most real-world data are not in a form that can be directly recorded by a computer. These quantities typically include temperature, pressure, distance, velocity, mass, and energy output (such as optical, acoustic, and electrical energy). Very often these quantities are measured versus time or position. A physical quantity must first be converted to an electrical quantity (voltage, current, or resistance) using a sensor or transducer. This enables it to be conditioned by electronic instrumentation, which operates on analog signals or waveforms (a signal or waveform is an electrical parameter, most often a voltage, which varies with time). This analog signal is continuous and monotonic, that is, its values can vary over a specified range (for example, somewhere between -5.0 volts and +3.2 volts) and they can change an arbitrarily small amount within an arbitrarily small time interval.

Analog data is generally acquired and transformed into the digital form for the purpose of processing, transmission and display. Rapid advances in Personal Computer (PC) hardware and software technologies have resulted in easy and efficient adoption of PCs in various precise measurement and complex control applications. A PC based measurement or control application requires conversion of real world analog signal into digital format and transfer of digitized data into the PC.

The process of converting an analog signal to a digital one is called analog-to-digital conversion, and the device that performs this is an analog-to-digital converter (ADC). The resulting digital signal is usually an array of digital values of known range (scale factor) separated by a fixed time interval (or sampling interval). If the values are sampled at irregular time intervals, the acquired data will contain both value and time information. The reverse process of converting digital data to an analog signal is called digital-to-analog conversion, and the device that does this is called a digital-to-analog converter (DAC). Some common applications for DACs include control systems, waveform generation and speech synthesis.

A general-purpose laboratory data acquisition system typically consists of ADCs, DACs, and digital inputs and outputs. Fig. 2.1 shows a block diagram of DAQ system.

The additional channels are often added to an ADC via a multiplexer, used to select which one of the several analog input signals to convert at any given time. This is an economical approach when all the analog signals do not need to be simultaneously monitored.

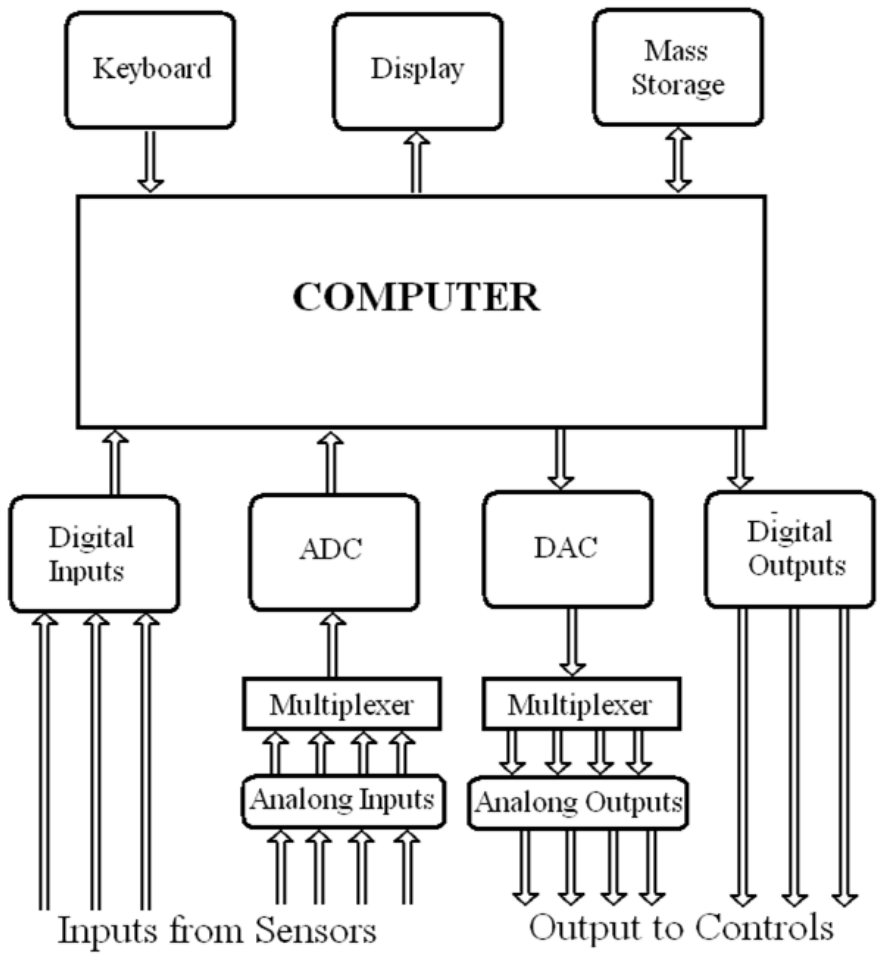


Fig.2.1: Block diagram of DAQ system

2.1 Transducers

Most real-world events and their measurements are analog. That is, the measurements can take on a wide, nearly continuous range of values. The physical quantities of interest can be as diverse as heat, pressure, light, force, velocity, or position. To be measured using an electronic data acquisition system, these quantities must first be converted to electrical quantities such as voltage, current, or impedance.

A transducer converts one physical quantity into electrical one, to be used with electronic instrumentation. The mathematical description of what a transducer does is its transfer function, H. So the operation of a transducer can be described as:

$$Outpot = H * Input \tag{2.1}$$

Since the transducer is the "front end" of the data acquisition system, its properties are critical to the overall system performance. Some of these properties are sensitivity, stability, noise, dynamic range, and linearity. Very often the transfer function is dependent on the input quantity. It may be a linear function for one range of input values and then become nonlinear for another range. Looking at sensitivity and noise, if the transducer's sensitivity is too low, or its noise level too high, signal conditioning may not produce an adequate signal-to-noise ratio.

Common transducers types are:

- **Temperature Sensors**: They have electrical parameters that vary with temperature, like Thermocouples, Thermistors, Resistance Temperature Detectors (RTDs) and Monolithic Temperature Transduces.
- **Optical Sensors**: They are used for detecting light intensity. Typically, they respond only to particular wavelengths or spectral bands. One sensor may respond only to visible light in the blue-green region, while another sensor may have a peak sensitivity to near-infrared radiation. Common optical transducers types are Vacuum Tube Photosensors, Photoconductive Cells, Photovoltaic Cells, Semiconductor Light Sensors and Thermoelectric Optical Sensors.
- **Force and Pressure Transducers**: They are used for measuring force and pressure. Most of them rely on the movement of a diaphragm mounted across a pressure differential. Common types are strain Gages and Piezoelectric Transducers.
- **Magnetic Field Sensors**: This group of transducers is used to measure either varying or fixed magnetic fields.
- **Position Sensors**: They are used to measure mechanical displacement or the position of an object. Common types are Potentiometers, Capacitive and Inductive Sensors, linear voltage differential transformers (LVDTs) and Optical Encoders.
- **Fluid Flow Sensors**: A wide range of transducers and techniques are commonly used to measure fluid flow rates, such as Head Meters, Rotational Flowmeters and Ultrasonic Flowmeters.

Other new technologies are gaining importance in commercial sensors. These include microelectromechanical systems (MEMS) and smart sensors. MEMS are small electromechanical devices fabricated using semiconductor integrated-circuit processing techniques. Smart sensors can range from a traditional transducer that simply contains its own signal conditioning circuitry to a device that can calibrate itself, acquire data, analyze it, and transmit the results over a network to a remote computer. An emerging class of smart sensors is defined by the family of IEEE 1451 standards, which are designed to simplify the task of establishing communications between transducers and networks.

2.1 Signal conditioning

Nearly all transducer signals must be conditioned by analog circuitry before they can be digitized and used by a computer. This conditioning often includes amplification and filtering, although more complex operations can also be performed on the waveforms.

Amplification (or occasionally attenuation) is necessary for the signal's amplitude to fit within a reasonable portion of the ADC's dynamic range. Also filtering must usually be performed on analog signals for several reasons. Sometimes noise or unwanted signal artifacts can be eliminated by filtering out certain portions of the signal's spectra. A low-frequency drift on a signal without useful DC information can be removed using a high-pass filter. Most often, low-pass filters are employed to limit the high end of a waveform's frequency response just prior to digitization, to prevent aliasing problems. Additional analog signal processing functions include modulation, demodulation, and other nonlinear operations.

The simplest conditioning analog circuit elements are passive components: resistors, capacitors, and inductors. They can be used as attenuators and filters. For example, a simple RC circuit can be used as a high-pass or low-pass filter. Discrete semiconductor devices, such as diodes and transistors, are commonly used in analog signal-conditioning circuits. Diodes are useful, among other things, as rectifiers/detectors, switches, clamps, and mixers.

Transistors are often used as amplifiers, switches, oscillators, phase shifters, filters, and many other applications. The most common analog circuit semiconductor component is the operational amplifier, called the op amp. This circuit element is usually a monolithic device (an integrated circuit), although hybrid modules, based on discrete transistors, are still used in special applications. The op amp is used in both linear and nonlinear applications involving amplification and signal conditioning.

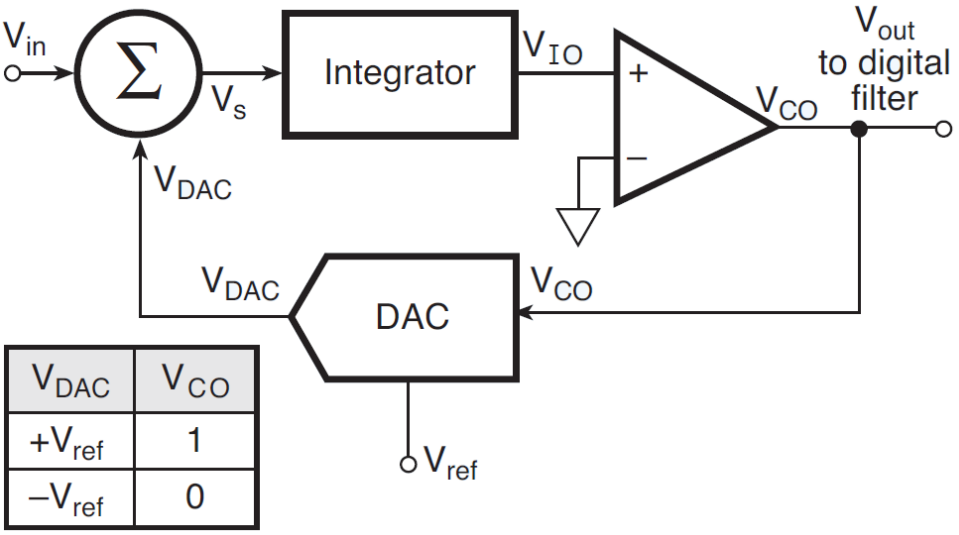
2.2 Analong / digital conversion

Analog-to-Digital Converters (ADCs) transform an analog voltage to a binary number (a series of 1's and 0's), and then eventually to a digital number (base 10) for reading on a meter, monitor, or chart. The number of binary digits (bits) that represents the digital number determines the ADC resolution. However, the digital number is only an approximation of the true value of the analog voltage at a particular instant because the voltage can only be represented (digitally) in discrete steps. How closely the digital number approximates the analog value also depends on the ADC resolution.

A mathematical relationship conveniently shows how the number of bits an ADC handles determines its specific theoretical resolution: An n-bit ADC has a resolution of one part in 2^n, For example, a 12-bit ADC has a resolution of one part in 4,096, where $2^{12} = 4{,}096$. Thus, a 12-bit ADC with a maximum input of 10 Vdc can resolve the measurement into $\frac{10}{4096}\text{Vdc} = 0.00244\ \text{Vdc} = 2.44\text{mV}$.

Some of the more common ADCs are Ramp ADC, Successive-Approximation ADC, Dual-Slope ADC, Voltage-to-Frequency Converter, Flash ADC and Sigma-Delta Converter.

Fig. 2.2 shows a block diagram of Sigma-Delta Converter, which are often used in digital multimeters, panel meters, and data acquisition boards. they are relatively inexpensive primarily because they have a single-bit DAC, but they can obtain high-resolution measurements using oversampling techniques.



V_{DAC}	V_{CO}
$+V_{ref}$	1
$-V_{ref}$	0

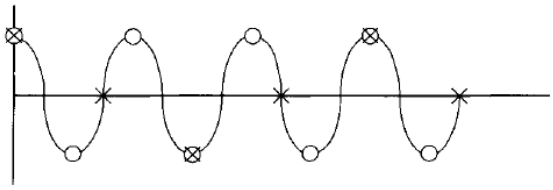
Fig. 2.2: Sigma-Delta ADC.

Many current ADC ICs use variations on the techniques we have previously examined, along with additional features such as input multiplexers, sampleand-hold amplifiers, and programmable gain amplifiers.

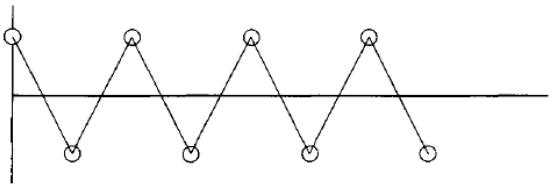
One important variant is the serial ADC, for medium-speed ADCs, up to about 1 million samples per second (MSPS), many IC manufacturers produce devices with serial outputs. For these converters, there is a single data line that is time-multiplexed: each bit of the output digital word is present in sequential order, for a fixed amount of time, usually one clock cycle. There are also ADCs designed for low-power applications, such as battery-powered accessories. These ICs can operate from low power supply voltages.

In recent years, both the speed and resolution of ADCs have increased. Commercial ADC ICs are available up to 1000 MSPS (for eight-bit resolution), this speed is one sample every nanosecond. Even at higher resolutions, ADC speeds have increased significantly. Currently, there are 10-bit ADC ICs as fast as 100 MSPS, 12-bit ADCs over 50 MSPS, 14-bit devices up to 10 MSPS, and 16-bit converters up to 5 MSPS. Very High Speed Flash ADCs and Pipelined ADCs.

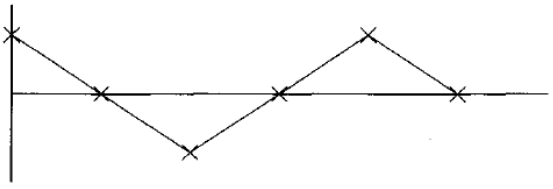
The most important ADC parameters are resolution and sampling rate. In practice, an ADC's sampling rate should be much higher than twice the maximum signal frequency. A value of five times is a good choice. In most data acquisition systems, the analog input is filtered to eliminate any signal components above the Nyquist frequency. This is often referred to as an anti-aliasing filter. For such a low-pass filter to produce adequate attenuation at the Nyquist frequency, it should have a cut off frequency well below that point, requiring a sampling rate many times higher than the maximum frequency of interest. Fig. 2.3 illustrates the effect of use low sampling frequency, less than twice of the signal frequency.



(a) Sine Wave of Frequency f_0 Sampled at 2 f_0 and 4/3 f_0



(b) Waveform Reconstructed from 2 f_0 Samples



(c) Waveform Reconstructed from 4/3 f_0 Samples

Fig.2.3: Effect of sampling frequency

Digital-To-Analog Converters (DACs) transform digital values to analog signals. DACs use either current or voltage switching techniques to produce an output analog value equal to the sum of several discrete analog values. Because it is easier to sum currents (rather than voltages) using analog circuitry, most commonly available DACs are current-mode devices. They produce the sum of internal current sources and use either an internal or external op amp as an output current-to-voltage converter.

Most common types of DACs are Fully Decoded DAC, Weighted Resistor DAC, Resistor Quad DAC and R-2R Ladder DAC. Fig. 2.4 shows 8-bit DAG using R-2R resistor ladder.

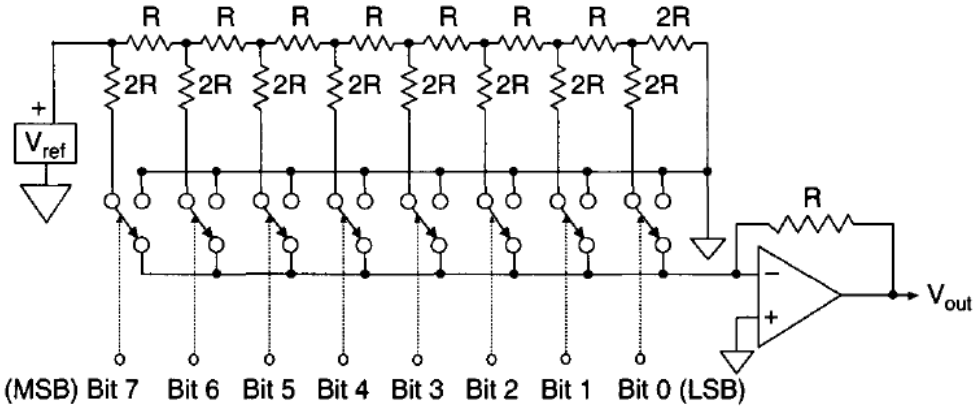


Fig.2.4: 8-bit R-2R Resistor Ladder DAC.

Some important criteria must be considered when choosing a DAC. The first parameter to determine is the number of bits of resolution. This is selected by knowing the desired dynamic range of the output signal. 8-, 12-, and 14-bit DACs are commonly available as monolithic devices or integrated circuits (ICs). Even 16-bit DACs are produced commercially. Another major parameter is settling time, which determines the speed of conversion, as shown graphically in Fig.2.5. This is the amount of time required for a DAC to move to and stay within its new output value. Linearity is another major DAC parameter. It is the maximum deviation of the DACs transfer curve from an ideal straight line, usually expressed as a fraction of the full-scale reading. One final DAC parameter to note is monotonicity. If the output of a DAC always increases for increasing digital input, the DAC is considered monotonic. Monotonicity is specified over a certain number of input bits, typically the full number of bits of resolution. A nonmonotonic DAC would have a dip in its transfer curve.

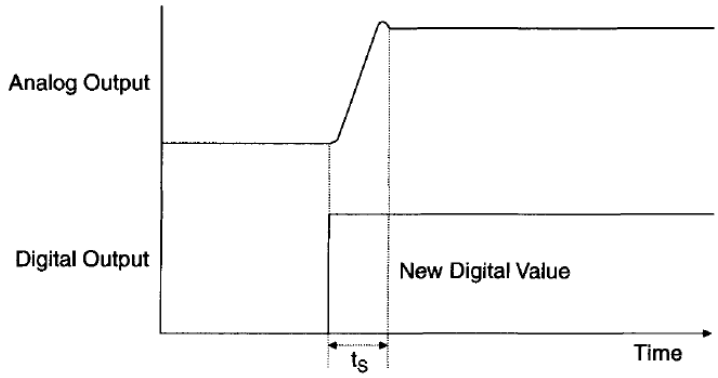


Fig.2.5: Setting Time of DACs.

There are monolithic DACs available with update rates in the range of 100-300 MSPS. These DACs can go up to 16-bit resolution (although 12- and 14-bit devices are more common at these high speeds). High-speed DACs typically employ a mixed architecture to achieve good performance at these speeds. Most use a segmented current source along with an R-2R ladder. The important specifications for these high-speed converters are update rate (in MSPS), settling time (in nanosec) and slew rate (in V/microsec).

Some of these fast DACs require emitter-coupled logic (ECL) digital control signals. As opposed to TTL digital signal levels , ECL signals are negative (relative to ground) and have a smaller difference between logic 0 and 1 levels. ECL logic devices are one of the fastest families of digital ICs commonly available. Some high-speed DACs use external TTL controls and translate them internally into ECL signals. ECL devices are powered by a -5.2 V supply (compared to +5 V or lower for TTL ICs). There are several different ECL families, with typical logic levels of -1.75 V representing 0 and -0.9 V representing 1.

2.3 DAQ systems using internet

The recent growth of networks and especially the spread of the Internet have boosted the development of distributed measurement systems for a variety of applications. Most distributed measurement systems follow the Client/Server architecture, as illustrated in Fig.2.6. According to this architecture, one or more instruments are connected to a measuring station, which operates as a server, while the acquired data are available through a network to the clients.

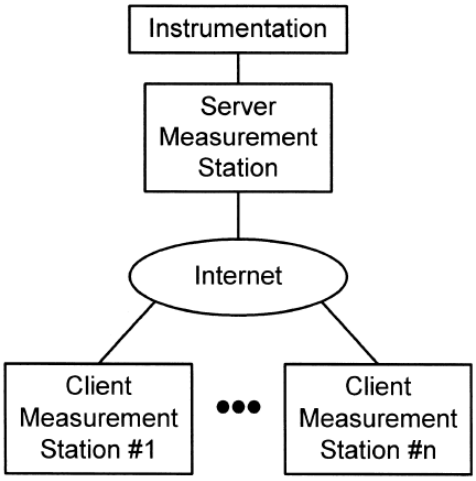


Fig.2.6: Distributed measurement system based on Client/Server architecture

Reference [40] describes a data aqcuisition system for remote monitoring of renewable energy systems, Fig.2.7 shows block diagram of developed system which consists of:

1- A number of data acquisition units (Fig. 2.7a), each connected to a set of sensors mounted on the RES system collects the desired data and communicates with the corresponding RES-Collector via an RF transceiver.
2- A number of computers each equipped with an RF transceiver connected to its RS-232 port and an interconnection to an Ethernet network or to the Internet. A program responsible for the collection of data from each RES station.
3- A computer interconnected to the Internet, run ning a program which collects data from all RES- Collectors and properly routes them to the users (RES-Server).
4- A number of users' computers interconnected to the Internet, which display the data using any browser, by means of the presentation interface (RES-Applet).

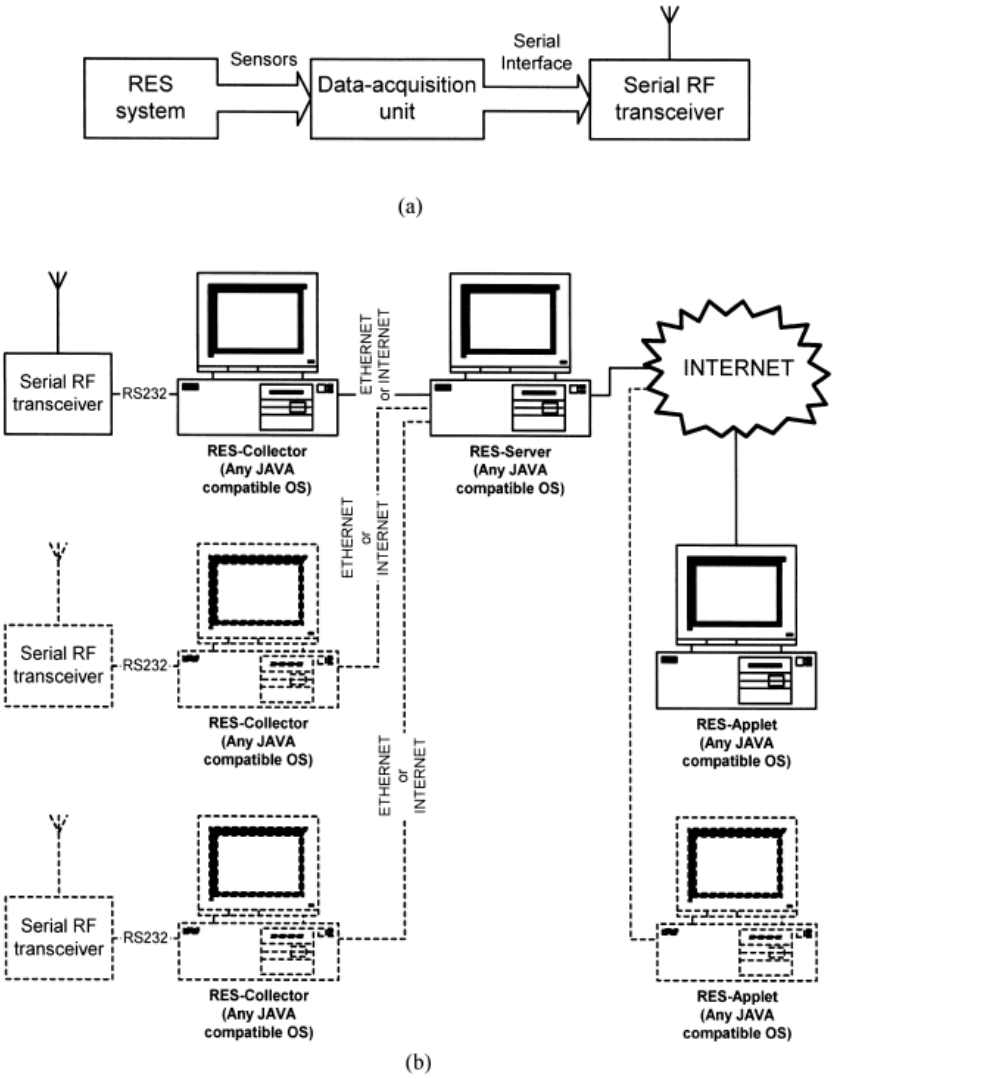


Fig.2.7: DAQ system Diagram, (a) DAQ unit, (b) Distributed networked system

In reference [41] a Railway Open Maintenance tool (RoMain) was presented, which is a web-based monitoring tool for trains that supports maintenance work. Fig.2.8 shows the RoMain system, the objective of this system was not to replace the existing control network, but rather to enhance it with a parallel, low-cost, online data network for railways, in order to support maintenance work. This data network allows maintenance staff to supervise railway equipment from anywhere at anytime. It also enables experts at different locations to collaborate and to anticipate maintenance tasks.

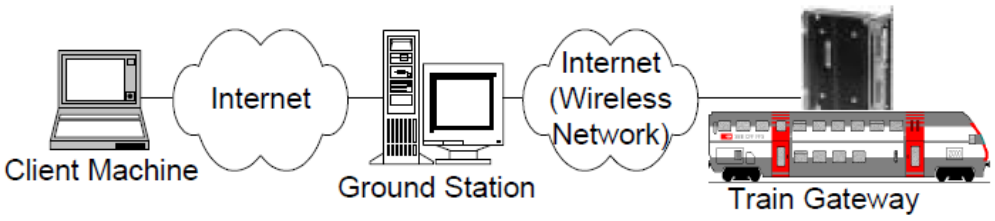


Fig 2.8: RoMain system.

2.4 Summary

In this chapter we reviewed the state of the art of DAQ systems. The main conclusion from the web-based DAQ system is that cost-effective maintenance of globally distributed devices, which is an important issue in industrial applications, which can be achieved by using Internet technology. Internet technology here means, the world wide TCP/IP network, protocols as HTTP and PPP, markup languages as HTML and XML, Internet browsers capable of downloading and running Java applets, web servers and application servers.

3 NETWORKED CONTROL SYSTEMS

3.1 Introduction to NCSs

Networked control systems (NCS) have been one of the main research focuses in academia as well as in industrial applications for many decades. A control system is a device or set of devices to manage, command, direct or regulate the behavior of other devices or systems.

The classical definition of NCS can be as follows: When a traditional feedback control system is closed via a communication channel, which may be shared with other nodes outside the control system, then the control system is called an NCS. An NCS can also be defined as a feedback control system wherein the control loops are closed through a real-time network. The defining feature of an NCS is that information (reference input, plant output, control input, etc.) is exchanged using a network among control system components (sensors, controllers, actuators, etc., see Fig.3.1).

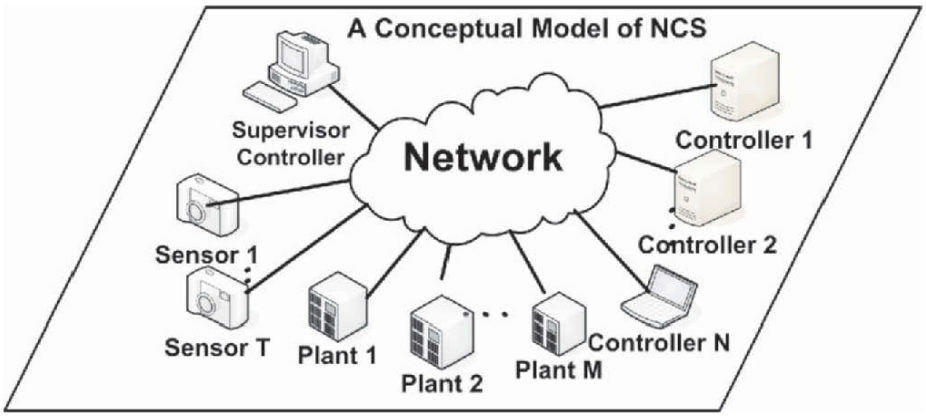


Fig.3.1: Typical Networked Control System

For many years now, data networking technologies have been widely applied in industrial and military control applications. These applications include manufacturing plants, automobiles, and aircraft. Connecting the control system components in these applications, such as sensors, controllers, and actuators, via a network can effectively reduce the complexity of systems, with nominal economical investments. Furthermore, network controllers allow data to be shared efficiently. It is easy to fuse the global information to take intelligent decisions over a large physical space. They eliminate unnecessary wiring. It is easy to add more sensors, actuators and controllers with very little cost and without heavy structural changes to the whole system. Most importantly, they connect cyber space to physical space making task execution from a distance easily accessible These systems are becoming more realizable today and have a lot of potential applications including factory automation, remote diagnostics and troubleshooting, hazardous environments and many other applications.

NCS components have to enable four functions which form the basis of the function an NCS is required to project. These basis functions are information acquisition (sensors), command (controllers), and communication and control (actuators).

The two major types of control systems that utilize communication networks are shared-network control systems and remote control systems. Using shared-network resources to

transfer measurements, from sensors to controllers and control signals from controllers to actuators, can greatly reduce the complexity of connections. This method, as shown in Fig.3.2, is systematic and structured, provides more flexibility in installation, and eases maintenance and troubleshooting. a remote control system can be thought of as a system controlled by a controller located far away from it. This is sometimes referred to as tele-operation control. Remote data acquisition systems and remote monitoring systems can also be included in this class of systems. The place where a central controller is installed is typically called a "local site", while the place where the plant is located is called a "remote site". See Fig. 3.3.

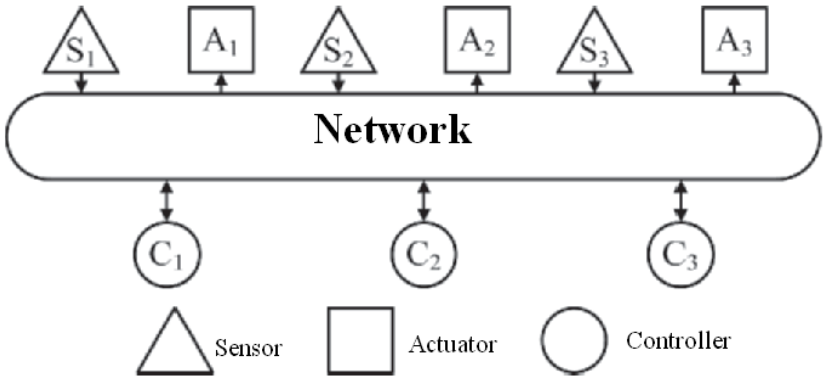


Fig.3.2: Shared-Network Control System

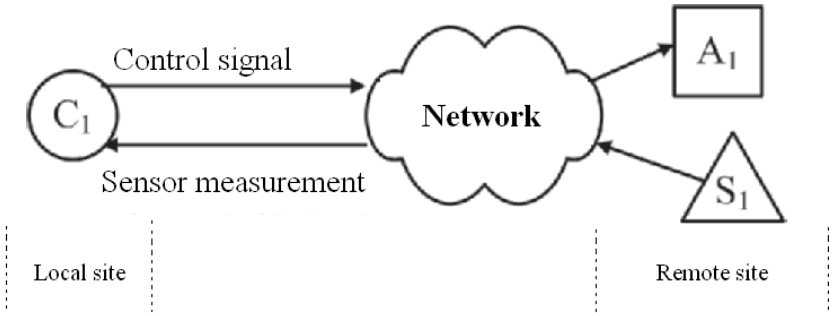


Fig.3.3: Remote Control System.

3.2 Network protocols for NCS:

Layering in network technology separates functional elements of a network into layers and then very carefully defines how each layer interacts with the layers above and below it. The classic work on layered design of networks is the Open Systems Interconnection (OSI) model from the International Organization for Standardization. Although this model is not used directly in commercial networking software, it forms the basis for design and construction of many networking technologies. The OSI model has seven layers:

1- Physical layer: Defines the physical and electrical characteristics of the network.
2- Data link layer: Defines the strategy for sharing the physical medium.
3- Network layer: Provides a means for communicating among open systems to establish, maintain and terminate network connections.
4- Transport layer: Ensures data reliability and integrity to the Session Layer.

5- Session layer: Provides for two communicating entities to exchange data with each other.
6- Presentation layer: This is where application data is either packed or unpacked, ready for use by the running application.
7- Application Layer: This layer is for end-user and end-application protocols.

By far the most widely used network technology is the Transmission Control Protocol operating over the Internet Protocol (TCP/IP). TCP/IP is based on a four-layer model that was largely designed before the final release of the OSI model. The TCP/IP model consists of:

1- Link layer.
2- Network layer.
3- Transport layer.
4- Application layer.

A survey of the types of control networks used in industry shows a wide variety of networks in use; see Tab 3.1. The networks are classified according to type: random access (RA) with collision detection (CD) or collision avoidance (CA), or time-division multiplexed (TDM) using token-passing (TP) or master-slave (MS).

Tab. 3.1: Worldwide most popular control networks.

Network	Type	Users	Application domain
Ethernet	RA/CD	50%	Various
Profibus	TDM/(TP and MS)	26%	Process control
CAN-based	RA/CA	25%	Automotive, process
Modbus	TDM/MS	22%	Various
ControlNet	TDM/TP	14%	Plant bus
ASI	TDM/MS	9%	Building systems
Interbus-S	TDM/MS	7%	Manufacturing
Fieldbus Foundation	TDM/TP	7%	Chemical industry

Note that the totals are more than 100% because many companies use more than one type of bus.

In this subsection, we discuss the Medium Access Control (MAC) sublayer protocol of Ethernet type control networks (including hub, switch, and wireless varieties). Ethernet generally uses the Carrier Sense Multiple Access (CSMA) with CD or CA mechanisms for resolving contention on the communication medium. There are three common flavors of Ethernet: First is hub-based Ethernet, which is common in office environments and is the most widely implemented form of Ethernet, second is switched Ethernet, which is more common in manufacturing and control environments, and third is wireless Ethernet.

Hub-based Ethernet uses hub(s) to interconnect the devices on a network. When a packet comes into one hub interface, the hub simply broadcasts the packet to all other hub interfaces. Hence, all of the devices on the same network receive the same packet simultaneously, and message collisions are possible. Collisions are dealt with utilizing the CSMA/CD protocol as specified in the IEEE 802.3 network standard.

This protocol operates as follows: when a node wants to transmit, it listens to the network. If the network is busy, the node waits until the network is idle; otherwise it transmits immediately. If two or more nodes listen to the idle network and decide to transmit simultaneously, the messages of these transmitting nodes collide and the messages are

corrupted. While transmitting, a node must also listen to detect a message collision. On detecting a collision between two or more messages, a transmitting node stops transmitting and waits a random length of time to retry its transmission. This random time is determined by the standard binary exponential backoff (BEB) algorithm: the retransmission time is randomly chosen between 0 and $(2^i - 1)$ slot times, where i denotes the ith collision event detected by the node and one slot time is the minimum time needed for a round-trip transmission. However, after 10 collisions have been reached, the interval is fixed at a maximum of 1023 slots. After 16 collisions, the node stops attempting to transmit and reports failure back to the node microprocessor. Further recovery may be attempted in higher layers.

The Ethernet frame format is shown in Fig. 3.4. The total overhead is 26 (=22+4) bytes. The data packet frame size is between 46 and 1500 bytes.

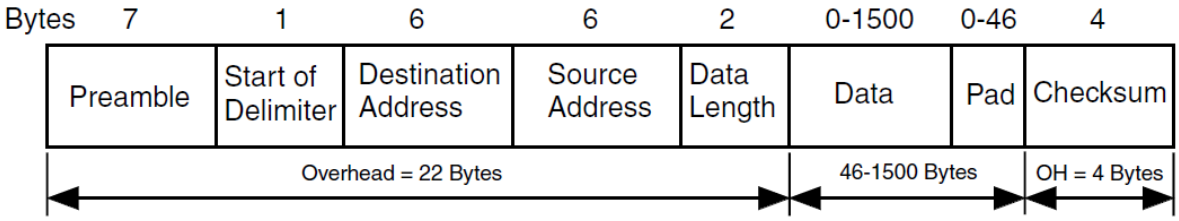


Fig.3.4: Ethernet (CSMA/CD) frame format

Switched Ethernet utilizes switches to subdivide the network architecture, thereby avoiding collisions, increasing network efficiency, and improving determinism. It is widely used in manufacturing applications. The main difference between switch-based and hub-based Ethernet networks is the intelligence of forwarding packets. Hubs simply pass on incoming traffic from any port to all other ports, whereas switches learn the topology of the network and forward packets to the destination port only. In a star-like network layout, every node is connected with a single cable to the switch. Thus, collisions can no longer occur on any network cable.

Switches employ the cut-through or store-and-forward technique to forward packets from one port to another, using per-port buffers for packets waiting to be sent on that port. Switches with cut-through first read the MAC address and then forward the packet to the destination port according to the MAC address of the destination and the forwarding table on the switch. On the other hand, switches with store-and-forward examine the complete packet first. Using the cyclic redundancy check (CRC) code, the switch will first verify that the frame has been correctly transmitted before forwarding the packet to the destination port. Although there are no message collisions on the networks, congestion may occur inside the switch when one port suddenly receives a large number of packets from the other ports. Three main queuing principles are implemented inside the switch in this case. They are first-in-first-out (FIFO) queue, priority queue, and per-flow queue. The FIFO queue is a traditional method that is fair and simple.

Wireless Ethernet, based on the IEEE 802.11 standard, can replace wired Ethernet in a transparent way since it implements the two lowest layers of the International Standards Organization (ISO)/Open Systems Interconnection (OSI) model. Besides the physical layer, the biggest difference between 802.11 and 802.3 is in the medium access control. Unlike wired Ethernet nodes, wireless stations cannot "hear" a collision. A collision avoidance mechanism is used but cannot entirely prevent collisions. Thus, after a packet has been successfully received by its destination node, the receiver sends a short acknowledgment

packet (ACK) back to the original sender. If the sender does not receive an ACK packet, it assumes that the transmission was unsuccessful and retransmits.

3.3 Discrete-time systems:

Practically all networked control systems that are implemented today are based on computer control or digital controllers, which use descrete-time control. It is therefore important to understand descrete-time control systems as well.

Fig. 3.5 shows block diagram of digital control system with discrete time controller. In such a digital control system, some points of the system pass signals of varying amplitude in either continuous time or discrete time, while other points pass signals in numerical code, as depicted in the figure.

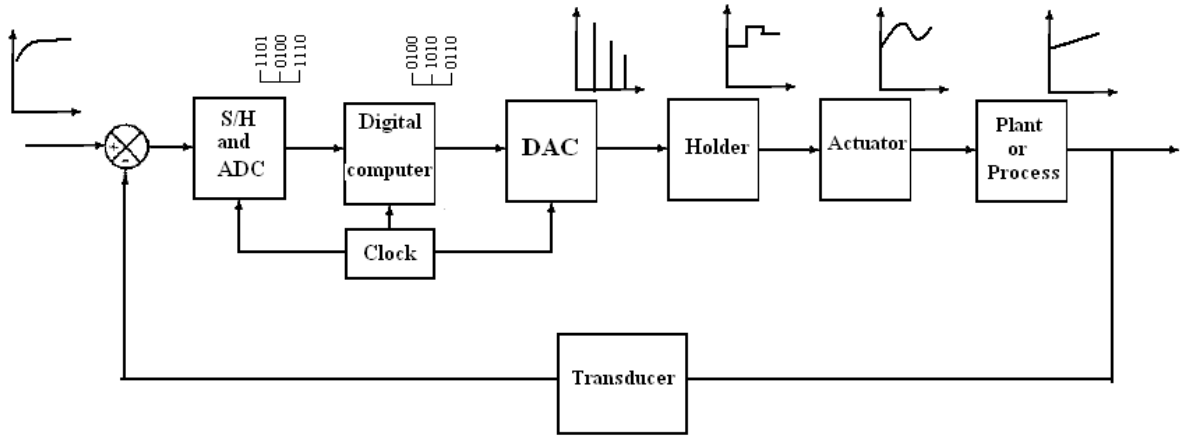


Fig. 3.5: Block diagram of digital control system.

The output of the plant is a continuous-time signal. The error signal is converted into digital form by the sample-and-hold circuit (S/H) and the analog-to-digital converter (ADC), The conversion is done at the sampling time. The digital computer processes the sequences of numbers by means of an algorithm and produces new sequences of numbers.. At every sampling instant a coded number (usually a binary number consisting of eight or more binary digits) must be converted to a physical control signal, which is usually a ountinuous-time or analog signal. The digital-to-analog converter (DAC) and the hold circuit convert the sequence of numbers in numerical code into a piecewise continuous-time signal. The real-time clock in the computer synchronizes the events, The output of the hold circuit, a continuous-time signal, is fed to the plant, either directly or through the actuator, to control its dynamics. The operation that transforms continuous-time signals into discrete-time data is called sampling or discretization.While the reverse operation is called data-hold.

A mathematical tool Commonly used for the analysis and synthesis of discrete-time control systems is the Z transform. The role of the Z transform in discrete-time systems is similar to that of the Laplace transform in continuous time systems. The z transform method is an operational method that is very powerful when working with discrete-time systems. In considering the Z transform of a time function $x(t)$, we consider only the sampled values of $x(t)$, that is: $x(0), x(T), x(2T), \ldots$, where T is sampling period. The Z transform of $x(t)$ is defined by the following equation:

$$X(z) = \mathcal{Z}[x(t)] = \mathcal{Z}[x(kT)] = \sum_{k=0}^{\infty} x(kT)z^{-k} \quad (3.1)$$

Expansion of the right-hand side of Equation (3.1) gives:

$$X(z) = x(0) + x(T)z^{-1} + x(2T)z^{-2} + \cdots + x(kT)z^{-k} + \cdots \quad (3.2)$$

Equation (3.2) implies that the Z transform of any continuous-time function $x(t)$ may be written in the series form. The z^{-k} in this series indicates the position in time at which the amplitude $x(kt)$ occurs. Conversely, if $X(z)$ is given in the series form as above, the inverse Z transform can be obtained by inspection as a sequence of the function $x(kT)$ that corresponds to the values of $x(t)$ at the respective instants of time.

If the Z transform is given as a ratio of two polynomials in z, then the inverse Z transform may be obtained by several different methods, such as the direct division method, the computational method, the partial-fraction expansion method, and the inversion integral method.

Discrete time systems are usually characterized by difference equations. Difference equations can be easily expressed and analyzed by Z transform. Let's consider the time-invariant discrete-time system characterized by the following linear difference equation:

$$x(k) + a_1x(k-1) + \cdots + a_nx(k-n) = b_0u(k) + b_1u(k-1) + \cdots + b_nu(k-n) \quad (3.3)$$

Where $u(k)$ and $x(k)$ are the system input and output respectively. This equation can be expressed in the z term by taking the Z-transform of each term in the equation according to following equations:

$$\mathcal{Z}[x(k+n)] = z^nX(z) - z^nx(0) - z^{n-1}x(1) - \cdots - zx(n-1) \quad (3.4)$$

$$\mathcal{Z}[x(k)] = X(z) \quad (3.5)$$

$$\mathcal{Z}[x(k-n)] = z^{-n}X(z) \quad (3.6)$$

The stability analyzes of a discrete-time closed-loop control system defined by the following pulse-transfer function:

$$\frac{X(z)}{U(z)} = \frac{G(z)}{1+H(z)} \quad (3.7)$$

may be determined from the locations of the closed-loop poles in Z plane, or the roots of the characteristic equation:

$$P(z) = 1 + H(z) = 0 \quad (3.8)$$

There are many stability tests that can be applied directly to the above characteristic equation without solving it to find its roots, such as Schur-Cohn stability test, Jury stability test and stability analysis by use of the Bilinear transformation and Routh stability criterion.

Fig. 3.6 show the transient response for unit-step input for the system described by the following transfer function:

$$\frac{X(z)}{U(z)} = \frac{0.46z-0.33}{z^2-1.53z+0.66} \tag{3.9}$$

The transient response of a digital control system may be characterized not only by the damping ratio and damped natural frequency, but also by the rise time, maximum overshoot, settling time,peak time and delay time.

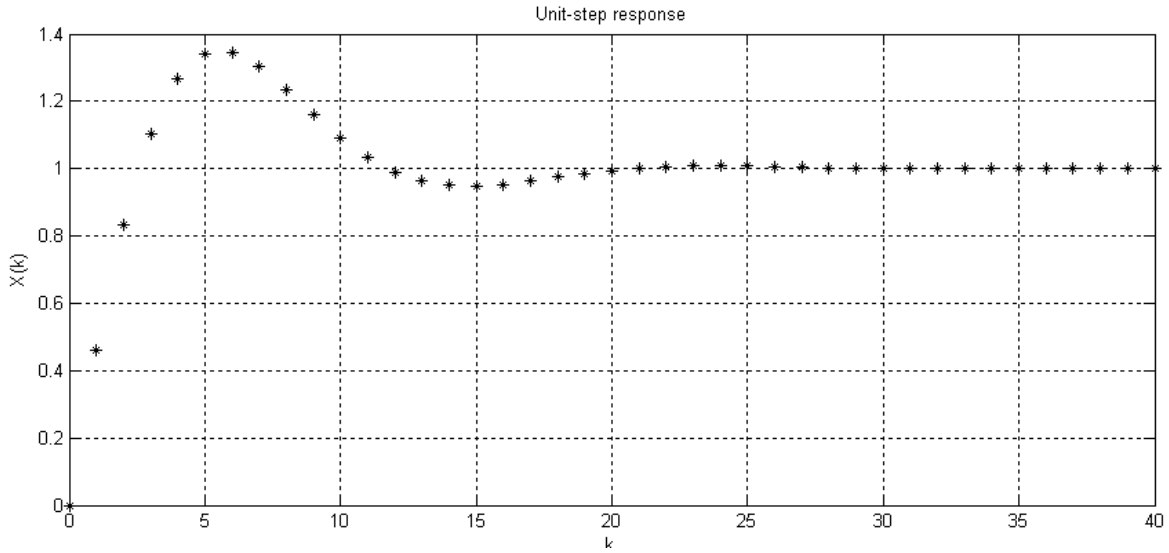


Fig. 3.6: Unit-step response of the system defined by Equation (3.9)

The transient response characteristics of the discrete-time control system depends on the sampling period *T*. A large sampling period has detrimmental effects on the relative stability of the system. A rule of thumb is eight to ten times sampling during one cycle of the damped sinusoidal oscillations of the closed-loop system output, if it is underdamped. For overdamped systems, sample eight to ten times during the rise time in the step response.

For a given value of gain *K*, increasing the sampling period *T* will make the discrete-time control system less stable and eventually will make it unstable. Conversely, making the sampling period *T* shorter allows the critical value of the gain K for stability to be larger: In fact, making the sampling period shorter tends to make the system behavior more like the continuous-time system.

For systems with sinusoidal input frequency-response concept plays the same powerful role in digital control systems as it does in continuous-time control systems. Frequency-response methods have very frequently been used in the compensator design. The basic reason is the simplicity of the methods. When performing frequency-response tests on a discrete-time system, it is important that the system have a low-pass filter before the sampler so that sidebands are filtered out. Then the response of the linear time-invariant system to a sinusoidal input preserves the frequency and modifies only the amplitude and phase of the input signal Thus, the amplitude and phase are the only two quantities that must be dealt with.

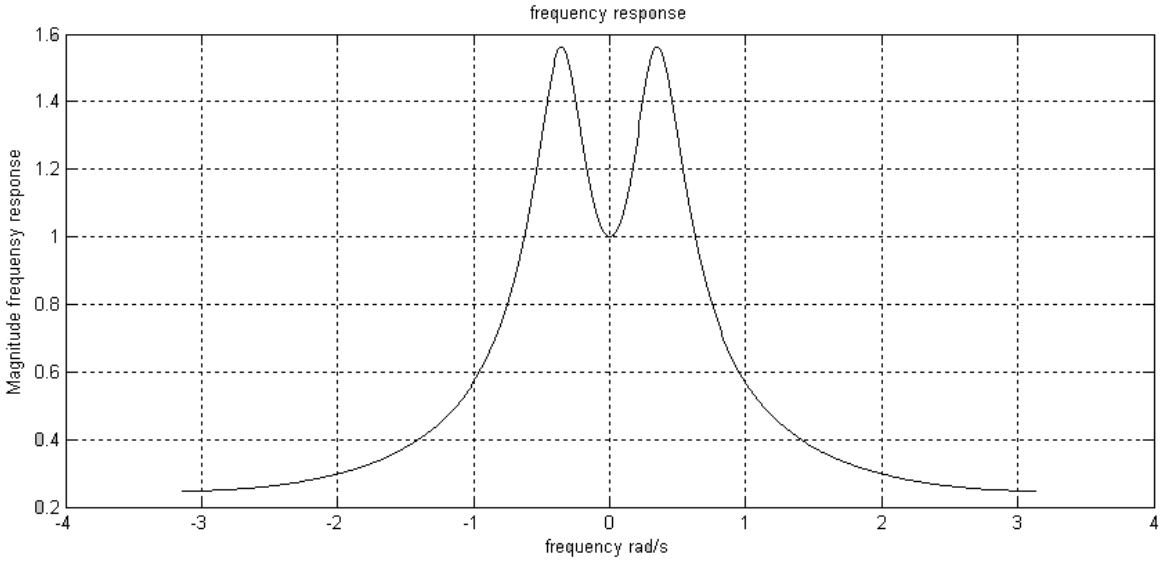


Fig. 3.7: Frequency responde of the system defined by Equation (3.9)

Fig. 3.7 shows the frequency response of the system defined by Equation (3.9).

4 SYNCHRONOUS MACHINE MODEL

4.1 Introduction

Three-phase synchronous machines are widely used for power generation and large motor drives. The stator of a synchronous machine consists of a stack of laminated ferromagnetic core with internal slots, a set of three-phase distributed stator windings placed in the core slots, and an outer framework with end shields and bearings for the rotor shaft. The turns of the stator windings are equally distributed over pole-pairs, and the phase axes are spaced 2π / 3 electrical radians apart.

The cross-sectional shape of the rotor can be salient or cylindrical. Salient pole construction is mostly used in low-speed applications where the diameter to length ratio of the rotor can be made larger to accommodate the high pole number. Salient pole synchronous machines are often used in hydro generators to match the low operating speed of the hydraulic turbines. The short, pancake-like rotor has separate pole pieces bolted onto the periphery of a spider-web-like hub. Salient here, refers to the protruding poles; the alternating arrangement of pole iron and interpolar gap results in preferred directions of magnetic flux paths or magnetic conductivity.

The cylindrical or round rotor construction is favoured in high-speed applications where the diameter to length ratio of the rotor has to be kept small to keep the mechanical stresses from centrifugal forces within acceptable limits. Two- and four-pole cylindrical rotor synchronous machines are used in turbo generators to efficiently match the high operating speed of steam turbines. The long cylindrical rotors are usually produced from solid castings of chromium-nickel-molybdenum steel, with axial slots for the field winding on both sides of a main pole.

Direct current excitation of the field winding can be supplied through a pair of insulated slip rings mounted on the rotor shaft. Alternatively, the dc excitation can be obtained from the rectified output of a small alternator mounted on the same rotor shaft of the synchronous machine. The second excitation method eliminates the slip rings and it is called brushless excitation.

In the basic two-pole representation of a synchronous machine, the axis of the pole is called the direct or d-axis. The quadrature, or q-axis, is defined in the direction 90 electrical degrees ahead of the direct axis. Under no-load operation with only field excitation, the field mmf will be along the d-axis, and the stator internal voltage, $d\lambda_{af}/dt$, will be along the q-axis.

The mathematical description and model developed in this section is based on the concept of an ideal synchronous machine with two basic poles. The fields produced by the winding currents are assumed to be sinusoidally distributed around the air-gap. This assumption of sinusoidal field distribution ignores the space harmonics, which may have secondary effects on the machine's behaviour. It is also assumed that the stator slots cause no appreciable variation of any of the rotor winding inductances with rotor angle. Although saturation is not explicitly taken into account in this model, it can be accounted by adjusting the reactances along the two axes with saturation factors, or by adding a compensating component to the main field excitation.

Although the rotor may have only one physically identifiable field winding, additional windings are often used to represent the damper windings and the effects of current flow in the rotor iron. For the salient-pole rotor machine, usually two additional windings are used,

one on the d-axis and the other on the q-axis. Over the years, experience on power system simulation has shown that most synchronous generators can be adequately represented by a model that is based on an equivalent idealized machine with one or two sets of damper windings, besides the field winding. Damper windings in the equivalent machine model can be used to represent physical amortisseur windings, or the damping effects of eddy currents in the solid iron portion of the rotor poles. Fig. 4.1 shows a circuit representation of an idealized machine model of the synchronous machine commonly used in analysis.

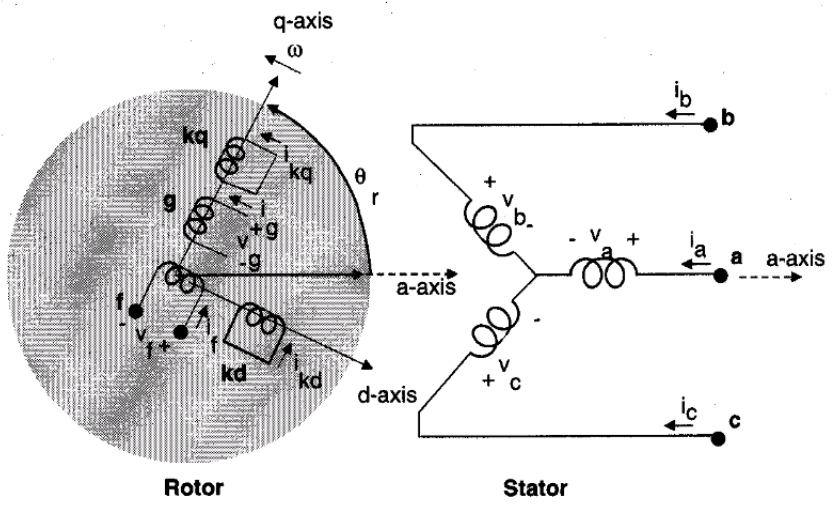


Fig. 4.1: Circuit representation of an idealized synchronous machine

4.2 Steady state model

4.2.1 Mathematical model

Voltage equation of a synchronous machine is:

$$\begin{bmatrix} \mathfrak{v}_s \\ \mathfrak{v}_r \end{bmatrix} = \begin{bmatrix} \mathfrak{r}_s & 0 \\ 0 & \mathfrak{r}_r \end{bmatrix} \begin{bmatrix} i_s \\ i_r \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} \Lambda_s \\ \Lambda_r \end{bmatrix} \tag{4.1}$$

Where:

$$\mathfrak{v}_s = [v_a, \quad v_b, \quad v_c]^t \tag{4.2}$$

$$\mathfrak{v}_r = [v_f, \quad v_{kd}, \quad v_g, \quad v_{kq}]^t \tag{4.3}$$

$$i_s = [i_a \quad i_b \quad i_c]^t \tag{4.4}$$

$$i_r = [i_f \quad i_{kd} \quad i_g \quad i_{kq}]^t \tag{4.5}$$

$$\mathfrak{r}_s = diag[r_a \quad r_b \quad r_c] \tag{4.6}$$

$$\mathfrak{r}_r = diag[r_f \quad r_{kd} \quad r_g \quad r_{kq}] \tag{4.7}$$

$$\Lambda_s = [\lambda_a, \quad \lambda_b, \quad \lambda_c]^t \tag{4.8}$$

$$\Lambda_r = \left[\lambda_f, \quad \lambda_{kd}, \quad \lambda_g, \quad \lambda_{kq}\right]^t \tag{4.9}$$

The symbols per phase parameters are as follows:

r_s	Armature or stator winding resistance.
r_f	d-axis field winding resistance.
r_g	q-axis field winding resistance.
r_{kd}	d-axis damper winding resistance.
r_{kq}	q-axis damper winding resistance.
$\lambda_a, \quad \lambda_b, \quad \lambda_c$	stator winding flux linkages.
λ_f	d-axis field winding flux linkage.
λ_g	q-axis field winding flux linkage.
λ_{kd}	d-axis damper winding flux linkage.
λ_{kq}	q-axis damper winding flux linkage.

We shall transform the stator quantities to a rotating qd0 reference frame that is attached to the machine rotor. The resulting voltage equation has time-invariant coefficients. In the idealized machine, the axes of the rotor windings are already along the q- and d-axes, and the qd0 transformation need only to be applied to the stator winding quantities. In vector notation, we define the augmented transformation matrix which called Park matrix or Park transformation:

$$C = \begin{bmatrix} T_{qd0}(\theta_r) & 0 \\ 0 & U \end{bmatrix} \tag{4.10}$$

Where θ_r is the electrical angel, U is a unit matrix and:

$$T_{qd0}(\theta_r) = \frac{2}{3}\begin{bmatrix} \cos(\theta_r) & \cos(\theta_r - \frac{2\pi}{3}) & \cos(\theta_r + \frac{2\pi}{3}) \\ \sin(\theta_r) & \sin(\theta_r - \frac{2\pi}{3}) & \sin(\theta_r + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \tag{4.11}$$

Then the transformed qd0 voltages, currents, and flux linkages of the stator will be:

$$\mathfrak{v}_{qd0} = T_{qd0}(\theta_r)\mathfrak{v}_s = \left[v_q, \quad v_d, \quad v_0\right]^t \tag{4.12}$$

$$\mathfrak{i}_{qd0} = T_{qd0}(\theta_r)\mathfrak{i}_s = \left[i_q, \quad i_d, \quad i_0\right]^t \tag{4.13}$$

$$\Lambda_{qd0} = T_{qd0}(\theta_r)\Lambda_s = \left[\lambda_q, \quad \lambda_d, \quad \lambda_0\right]^t \tag{4.14}$$

Fig. 4.2 shows qd-model of synchronous machine.

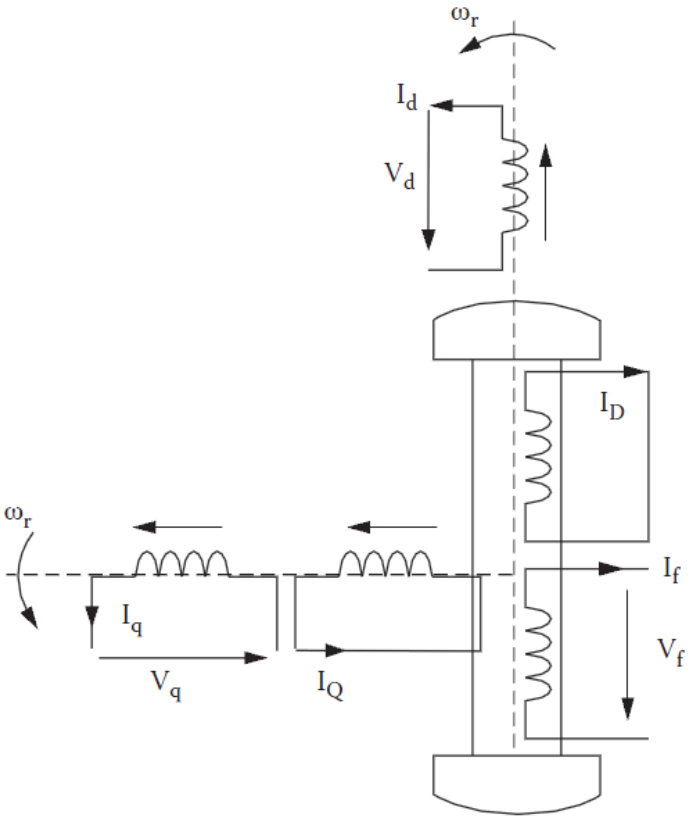


Fig. 4.2: The d–q model of synchronous machine

4.2.2 Voltages and currents equations

The winding equations of the synchronous machine model derived in last subsection can be implemented in a simulation that uses voltages as input and currents as output. The main inputs to the machine simulation are the stator abc phase voltages, the excitation voltage applied to the field windings, and the applied mechanical torque to the rotor.

Performing the transformation to qd0 reference frame of voltages yields:

$$v_q = \frac{2}{3}\left\{v_a cos(\theta_r(t)) + v_b \cos\left(\theta_r(t) - \frac{2\pi}{3}\right) + v_c \cos\left(\theta_r(t) - \frac{4\pi}{3}\right)\right\} \quad (4.15)$$

$$v_d = \frac{2}{3}\left\{v_a sin\,(\theta_r(t)) + v_b \sin\left(\theta_r(t) - \frac{2\pi}{3}\right) + v_c \sin\left(\theta_r(t) - \frac{4\pi}{3}\right)\right\} \quad (4.16)$$

$$v_0 = \frac{1}{3}(v_a + v_b + v_c) \quad (4.17)$$

Where:

$$\theta_r(t) = \int_0^t \omega_r(t)dt + \theta_r(0) \quad (4.18)$$

Where ω_r is rotor speed in rad/s.

Although the angle, $\theta_r(t)$, increases with time, but $\cos(\theta_r(t))$ and $\sin(\theta_r(t))$ will remain bounded. In simulation, the values of $\cos(\theta_r(t))$ and $\sin(\theta_r(t))$ can be obtained from

a variable-frequency oscillator circuit which has a provision for setting the proper initial value of θ_r .

The qd0 voltage equations can be used in the integral equations of flux linkages of the windings, the above stator qd0 voltages with other inputs can then be used in the integral equations to express the flux linkages of the windings. For the case of a machine with only one field winding in the d -axis and a pair of damper windings in the d - and q-axes, the integral equations of the winding flux linkages per second are as follows:

$$\psi_q = \omega_b \int \left\{ v_q - \frac{\omega_r}{\omega_b}\psi_d + \frac{r_s}{x_{ls}}(\psi_{mq} - \psi_q) \right\} dt \tag{4.19}$$

$$\psi_d = \omega_b \int \left\{ v_d + \frac{\omega_r}{\omega_b}\psi_q + \frac{r_s}{x_{ls}}(\psi_{md} - \psi_d) \right\} dt \tag{4.20}$$

$$\psi_0 = \omega_b \int (v_0 - \frac{\omega_r}{\omega_b}\psi_0)\, dt \tag{4.21}$$

$$\psi'_{kq} = \frac{\omega_b r'_{kq}}{x'_{lkq}} \int (\psi_{mq} - \psi'_{kq})\, dt \tag{4.22}$$

$$\psi'_{kd} = \frac{\omega_b r'_{kd}}{x'_{lkd}} \int (\psi_{md} - \psi'_{kd})\, dt \tag{4.23}$$

$$\psi'_f = \frac{\omega_b r'_f}{x_{md}} \int \left\{ E_f + \frac{x_{md}}{x'_{lf}}(\psi_{mq} - \psi'_f) \right\} dt \tag{4.24}$$

Where:

$$\psi_{mq} = \omega_b L_{mq}(i_q + i'_{kq}) \tag{4.25}$$

$$\psi_{md} = \omega_b L_{md}(i_d + i'_{kd} + i'_f) \tag{4.26}$$

$$E_f = x_{md}\frac{v'_f}{r'_f} \tag{4.27}$$

$$\psi_q = x_{ls} i_q + \psi_{mq} \tag{4.28}$$

$$\psi_d = x_{ls} i_d + \psi_{md} \tag{2.29}$$

$$\psi_0 = x_{ls} i_0 \tag{4.30}$$

$$\psi'_f = x'_{lf} i'_f + \psi_{md} \tag{4.31}$$

$$\psi'_{kd} = x'_{lkd} i'_{kd} + \psi_{md} \tag{4.32}$$

$$\psi'_{kq} = x'_{lkq} i'_{kq} + \psi_{mq} \tag{4.33}$$

Where: ψ is flux linkage per second, r is resistance, x_l is reactance, L is inductance and *i* is current. The subsequent q refer to q-axis quantities, d refer to d-axis quantities, m refer to mutual flux linkages or inductances, s refer to stator windings quantities, k refer to damper windings quantities, f refer to field windings quantities and the apostrophe refer quantities which are referred to the stator.

The above equations are in motoring convention, that is the currents, i_q andi_d, are in the positive polarity of the stator windings terminal voltages. Now, to handle the cut set of inductors in the q- and d -axis circuits, we will express the mutual flux linkages in terms of the total flux linkages of the windings as:

$$\psi_{mq} = x_{MQ}\left(\frac{\psi_q}{x_{ls}} + \frac{\psi'_{kq}}{x'_{lkq}}\right) \tag{4.34}$$

$$\psi_{md} = x_{MD}\left(\frac{\psi_d}{x_{ls}} + \frac{\psi'_{kd}}{x'_{lkd}} + \frac{\psi'_f}{x'_{lf}}\right) \tag{4.35}$$

Where:

$$\frac{1}{x_{MQ}} = \frac{1}{x_{mq}} + \frac{1}{x'_{lkq}} + \frac{1}{x_{ls}} \tag{4.36}$$

$$\frac{1}{x_{MD}} = \frac{1}{x_{md}} + \frac{1}{x'_{lkd}} + \frac{1}{x'_{lf}} + \frac{1}{x_{ls}} \tag{4.37}$$

Having the values of the flux linkages of windings and the mutual flux linkages among the d- and q-axes, we can determine the winding currents using following equations:

$$i_q = \frac{\psi_q - \psi_{mq}}{x_{ls}} \tag{4.38}$$

$$i_d = \frac{\psi_d - \psi_{md}}{x_{ls}} \tag{4.39}$$

$$i'_{kd} = \frac{\psi'_{kd} - \psi_{md}}{x'_{lkd}} \tag{4.40}$$

$$i'_{kq} = \frac{\psi'_{kq} - \psi_{mq}}{x'_{lkq}} \tag{4.41}$$

$$i'_f = \frac{\psi'_f - \psi_{md}}{x'_{lf}} \tag{4.42}$$

The stator winding qd currents can be transformed back to abc winding currents using the following rotor to stationaryqd, and stationary qd0 to abc transformations:

$$i_q^s = i_q \cos\left(\theta_r(t)\right) + i_d \sin\left(\theta_r(t)\right) \tag{4.43}$$

$$i_d^s = -i_q \sin\left(\theta_r(t)\right) + i_d \cos\left(\theta_r(t)\right) \tag{4.44}$$

$$i_c = i_q^s + i_0 \tag{4.45}$$

$$i_b = -\frac{1}{2} i_q^s - \frac{1}{\sqrt{3}} i_d^s + i_0 \tag{4.46}$$

$$i_c = -\frac{1}{2} i_q^s + \frac{1}{\sqrt{3}} i_d^s + i_0 \tag{4.47}$$

Fig.4.3 and Fig.4.4 show equivalent q and d circuits of qd0 model. While Fig. 4.5 and Fig. 4.6 illustrate d and q axes functional block diagrams.

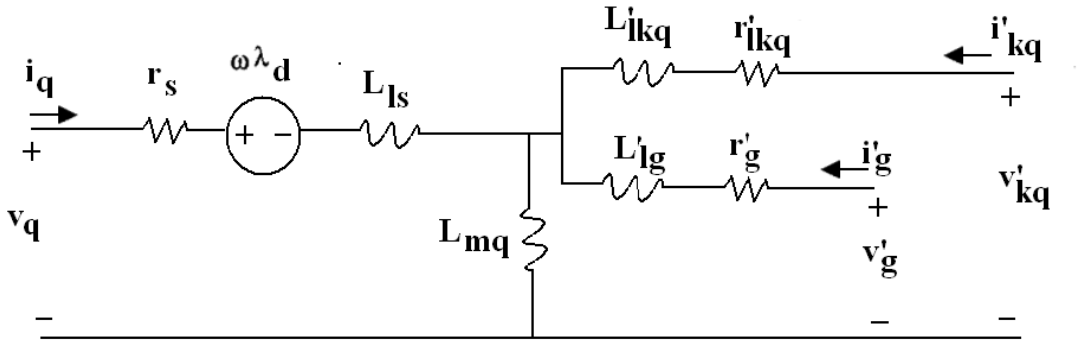


Fig. 4.3: q-axis equivalent circuit

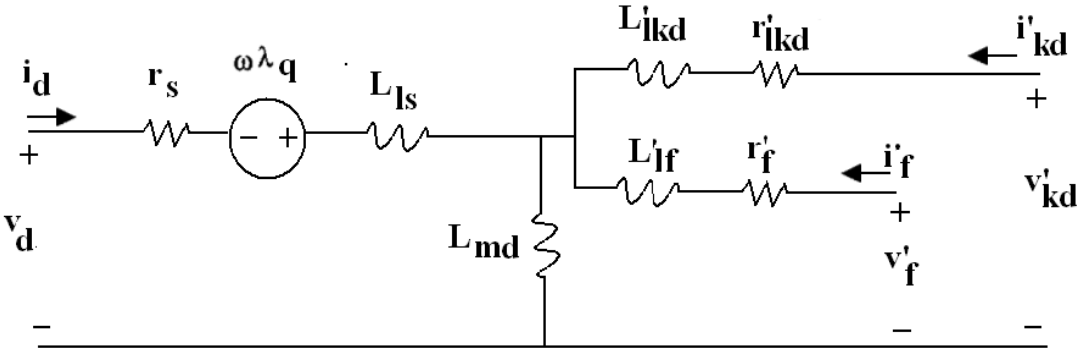


Fig. 4.4: d-axis equivalent circuit

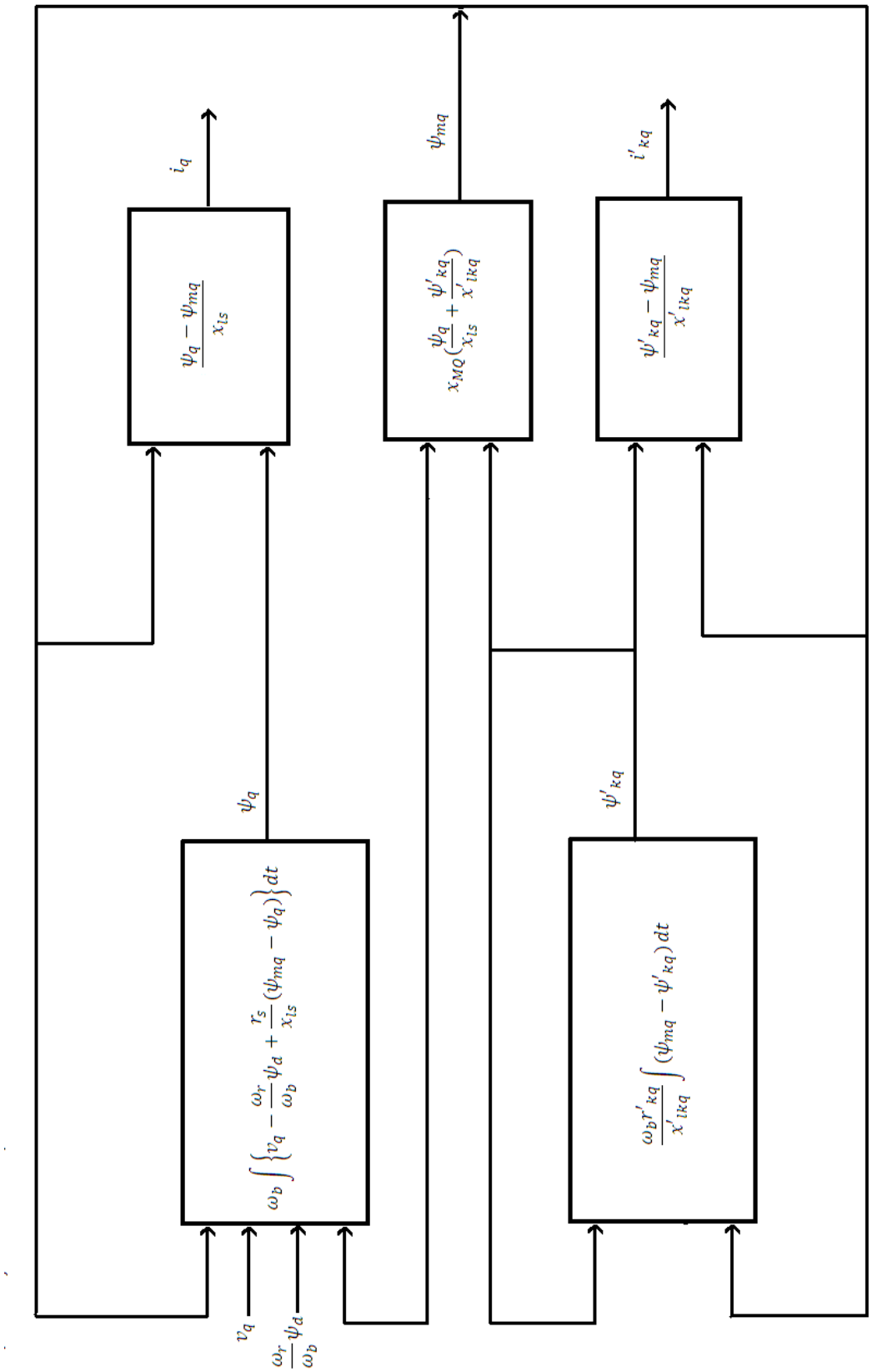


Fig. 4.5: q-axes functional block diagram

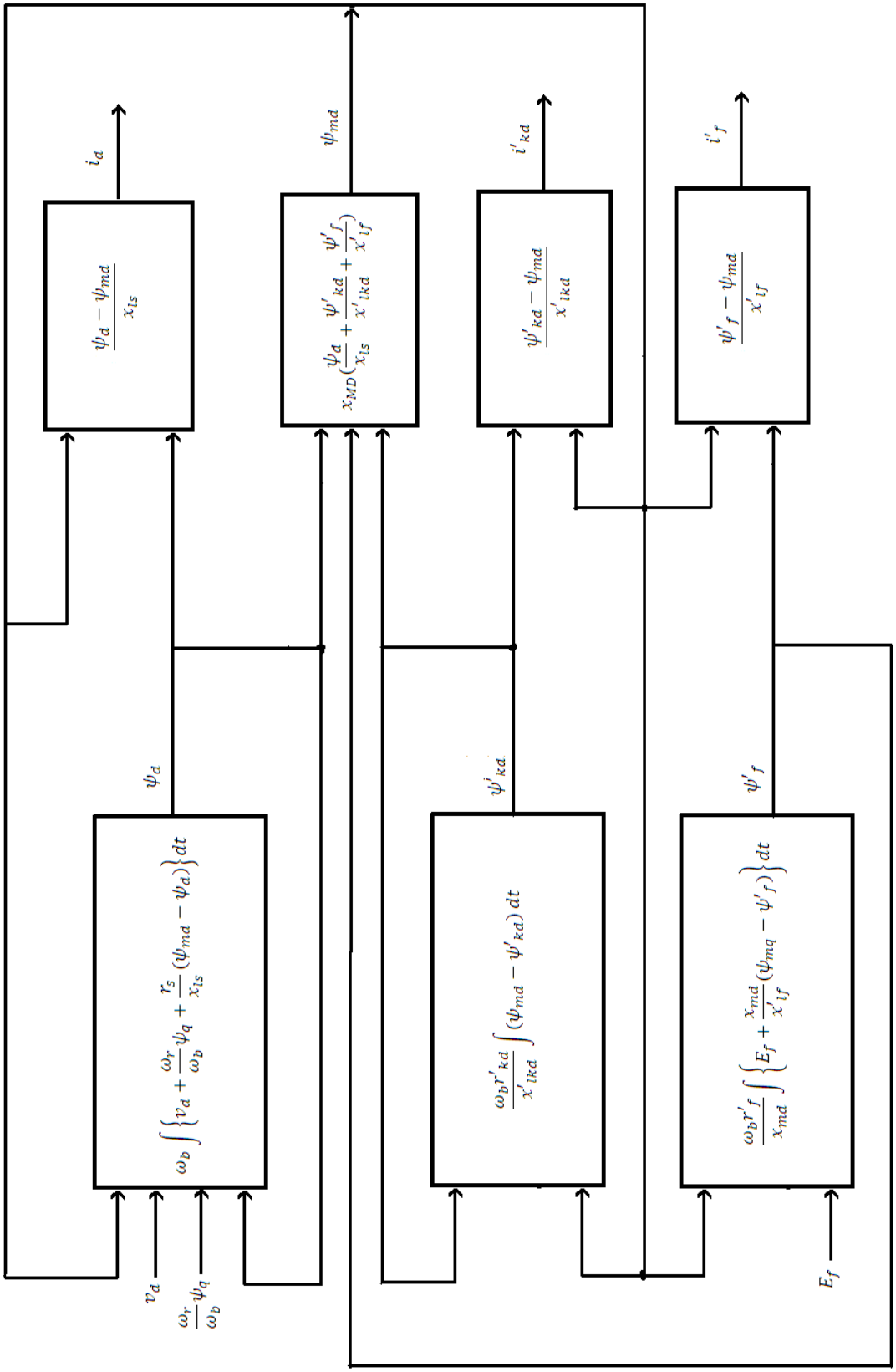


Fig. 4.6: d-axes functional block diagram

4.2.3 Torque equation

The expression for the electromagnetic torque developed by the machine can be obtained from the component of the input power that is transferred across the air-gap. The total input power is given by:

$$P_{in} = v_a i_a + v_b i_b + v_c i_c + v_f i_f + v_g i_g \tag{4.48}$$

When the stator phase quantities are transformed to the rotor qd0 reference frame that rotates at a speed of $\omega_r = \frac{d\theta_r}{dt}$, then equation (4.48) becomes:

$$P_{in} = \frac{3}{2}(v_q i_q + v_d i_d) + 3v_o i_o + v_f i_f + v_g i_g \tag{4.49}$$

Then:

$$P_{in} = \frac{3}{2}\left(r_s\left(i_q^2 + i_d^2\right) + i_q \frac{d\lambda_q}{dt} + i_d \frac{d\lambda_d}{dt} + \omega_r\left(\lambda_d i_q - \lambda_q i_d\right)\right) + 3i_0^2 r_0 + 3i_0 \frac{d\lambda_0}{dt} +$$

$$i_f^2 r_f + i_f \frac{d\lambda_f}{dt} + i_g^2 r_g + i_g \frac{d\lambda_g}{dt} \tag{4.50}$$

Eliminating terms in Equation (4.50) identified with the ohmic losses and the rate of change in magnetic energy, the above expression of the electromechanical power developed reduces to:

$$P_{em} = \frac{3}{2}\omega_r(\lambda_d i_q - \lambda_q i_d) \tag{4.51}$$

For a p-pole machine, $\omega_r = \frac{p}{2}\omega_{rm}$, with ω_{rm} is the rotor speed in mechanical radians per second. Thus, Equation (4.51) for a p-pole machine can also be written as:

$$P_{em} = \frac{3p}{4}\omega_r\left(\lambda_d i_q - \lambda_q i_d\right) \tag{4.52}$$

Dividing the electromechanical power by the mechanical speed of the rotor, we obtain the following expression for the electromechanical torque developed by a p-pole machine:

$$T_{em} = \frac{3p}{4}\left(\lambda_d i_q - \lambda_q i_d\right) \tag{4.53}$$

Or:

$$T_{em} = \frac{3p}{4\omega_b}\left(\psi_d i_q - \psi_q i_d\right) \tag{4.54}$$

The value of T_{em} in the above expression is positive for motoring operation and negative for generating operation.

Equation of motion of rotor is given by:

$$T_{em} + T_{mech} - T_{damp} = J\frac{d\omega_{rm}(t)}{dt} = \frac{2J}{p}\frac{d\omega_e(t)}{dt} \quad (4.55)$$

Where T_{mech} is the externally-applied mechanical torque in the direction of rotation, it will be negative when the machine is motoring a load and positive when the rotor is being driven by a prime mover as in generating; and T_{damp} is the frictional torque, acts always in a direction opposite to the rotor rotation, and J is the moment of inertia.

Since ω_e is constant then:

$$\frac{d\{\omega_r(t)-\omega_e\}}{dt} = \frac{d\omega_r(t)}{dt} \quad (4.56)$$

The slip speed can be determined from Equation (4.55) and Equation (4.56) by the integration:

$$\omega_r(t) - \omega_e = \frac{p}{2J}\int_0^t \left(T_{em} + T_{mech} - T_{damp}\right)dt \quad (4.57)$$

For studying power systems, where there are many transformers and a wide range of equipment ratings, there are many advantages of choosing an appropriate per unit system which will eliminate the bother of keeping track the primary and secondary side quantities of the transformers and also provide a rough but quick check on the values of the parameters. In the case of studying just one synchronous machine, the use of a per unit system offers no such advantage, other than perhaps the convenience of having the per unit parameters of the machine already available in terms of a set of base values that correspond to those of the rating of the machine. In such a situation, the base power, S_b , is the rated kVA of the machine.

For transient studies, the peak value rather than the rms value of the rated phase voltage is to be chosen as the base voltage, that the base voltage is:

$$V_b = \frac{\sqrt{2}}{\sqrt{3}}V_{line-to-line} \quad (4.58)$$

Similarly, choosing the peak value of the rated current as the base current, that is:

$$I_b = \frac{2}{3}\frac{S_b}{V_b} \quad (4.59)$$

The base values for the stator impedance, torque and mechanical angular frequency are given by:

$$Z_b = \frac{V_b}{I_b} \quad (4.60)$$

$$T_b = \frac{S_b}{\omega_{bm}} \quad (4.61)$$

$$\omega_{bm} = \frac{2\omega_b}{p} \quad (4.62)$$

Where ω_b is the base electrical angular frequency.

Since the base of the flux linkages, ψ_q and ψ_d , is the same as V_b for the stator voltage, Equation (4.54) of the torque in per unit reduces to:

$$T_{em(pu)} = \psi_{d(pu)} i_{q(pu)} - \psi_{q(pu)} i_{d(pu)} \tag{4.63}$$

Equation of the motion of the rotor assembly, expressed in per unit, is:

$$T_{em(pu)} + T_{mech(pu)} - T_{damp(pu)} = 2H \frac{d(\frac{\omega_r}{\omega_b})}{dt} = 2H \frac{d(\frac{\omega_r - \omega_e}{\omega_b})}{dt} \tag{4.64}$$

Where H is the inertia constant, which is defined as:

$$H = \frac{1}{2} J \frac{\omega_{bm}^2}{S_b} \tag{4.65}$$

4.3 Transient model

In large power systems, a number of synchronous generators are operated in parallel. Studies are routinely conducted to ensure that the generators will operate properly in occurrence of probable faults or changes in system conditions. Studies concerned with the dynamic behavior of synchronous generators are often divided into three classes: Transient stability studies, Dynamic stability studies and Long-term dynamic energy studies.

When dealing with a large power system, it is not practical to represent each and every component in full detail. The electromechanical oscillation frequency between synchronous generators in a power system typically lies between 0.5 to 3 Hz. The sub-transient time constant of most machines is between 0.03 to 0.04 seconds, which is short compared to the typical period of the electromechanical oscillations of machines. But the transient time constant ranging from 0.5 to 10 seconds is usually longer than the period of the electromechanical oscillations. In practice, the range of frequency response considered is important for the problem at hand, appropriate models of the required fidelity would be selected.

In addition to minimize the effort spent on setting up the models and on computation, we also take advantage of simplifications that are offered by the separation in time scales of the different dynamic systems behaviors, and the fact that the severity of a disturbance is usually attenuated as it propagates through the system. For example, the duration of the electrical transients of the network is very short relative to the electromechanical dynamics of the generator; as such, a static representation of the network can be used where longer electromechanical oscillations are primarily of interest. On the other hand, the duration of interest may not be long enough to require the inclusion of the slower-acting dynamics components, such as those of boilers and automatic generation control. However, where separate groups of machines are interconnected by weak ties and the stability of the generators cannot be determined from the first few swings, the dynamics of slower-acting components should be represented. For extended time studies, the choice of models would be a compromise between better fidelity offered by a full model and the savings in computational effort offered by simpler models.

Some understanding of the basic dynamic behavior of synchronous generators in transient situations can be obtained using a simple model of a generator and network. Fig. 4.7 shows such a simplified representation of a system in which the generator is represented by a

voltage behind the transient reactance model and the rest of the ac network, to which the generator is connected, is represented by a single-phase Thevenin's equivalent.

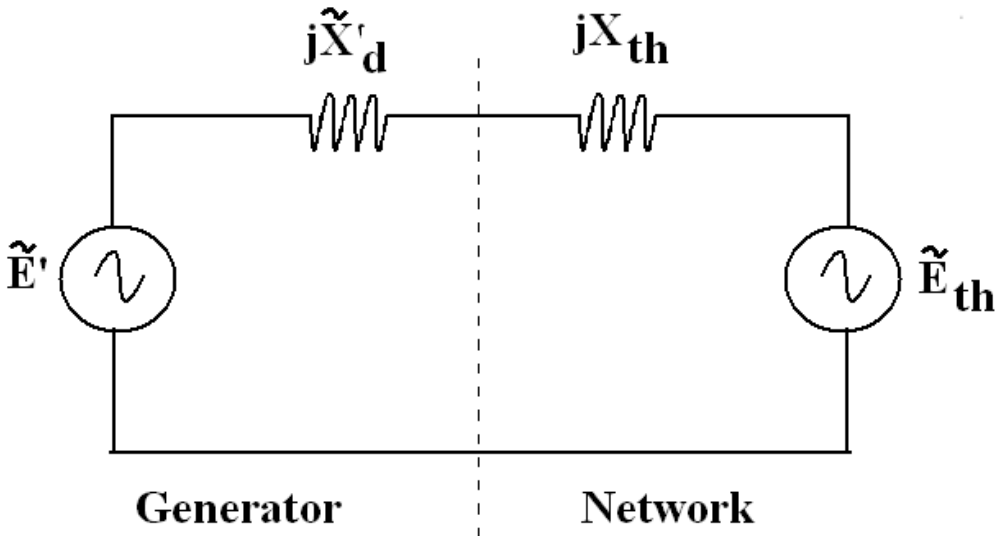


Fig. 4.7: Equivalent circuit representation of generator and network

Equation of the equivalent circuit representation of the system is:

$$\widetilde{E'} = \tilde{E}_{th} + jX_t\tilde{I} \tag{4.66}$$

$$X_t = X'_d + X_{th} \tag{4.67}$$

If the Thevenin's voltage is used as a reference pharos, that is:

$$\tilde{E}_{th} = E_{th}\angle 0 \tag{4.68}$$

$$\widetilde{E'} = E'\angle\delta \tag{4.69}$$

Then the electrical output power of the generator is given by:

$$P_{gen} = \frac{E\prime E_{th}}{X_t}\sin\delta \tag{4.70}$$

The above expression describes the power vs. angle relationship of the simple system. While the swing equation is described by the following expression:

$$\left(\frac{d\delta}{dt}\right)^2 = \frac{\omega_b}{H}\int\left(P_{mech} - P_{gen}\right)d\delta \tag{4.71}$$

Now we will show the equations of the transient model in the q and d -axis for the transient period where the damper windings may be assumed to be no longer active.

The following equations describe a simplified transient model without damper windings and with changes in stator qd flux linkages are neglected.

Considering that x'_d, x'_q are q- and d-axis transient generator reactance, L'_q, L'_d are q- and d-axis transient generator inductances, E'_q, E'_d are q- and d-axis transient electromotive forces, λ'_q, λ'_d are q- and d-axis transient flux linkages and T'_{do}, T'_{qo} are the transient open circuit time constants, then the stator winding equations are:

$$v_q = -r_s i_q - x'_d i_d + E'_q \tag{4.72}$$

$$v_d = -r_s i_d - x'_q i_q + E'_d \tag{4.73}$$

Rotor winding equations are:

$$T'_{do} \frac{dE'_q}{dt} + E'_q = E_f - (x_d - x'_d) i_d \tag{4.74}$$

$$T'_{qo} \frac{dE'_d}{dt} + E'_d = -E_g - \left(x_q - x'_q\right) i_q \tag{4.75}$$

Where:

$$\lambda'_q = \lambda_q - L'_q(-i_q) \tag{4.76}$$

$$\lambda'_d = \lambda_d - L'_d(-i_d) \tag{4.77}$$

$$E'_d = -\omega_b \lambda'_q \tag{4.78}$$

$$E'_q = -\omega_b \lambda'_d \tag{4.79}$$

The torque and motion equations are:

$$T_{em} = -\left\{E'_q i_q + E'_d i_d + \left(x'_q - x'_d\right) i_d i_q\right\} \tag{4.80}$$

$$2H \frac{d\left(\frac{\omega_r - \omega_e}{\omega_b}\right)}{dt} = T_{em} + T_{mech} - T_{damp} \tag{4.81}$$

Where:

$$\omega_r - \omega_e = \frac{d\delta_e}{dt} \tag{4.82}$$

$$\omega_r = \frac{p}{2} \omega_{rm} \tag{4.83}$$

4.4 Sub-transient model

Now we shall show the equations which describe sub-transient model that has *f* and *g* field windings, and *kd* and *kq* damper windings.

During the sub-transient period, the coupling effects of both field and damper windings are active in that the currents in these windings are free to change. When a transient disturbance occurs on the stator side and the stator currents change, the corresponding change in currents flowing in the field and damper windings on the rotor would initially maintain the flux linkages of these rotor windings as constant.

Following sets of equations represent a simplified sub-transient model with changes in stator qd flux linkages are neglected.

Stator winding equations:

$$v_q = -ri_q + E_q^{''} - \omega_b L_d^{''} i_d \tag{4.84}$$

$$v_d = -ri_d + E_d^{''} + \omega_b L_q^{''} i_q \tag{4.85}$$

Where:

$$E_q^{''} = \left(\frac{L_d^{''}-L_{ls}}{L_d'-L_{ls}}\right) E_q' + \left(\frac{L_d'-L_d^{''}}{L_d'-L_{ls}}\right) \omega_e \lambda_{kd}' \tag{4.86}$$

$$E_d^{''} = \left(\frac{L_q^{''}-L_{ls}}{L_q'-L_{ls}}\right) E_d' + \left(\frac{L_q'-L_q^{''}}{L_q'-L_{ls}}\right) \omega_e \lambda_{kq}' \tag{4.87}$$

Rotor winding equations:

$$T_{do}' \frac{dE_q'}{dt} = E_f - \omega_b \left\{\frac{(L_d-L_d')(L_d^{''}-L_{ls})}{L_d'-L_{ls}}\right\} i_d$$

$$- \left\{1 + \frac{(L_d'-L_d^{''})(L_d-L_d')}{(L_d'-L_{ls})^2}\right\} E_q' + \left\{\frac{(L_d-L_d')(L_d'-L_d^{''})}{(L_d'-L_{ls})^2}\right\} \omega_e \lambda_{kd}' \tag{4.88}$$

$$T_{qo}' \frac{dE_d'}{dt} = -E_g + \omega_b \left\{\frac{(L_q-L_q')(L_q^{''}-L_{ls})}{L_q'-L_{ls}}\right\} i_q$$

$$- \left\{1 + \frac{(L_q'-L_q^{''})(L_q-L_q')}{(L_q'-L_{ls})^2}\right\} E_d' - \left\{\frac{(L_q-L_q')(L_q'-L_q^{''})}{(L_q'-L_{ls})^2}\right\} \omega_e \lambda_{kq}' \tag{4.89}$$

$$T_{do}^{"}\frac{d\lambda'_{kd}}{dt} = \frac{E'_q}{\omega_e} - \lambda'_{kd} - (L'_d - L_{ls})i_d \tag{4.90}$$

$$T_{qo}^{"}\frac{d\lambda'_{kq}}{dt} = \frac{-E'_d}{\omega_e} - \lambda'_{kq} - \left(L'_q - L_{ls}\right)i_q \tag{4.91}$$

Torque and motion equations are:

$$T_{em} = -\left\{\left(\frac{E_q^{"}i_q + E_d^{"}i_d}{\omega_e/\omega_b}\right) + \omega_b\left(L_q^{"} - L_d^{"}\right)i_d i_q\right\} \tag{4.92}$$

$$2H\frac{d\left(\frac{\omega_r - \omega_e}{\omega_b}\right)}{dt} = T_{em} + T_{mech} - T_{damp} \tag{4.93}$$

Where:

$$\omega_r - \omega_e = \frac{d\delta_e}{dt} \tag{4.94}$$

$$\omega_r = \frac{p}{2}\omega_{rm} \tag{4.95}$$

Where symbols with double apostrophe refer to sub-transient quantities.

5 HYDRAULIC TURBINES

5.1 Introduction

In a hydraulic power generation plant, the stored energy in water as a hydraulic fluid is converted into mechanical energy by means of hydraulic turbine. Hydraulic turbines are of two basic types: impulse turbines and reaction turbines. Selection of the type of the turbine depends upon the head and water flow rate of the dam. The shaft of the generating unit may be in a vertical, horizontal, or inclined direction depending on conditions of the plant and the type of turbine applied. The majority of new installations are vertical.

Fig. 5.1 illustrates the main components of a typical hydroelectric generating unit. From the reservoir, water is drawn from an area called the forebay to the turbine through the water column. The water column comprises all of the structures used to convey water from the forebay to the turbine. It may include an intake structure, a penstock, one or more surge tanks, and a spiral case. The composite water column inertias and elasticity of these structures contribute to the water hammer effect that impacts the performance of the turbine governing system.

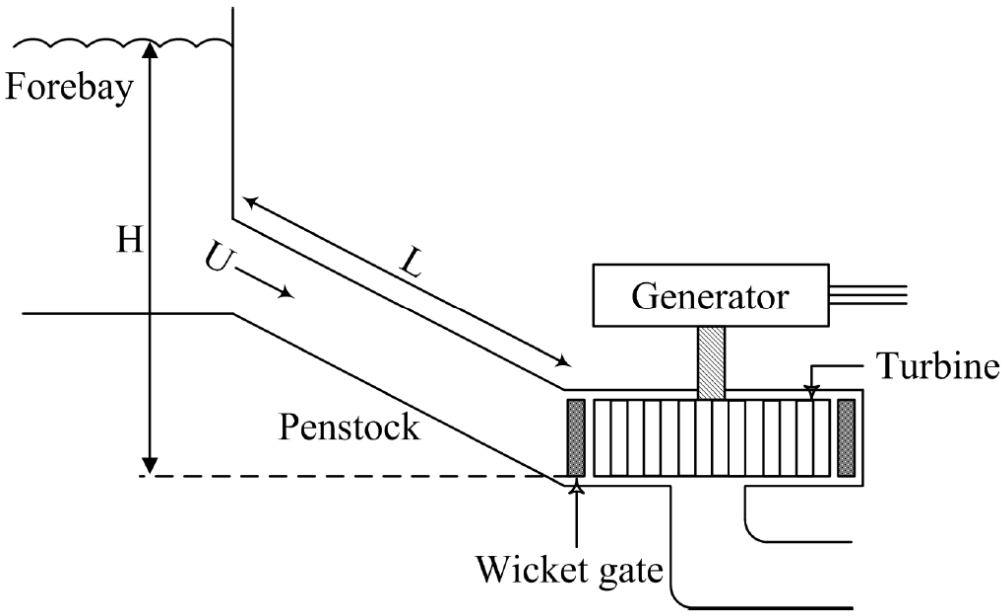


Fig. 5.1: Simplified schematic of a hydroelectric power plant

If water flows from a high level to a low level through a hydraulic turbine the potential energy of the water stored in a high reservoir is converted into mechanical work on the turbine shaft. The turbine may be either an impulse turbine or a reaction turbine and, although the way in which they operate hydraulically differs, the work done by both types is entirely due to the conversion of kinetic energy. In impulse turbines, such as the Pelton wheel, all the available energy in the water is converted into kinetic energy as the water passes through the nozzle. The water forms a free jet as it leaves the nozzle and strikes the runner where the kinetic energy is converted into mechanical work. In the reaction turbine, such as the Francis turbine, only a part of the energy in the water is converted into kinetic energy as the water passes through the adjustable wicket gates, with the remaining conversion taking place inside

the runner itself. All passages are filled with water including the draft tube. In both turbines the power is controlled by regulating the flow into the turbine by wicket gates on the reaction turbine and by a needle, or spear, on the impulse turbine. What is required is a mathematical description of how the turbine power changes as the position of the regulating device is changed.

This chapter examines the characteristics of prime movers and energy supply systems and develops appropriate models suitable for their representation in power system dynamic studies. The principles and implementation of automatic generation control are produced. The focus here is on those characteristics of the power plants that impact on the overall performance of the power system, and not on a detailed study of the associated processes.

5.2 Hydraulic turbines systems

Hydraulic turbines are of two basic types: impulse turbines and reaction turbines, which are briefly described here; it may refer to specialized mechanical references for a more detailed description.

The impulse-type turbine (also known as Pelton wheel) is used for high heads - 300 meters or more. The runner is at atmospheric pressure, and the whole of the pressure drop takes place in stationary nozzles that convert potential energy to kinetic energy. The high-velocity jets of water impinge on spoon-shaped buckets on the runner, which deflect the water axially through about 160°; the change in momentum provides the torque to drive the runner, the energy supplied being entirely kinetic.

In a reaction turbine the pressure within the turbine is above atmospheric; the energy is supplied by the water in both kinetic and potential (pressure head) forms. The water first passes from a spiral casing through stationary radial guide vanes and gates around its entire periphery. The gates control water flow. There are two subcategories of reaction turbines: Francis and propeller.

The Francis turbine is used for heads up to 360 meters. In this type of turbine, water flows through guide vanes impacting on the runner tangentially and exiting axially.

The propeller turbine, as the name implies, uses propeller-type wheels. It is for use on low heads - up to 45 meters. Either fixed blades or variable-pitch blades may be used. The variable-pitch blade propeller turbine, commonly known as the Kaplan wheel, has high efficiency at all loads.

The performance of a hydraulic turbine is influenced by the characteristics of the water column feeding the turbine; these include the effects of water inertia, water compressibility, and pipe wall elasticity in the penstock. The effect of water inertia is to cause changes in turbine flow to lag behind changes in turbine gate opening. The effect of elasticity is to cause travelling waves of pressure and flow in the pipe; this phenomenon is commonly referred to as “water hammer”.

Fig. 5.2 shows types of hydraulic turbine runners used in hydro power plants.

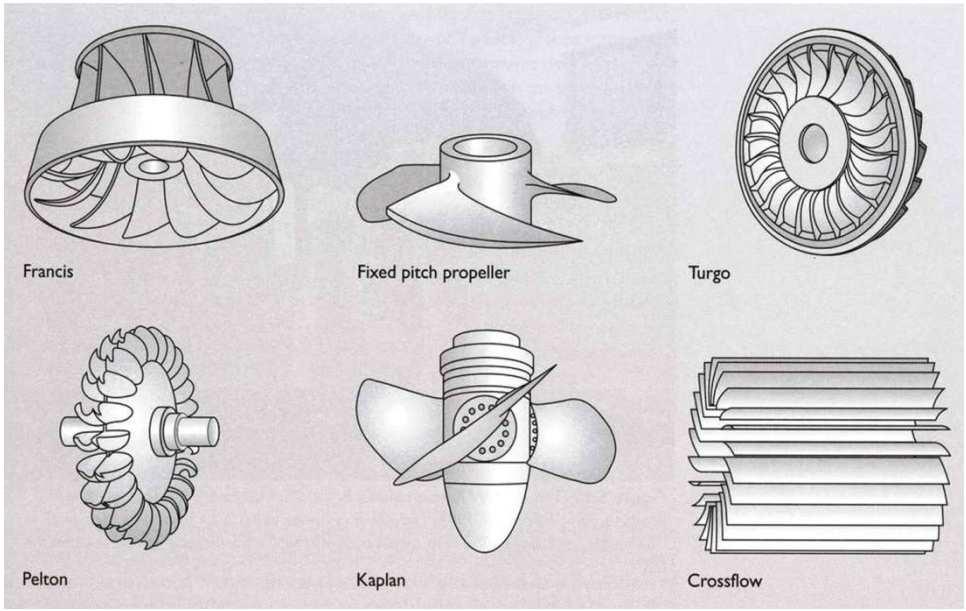


Fig. 5.2: Types of turbine runner

Precise modeling of hydraulic turbines requires inclusion of transmission-line- like reflections which occur in the elastic-walled pipe carrying compressible fluid. Typically, the speed of propagation of such travelling waves is about 1200 meters per second. Therefore, travelling wave models may be required only if penstocks are long.

5.3 Hydraulic turbine models

5.3.1 Non-linear model

Fig. 5.3 shows relationship of hydro prime mover system and controls to complete system.

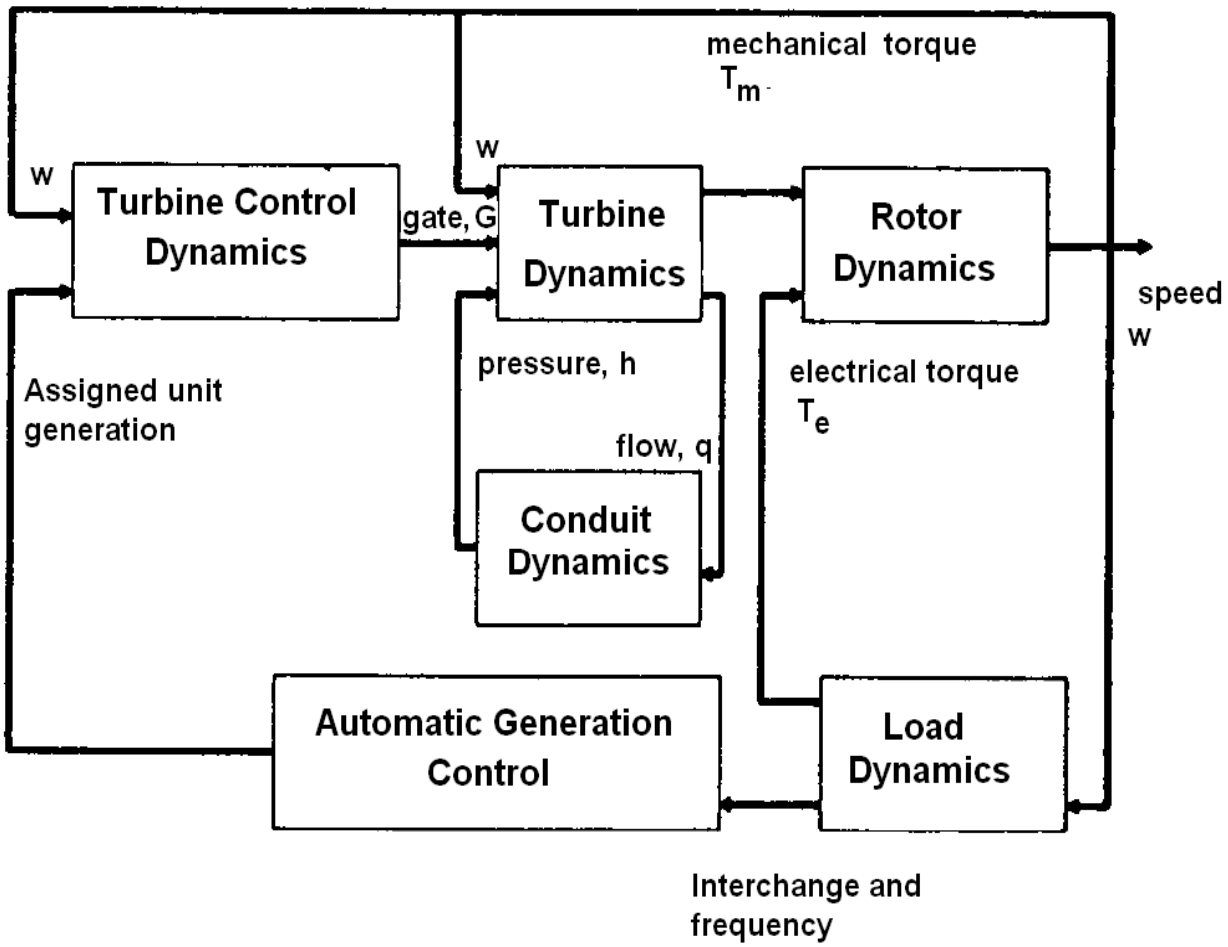


Fig. 5.3: Functional Block Diagram of power generation and control system

The penstock is modeled by assuming that the flow is incompressible when the rate of change of flow in the penstock is obtained by equating the rate of change of momentum of the water in the penstock to the net force on the water in the penstock when:

$$\rho L \frac{dQ}{dt} = F_{net} \tag{5.1}$$

Where Q is the volumetric flow rate, L the penstock length and ρ the mass density of water.

The net force on the water can be obtained by considering the pressure head at the conduit. On entry to the penstock the force on the water is simply proportional to the static head H_s, while at the wicket gate it is proportional to the head H across the turbine. Due to friction effects in the conduit, there is also a friction force on the water represented by the head loss H_l so that the net force on the water in the penstock is:

$$F_{net} = (H_s - H_l - H)A.\rho.g \tag{5.2}$$

Where A is the penstock cross-sectional area and g is the acceleration due to gravity. Substituting the net force into Equation (5.2) gives:

$$\rho L \frac{dQ}{dt} = (H_s - H_l - H)A.\rho.g \tag{5.3}$$

It is usual to normalize this equation to a convenient base. Although this base system is arbitrary, the base head h_{base} is taken as the static head above the turbine, in this case H_s, while the base flowrate q_{base} is taken as the flowrate through the turbine with the gates fully open and the head at the turbine equal to h_{base} (IEEE Committee Report, 1992). Dividing both sides of Equation (5.3) by $h_{base}.q_{base}$ gives:

$$\frac{dq}{dt} = \frac{1}{T_w}(1 - h_l - h) \tag{5.4}$$

Where

$$q = \frac{Q}{q_{base}} \tag{5.5}$$

$$h = \frac{H}{h_{base}} \tag{5.6}$$

q, h are the normalized flowrates and pressure heads respectively.

$$T_w = \frac{Lq_{base}}{A.g.h_{base}} \tag{5.7}$$

T_w is the water starting time, which is theoretically defined as the time taken for the flowrate in the penstock to change by a value equal to q_{base} when the head term in brackets changes by a value equal to h_{base}. The head loss h_l is proportional to the flowrate squared and depends on the conduit dimensions and friction factor. It suffices here to assume that $h_l = k_f q^2$ and can often be neglected. This equation defines the penstock model.

In modeling the turbine itself both its hydraulic characteristics and mechanical power output must be modeled. Firstly, the pressure head across the turbine is related to the flowrate by assuming that the turbine can be represented by the valve characteristic:

$$Q = kG\sqrt{H} \tag{5.8}$$

Where G is the gate position between 0 and 1, k is a constant. With the gate fully open G=1 and this equation can be normalized by dividing both sides by:

$$q_{base} = k\sqrt{h_{base}} \tag{5.9}$$

Then the per unit form is:

$$q = G\sqrt{H} \tag{5.10}$$

Secondly, the power developed by the turbine is proportional to the product of the flowrate and the head and depends on the efficiency. To account for the turbine not being 100% efficient the no-load flow q_{nl} is subtracted from the actual flow to give, in normalized parameters:

$$P_m = h(q - q_{nl}) \tag{5.11}$$

Unfortunately this expression is in a different per-unit system to that used for the generator whose parameters are normalized to the generator MVA base so that last equation is written as:

$$P_m = A_t h(q - q_{nl}) \tag{5.12}$$

Where the factor A_t is introduced to account for the difference in the bases. The value of the factor A_t can be obtained by considering the operation of the turbine at rated load when:

$$P_m = A_t h_r(q_r - q_{nl}) = \frac{turbine\ power(MW)}{generator\ MVA\ rating} \tag{5.13}$$

and the suffix 'r' indicates the value of the parameters at rated load. Rearranging the equation gives:

$$A_t = \frac{turbine\ power(MW)}{generator\ MVA\ rating} \frac{1}{h_r(q_r - q_{nl})} \tag{5.14}$$

A damping effect is also present that is dependent on gate opening so that at any load condition the turbine power can be expressed by:

$$P_m = A_t h(q - q_{nl}) - DG\Delta\omega \tag{5.15}$$

Where D is the damping coefficient.

Fig. 5.4 shows the turbine non-linear model.

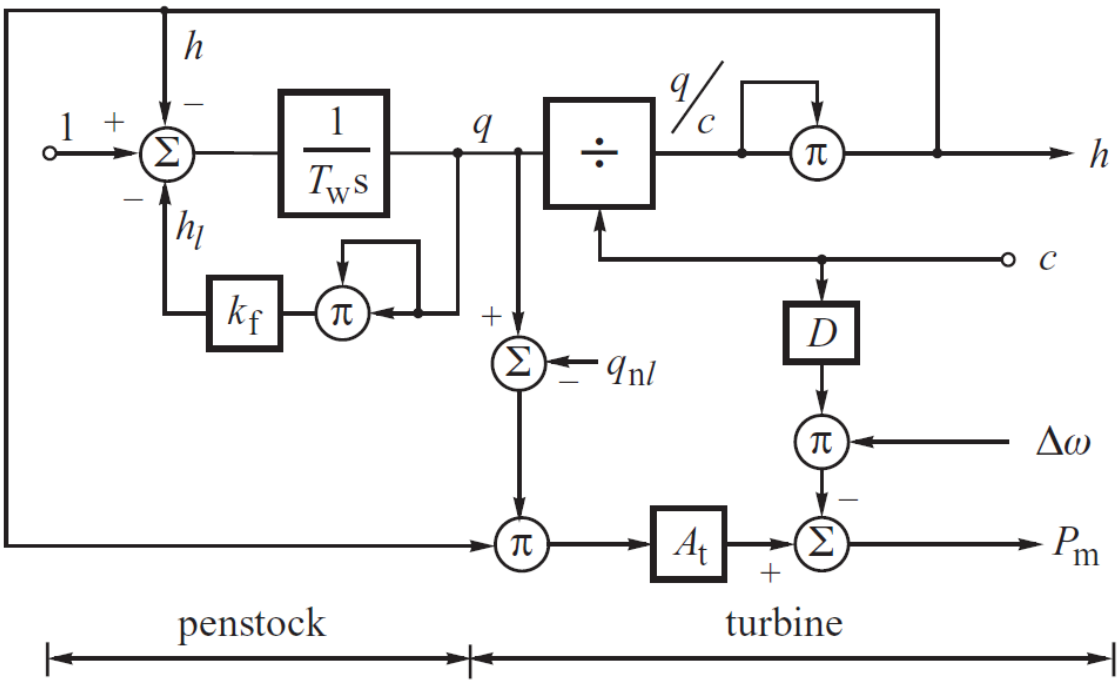


Fig 5.4: Hydraulic turbine non-linear model

5.3.2 Linearized model

The classical model of the water turbine (IEEE Committee Report, 1973a, 1973b, 1973c) uses a linearized version of the nonlinear model. Such a model is valid for small changes of mechanical power and can be obtained by linearizing non-linear model Equations about an initial operating point to give following equations:

$$\frac{d\Delta q}{dt} = -\frac{\Delta h}{T_w} \tag{5.16}$$

$$\Delta q = \frac{\partial q}{\partial G}\Delta G + \frac{\partial q}{\partial h}\Delta h \tag{5.17}$$

$$\Delta P_m = \frac{\partial P_m}{\partial h}\Delta h + \frac{\partial P_m}{\partial q}\Delta q \tag{5.18}$$

Introducing the Laplace operator " s " and eliminating Δh and Δq from the equations gives:

$$\frac{\Delta P_m}{\Delta G} = \frac{\left[\frac{\partial q}{\partial G}\frac{\partial P_m}{\partial q} - sT_w\frac{\partial P_m}{\partial h}\frac{\partial q}{\partial G}\right]}{1+sT_w\frac{\partial q}{\partial h}} \tag{5.19}$$

Where the partial derivatives are:

$$\frac{\partial q}{\partial h} = \frac{1}{2}\frac{G_0}{\sqrt{h_0}} \tag{5.20}$$

$$\frac{\partial q}{\partial G} = \sqrt{h_0} \tag{5.21}$$

$$\frac{\partial P_m}{\partial q} = A_t h_0 \tag{5.22}$$

$$\frac{\partial P_m}{\partial h} = A_t(q_0 - q_{nl}) \approx A_t(q_0) \tag{5.23}$$

The suffix 0 indicates an initial value. Substituting into Equation (5.19) and noting that:

$$q_0 = G_0\sqrt{h_0} \tag{5.24}$$

Gives:

$$\frac{\Delta P_m}{\Delta G} = A_t h_0^{3/2}\frac{1-sT_w'}{1+s\frac{T_w'}{2}} \tag{5.25}$$

Where:

$$T_w' = T_w\frac{q_0}{h_0} = \frac{L}{Ag}\frac{Q_0}{H_0} \tag{5.26}$$

Typically T'_w is between 0.5 and 5 s.

Fig. 5.5 and Fig. 5.6 show hydraulic turbine linear model and the response of the linear turbine model to a step change in gate position.

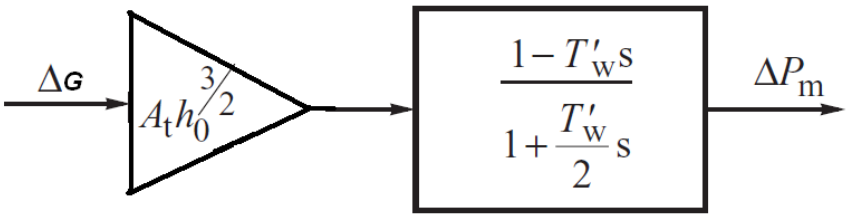


Fig.5.5: Hydraulic Turbine Linear Model

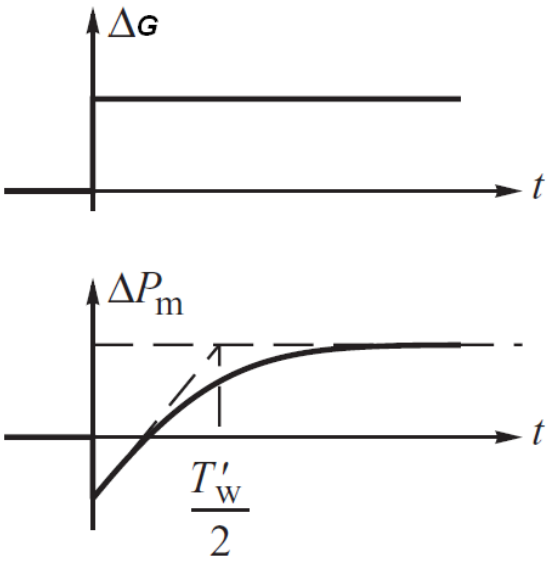


Fig.5.6: Response Of The Linear Turbine Model to a Step Change In Gate Position

This is the classic definition of water starting time but is dependent on the values of the head and flowrate at the linearization point. It therefore varies with load. If required, the constant A_t can be absorbed into the gate position when it effectively converts the gate opening to per unit turbine power on the generator MVA base.

Equation (5.25) describes an interesting and important characteristic of water turbines. For example, suppose that the position of the gate is suddenly closed slightly so as to reduce the turbine power output. The flowrate in the penstock cannot change instantaneously so the velocity of the flow through the turbine will initially increase. This increase in water velocity will produce an initial increase in the turbine power until, after a short delay, the flowrate in the penstock has time to reduce when the power will also reduce. This effect is reflected in Equation (5.25) by the minus sign in the numerator. This characteristic is shown in Fig. 5.6 where a step increase in the gate position ΔG initially produces a rapid drop in power output. As the flowrate in the penstock increases, the power output increases.

Although the linearized model shown by Fig. 5.5 has been successfully used in both steady-state and transient stability studies, IEEE Committee Report (1992) recommends the use of the nonlinear turbine model in power system studies because its implementation using computers is no more difficult than the approximate linear transfer function.

In IEEE Committee Report (1992) was presented linearized model for non-elastic water column which is described by the following equation:

$$\Delta P_m = A_t \frac{(1-T_1 s)\Delta G}{(1+T_2 s)} - D G_0 \Delta\omega \tag{5.27}$$

Where:

$$T_1 = (q_0 - q_{nl}) T_w \tag{5.28}$$

$$T_2 = G_0 \frac{T_w}{2} \tag{5.29}$$

Fig. 5.7 shows the block diagram of this model.

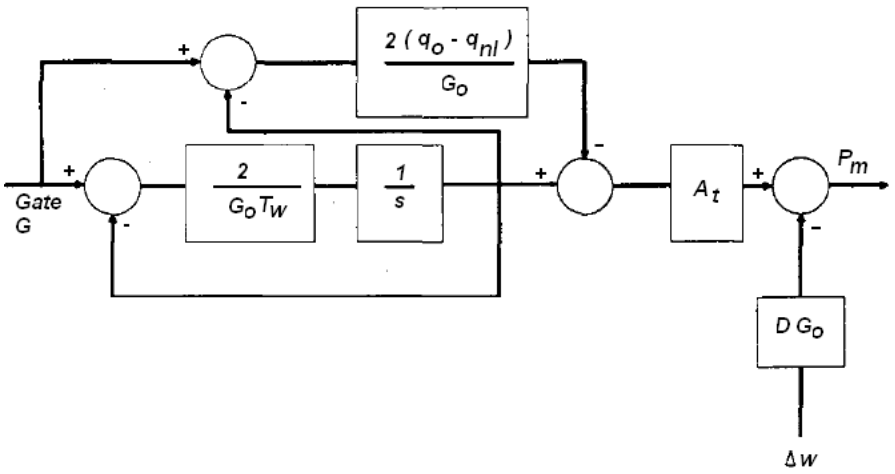


Fig. 5.7: Linearized model of turbine, with a non-elastic water coloumn

5.4 Governing systems of hydraulic turbine

The basic function of a governor is to control speed and/or load. In this section we discuss the special requirements of governing hydraulic turbines, their physical realization and modeling in system studies.

The primary speed/load control function involves feeding back speed error to control the gate position. In order to ensure satisfactory and stable parallel operation of multiple units, the speed governor is provided with a droop characteristic. The purpose of the droop is to ensure equitable load sharing between generating units. Typically, the steady-state droop is set at about 5%, such that a speed deviation of 5% causes 100% change in gate position or power output; this corresponds to a gain of 20. For a hydro turbine, however, such a governor with a simple steady-state droop characteristic would be unsatisfactory.

Hydro turbines have a peculiar response due to water inertia: a change in gate position produces an initial turbine power change which is opposite to that sought. For stable control performance, a large transient (temporary) droop with a long resetting time is therefore required. This is accomplished by the provision of a rate feedback or transient gain reduction compensation as shown in Fig. 5.8 The rate feedback retards or limits the gate movement until the water flow and power output have time to catch up. The result is a governor which exhibits a high droop (low gain) for fast speed deviations, and the normal low droop (high gain) in the steady state. Example 9.3 illustrates the effect of the transient droop compensation on the stability characteristics of the governing system.

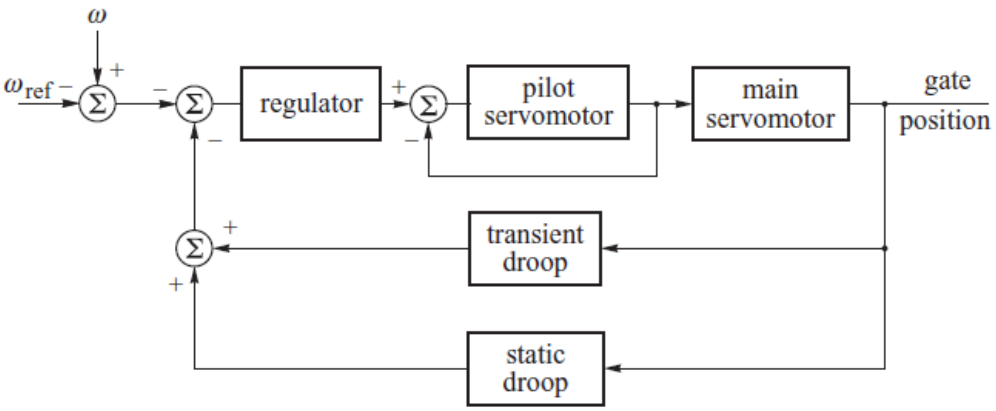


Fig. 5.8: Block diagram of hydraulic turbine governing system

On older units the governing function is realized using mechanical and hydraulic components. Fig. 5.9 shows a simplified schematic of a mechanical hydraulic governor. Speed sensing, permanent droop feedback, and computing functions are achieved through mechanical components; functions involving higher power are achieved through hydraulic components. A dashpot is used to provide transient droop compensation. A bypass arrangement is usually provided to disable the dashpot if so desired.

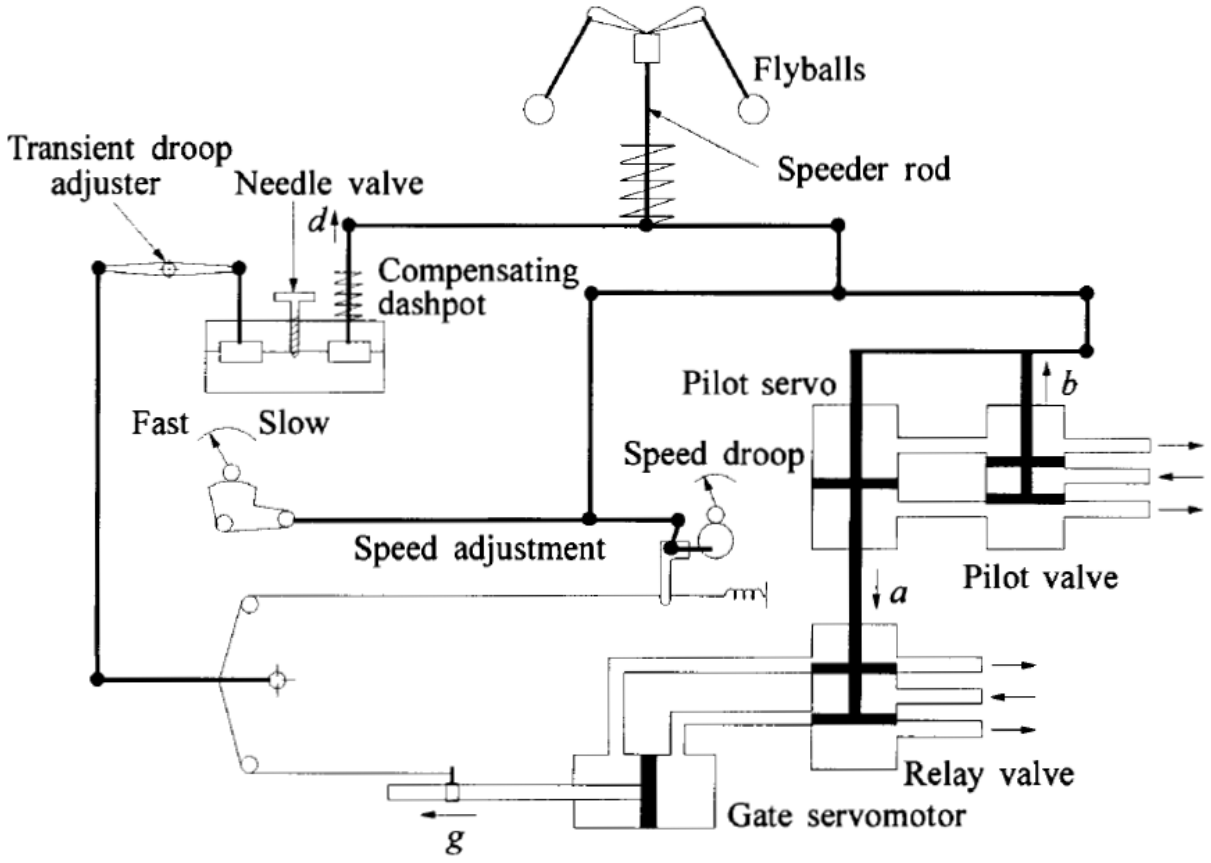


Fig. 5.9: Schematic of mechanical hydraulic governor for hydro turbine

Newer speed governors for hydraulic turbines use electrohydraulic systems. Functionally, their operation is very similar to that of mechanical-hydraulic governors. Speed sensing, permanent droop, temporary droop, and other measuring and computing functions are performed electrically. The electric components provide greater flexibility and improved performance with regard to dead bands and time lags. The dynamic characteristics of electric governors are usually adjusted to be essentially similar to those of the mechanical-hydraulic governors.

Modern governors are provided with PID controllers, which are three-term controllers with proportional-integral-derivative (PID) action, as shown in Fig. 5.10. These allow the possibility of higher response speeds by providing both transient gain reduction and transient gain increase. The derivative action is beneficial for isolated operation, particularly for plants with large water starting time (T_w = 3s or more). Typical values are K_P=3.0, K_I=O.7, and K_D=O.5. However, the use of a high derivative gain or transient gain increase will result in excessive oscillations and possibly instability when the generating Limit is strongly connected to an interconnected system. Therefore, the derivative gain is usually set to zero; without the derivative action, the transfer function of a PID (now PI) governor is equivalent to that of the mechanical-hydraulic governor. The proportional and integral gains may be selected to result in the desired temporary droop and reset time.

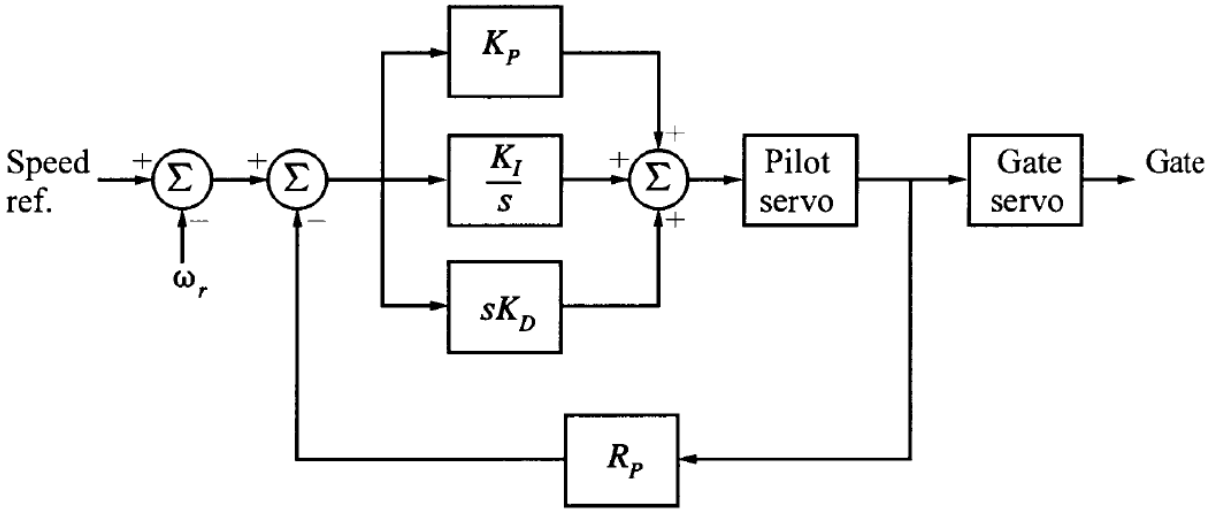


Fig. 5.10: PID Governor

Digital PI (or PID) controlled governor can be designed to offer:

- Speed control of the turbine;
- Operate sensitively and respond to errors 7 0.15 Hz to restore the normal condition.
- Parallel operation in multi-machine system.
- A minimum dead band.
- Good dynamic response on load throw-off and during static frequency condition.
- Steep droop characteristics.
- Control of load based on load reference and line frequency using feed back control loops.
- A self-regulating feature to stab

Also these are multi-channeled so as to operate in integrated control of the turbine generating set. It uses optimal gain values for each point of desired operation stored in memory. The controller then updates the gain in the control algorithm to reflect changes in the operating point. Computer programs are used to generate and implement variable gains. The need for improvement in reliability and redundancy led to development of a duplicate microprocessor based governor. Because of the fast start-up and synchronization, the no-load operation and the no-load flow loss decrease.

6 EXCITATION SYSTEM

The generator excitation system consists of an exciter and an AVR , it is necessary to supply the generator with DC field current. The power rating of the exciter is usually in the range 0.2–0.8% of the generator's megawatt rating. In the case of a large generator this power is quite high, in the range of several megawatts. The voltage rating of the exciter will not normally exceed 1000 V as any higher voltage would require additional insulation of the field winding.

6.1 Automatic voltage regulators AVRs

The AVR regulates the generator terminal voltage by controlling the amount of current supplied to the generator field winding by the exciter. The general block diagram of the AVR subsystem is shown in Fig. 6.1. The measuring element senses the current, power, terminal voltage and frequency of the generator. The measured generator terminal voltage V_g is compensated for the load current I_g and compared with the desired reference voltage V_{ref} to produce the voltage error ΔV. This error is then amplified and used to alter the exciter output, and consequently the generator field current, so that the voltage error is eliminated. This represents a typical closed-loop control system. The regulation process is stabilized using a negative feedback loop taken directly from either the amplifier or the exciter.

Fig. 6.1shows block diagram of excitation system with Automatic Voltage Regulator AVR and Power System Stabilizer PSS.

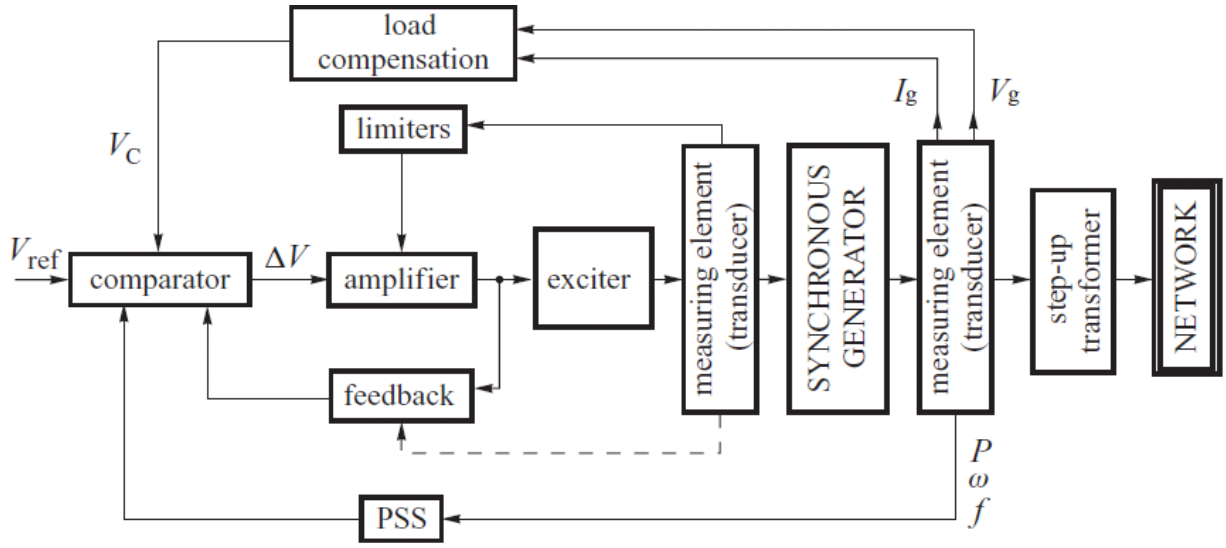


Fig. 6.1: Block Diagram of the Excitation System with AVR and PSS.

The load compensation element, together with the comparator, is shown in Fig. 6.2. The voltage drop across the compensation impedance $Z_C = R_C + jX_C$ due to the generator current I_g is added to the generator voltage V_g to produce the compensated voltage V_C according to the equation:

$$V_C = V_g + (R_C + jX_C)I_g \tag{6.1}$$

If the load compensation is not employed, then $V_C = V_g$ and the AVR subsystem input maintains constant and equal to generator terminal voltage. The use of load compensation effectively means that the point at which constant voltage is maintained is ‚pushed' into the network by a distance that is electrically equal to the compensation impedance. The assumed direction of the phasors in Fig. 6.2 means that moving the voltage regulation point towards the grid corresponds to negative compensation impedance.

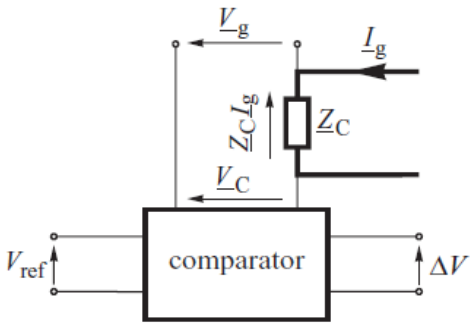


Fig. 6.2: Load compensation element together with the comparator.

The AVR subsystem also includes a number of limiters whose function is to protect the AVR, exciter and generator from excessive voltages and currents. They do this by maintaining the AVR signals between preset limits. Thus the amplifier is protected against excessively high input signals, the exciter and the generator against too high field currents, and the generator against too high armature currents and too high power angle. The last three limiters have built-in time delays to reflect the thermal time constant associated with the temperature rise in the winding.

A power system stabilizer (PSS) is sometimes added to the AVR subsystem to help damp power swings in the system.

6.2 Exciter systems

Generally exciters can be classified as either rotating or static. Fig. 6.3 shows some typical systems.

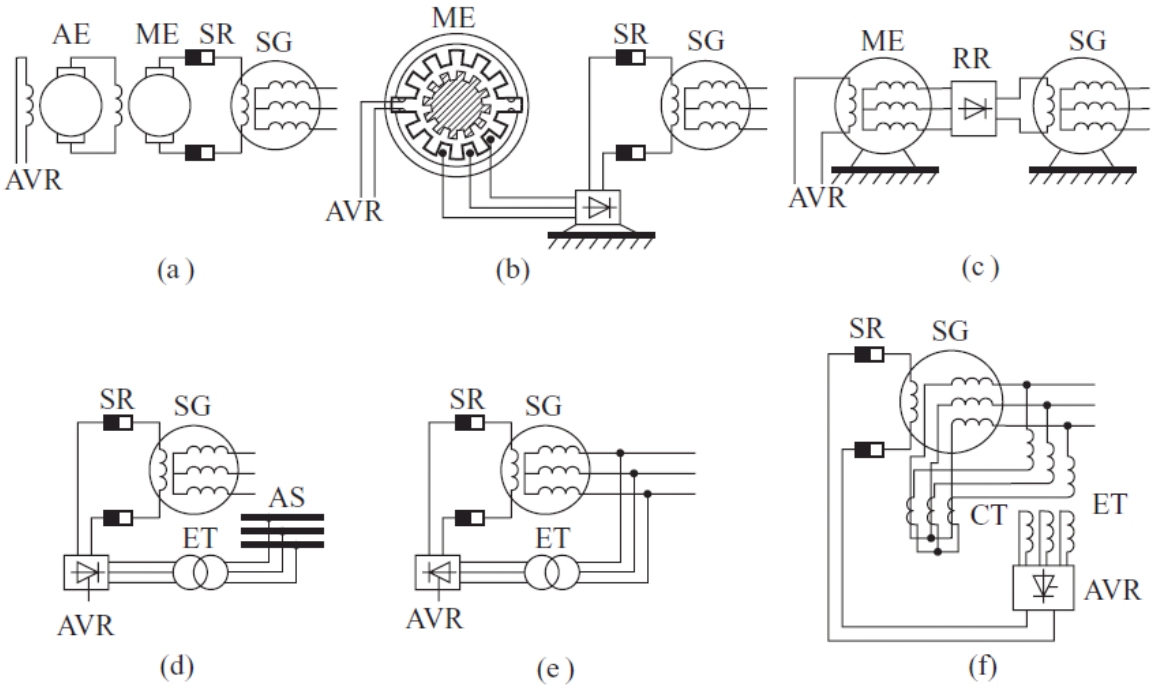


Fig. 6.3: Typical exciter systems.

The exciter systems shown by Fig. 6.3 are:

(a) cascaded DC generators.

(b) reluctance machine with rectifier.

(c) inside-out synchronous generator with rotating rectifier.

(d) controlled rectifier fed from the auxiliary supply.

(e) controlled rectifier fed from the generator terminals.

(f) controlled rectifier fed by the generator's voltage and current.

Typical models of the excitation systems in use today are DC exciters, AC exciters and static exciters.

6.2.1 DC exciters

Two different DC exciters are shown in Fig. 6.4, the first being separately excited and the second self-excited. A change in the exciter field current i_{exf} can be described by the following equation:

$$v_R = R_{exf} i_{exf} + L_{exf} \frac{di_{exf}}{dt} \tag{6.2}$$

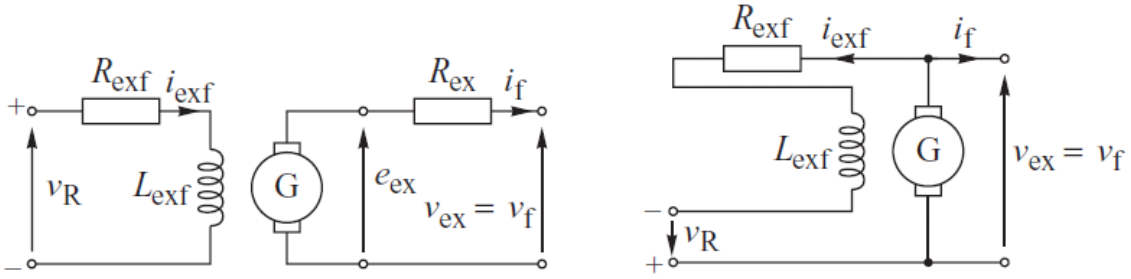


Fig. 6.4: Separately excited and self excited equivalent circuit diagram of DC exciters

The block diagramof the main part of the excitation system can be formulated by combining the block diagram of the exciter with that of the regulator and the stabilizing feedback signal as shown in Fig. 6.5. The regulator is represented by a first-order transfer function with a time constant T_A and gain K_A. Typical values of these parameters are T_A= 0.05–0.2 s and K_A= 20–400. The high regulator gain is necessary to ensure small voltage regulation of the order of 0.5%.

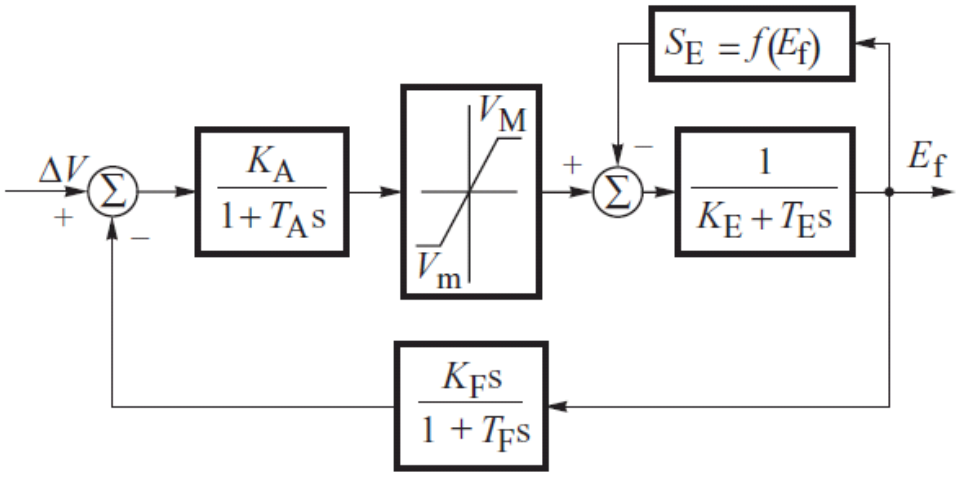


Fig. 6.5: Block dagram of the excitation system with DC exciter

Unfortunately, although this high gain ensures low steady-state error, when coupled with the length of the time constants the transient performance of the exciter is unsatisfactory. To achieve acceptable transient performance the system must be stabilized in some way that reduces the transient (high-frequency) gain. This is achieved by a feedback stabilization signal represented by the first-order differentiating element with gain K_F and time constant T_F. Typical values of the parameters in this element are T_F = 0.35 – 1s and K_F= 0.01 – 0.1. Although the saturation function $S_E = f(E_f)$ can be approximated by any nonlinear function, an exponential function of the following form is commonly used.

$$S_E = A_{ex} e^{B_{ex} E_f} \tag{6.3}$$

As this functionmust model the saturation characteristic over a wide range of exciter operating conditions, the parameters A_{ex} and B_{ex} of the exponential function are determined by considering the heavily saturated region of the characteristic corresponding to high excitation voltage and high exciter field current.

6.2.2 AC rotating exciters

These exciters usually use a three-phase bridge rectifier consisting of six diode modules as shown in Fig. 6.6. The rectifier is fed from a three-phase AC voltage source of emf V_E and reactance X_E. As with any rectifier system, the output voltage depends on the commutation characteristics of the rectifier as determined by the degree of commutation overlap. As the effect of commutation overlap depends on the current through the rectifiers and the commutating reactance X_E. The bridg characteristic and the Block diagram are illusturated by Fig. 6.7.

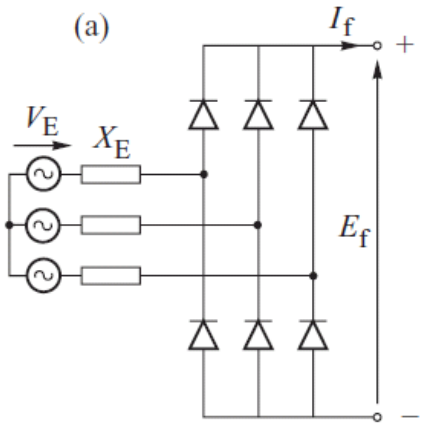


Fig. 6.6: Three phase uncontrolled bridge rectifier

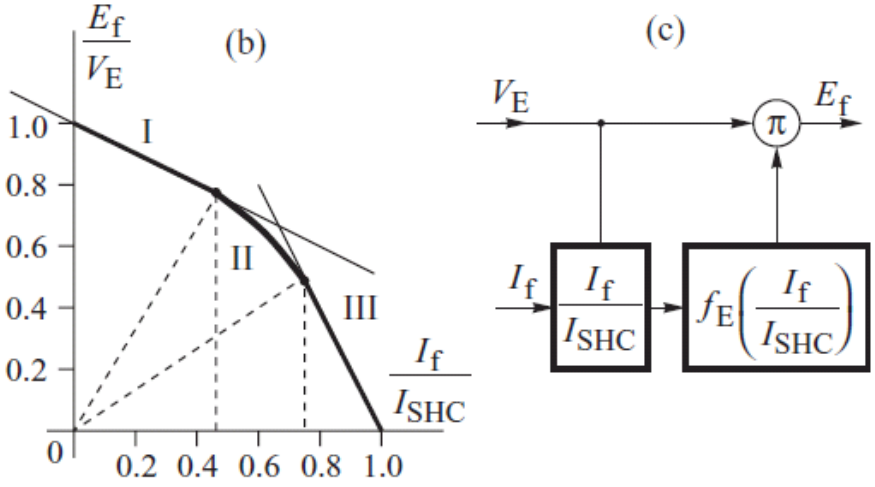


Fig. 6.7: Voltage-Current characteristic of rectifier with rectifier block diagram

The alternator rectifier is modelled by an integrating element with three feedback loops as shown in Fig. 6.8. The feedback loops with gain K_E and S_E play the same role as in the DC excitation system. However, the effect of the alternator resistance is now included in the voltage–current characteristics of the rectifier when S_E is determined from the no-load saturation characteristic rather than the load saturation line as used for the DC exciter. As the armature current in the AC exciter is proportional to the current in the main generator field winding, the third feedback loop, with gain K_D, uses this current to model the demagnetizing

effect of the armature reaction in the AC exciter. As the output voltage of a diode rectifier cannot drop below zero, this is modelled by the negative feedback loop containing the signal limiter. If the excitation voltage drops below zero a large negative signal is fed back to the summing point that prevents a further drop in the voltage, thereby keeping it at zero.

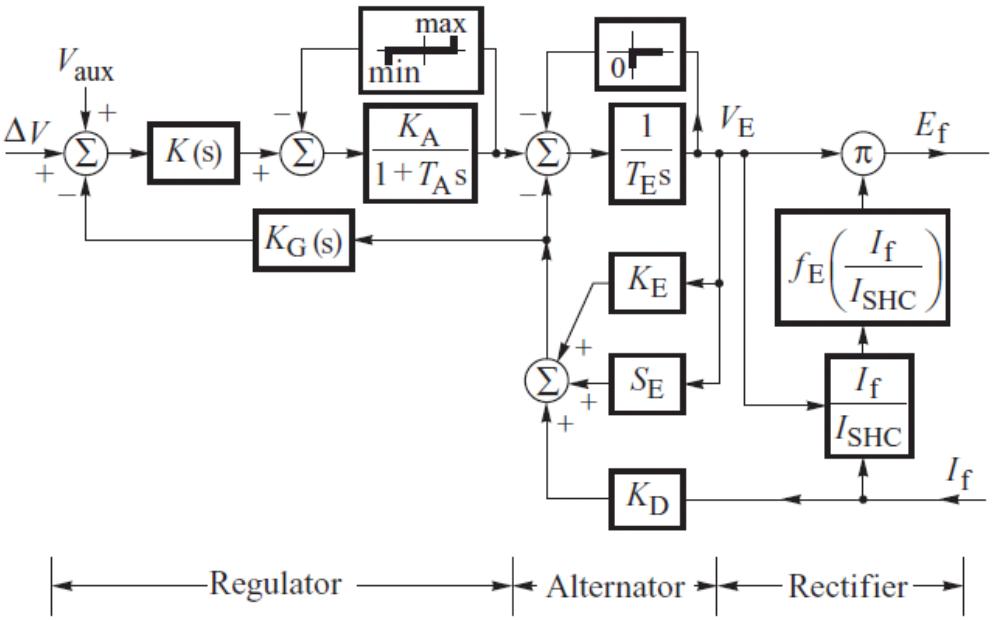


Fig. 6.8: Block diagram of exctation system with AC alternator and uncontrolled rectifier

The system is stabilized by the feedback loop with transfer function:

$$K_G(s) = \frac{K_F s}{1+T_F s} \tag{6.4}$$

In this case the stabilizing loop is supplied by a signal that is proportional to the exciter field current. Alternatively the system could be stabilized by supplying this block directly from the output of the voltage regulator or from the excitation voltage Ef. In this diagram the feedback stabilization is supplemented by an additional block with the transfer function K(s) in the forward path preceding the regulator block. Both $K_G(s)$ and K(s) depend on the specific excitation system and can be implemented by either analogue or digital techniques. Normally K(s) will have a PI or PID type of structure and is often represented by the transfer function:

$$K(s) = \frac{1+T_C s}{1+T_B s} \tag{6.5}$$

A major simplification to the model can be made by neglecting the variable effect of the field current on the rectifier voltage.

6.2.3 Static exciters

In static excitation systems the source of the direct current is a controlled three-phase bridge rectifier consisting of six thyristor modules as shown in Fig. 6.9.

The output characteristic of the rectifier depends on both the thyristor firing angle α and the system commutation characteristic. The rectifier characteristics shown in Fig. 6.10.

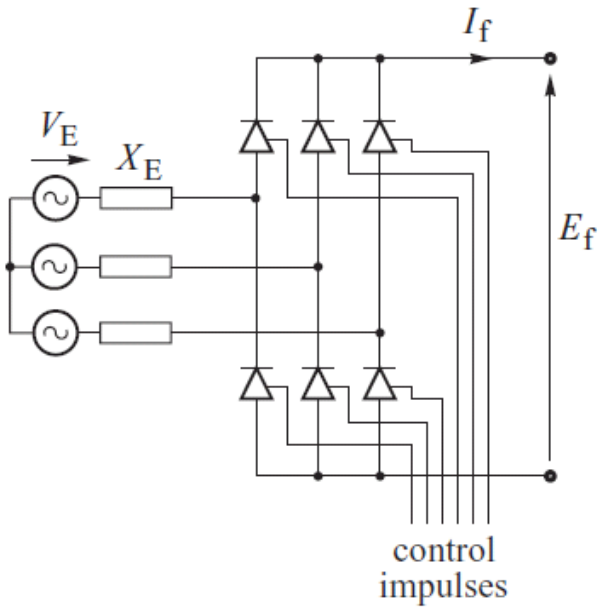


Fig. 6.9: Three phase controlled bridge rectifier

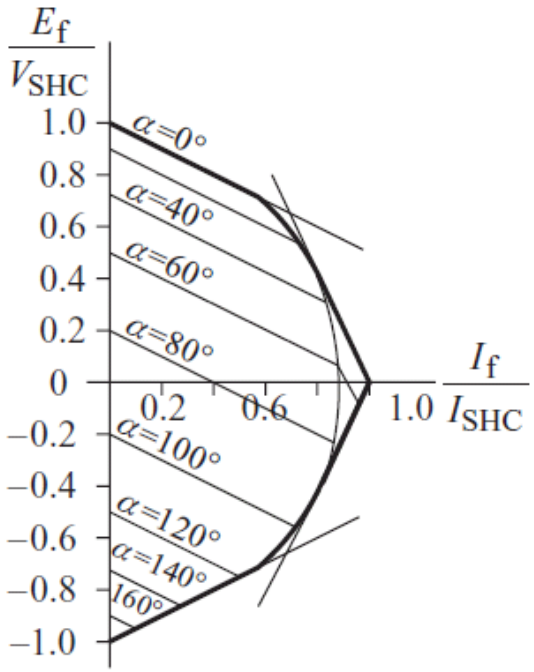


Fig. 6.10: Voltage-Current characteristic of three phase controlled bridge rectifier

The complete excitation system with static exciter can be modelled by the block diagram shown in Fig. 6.11.

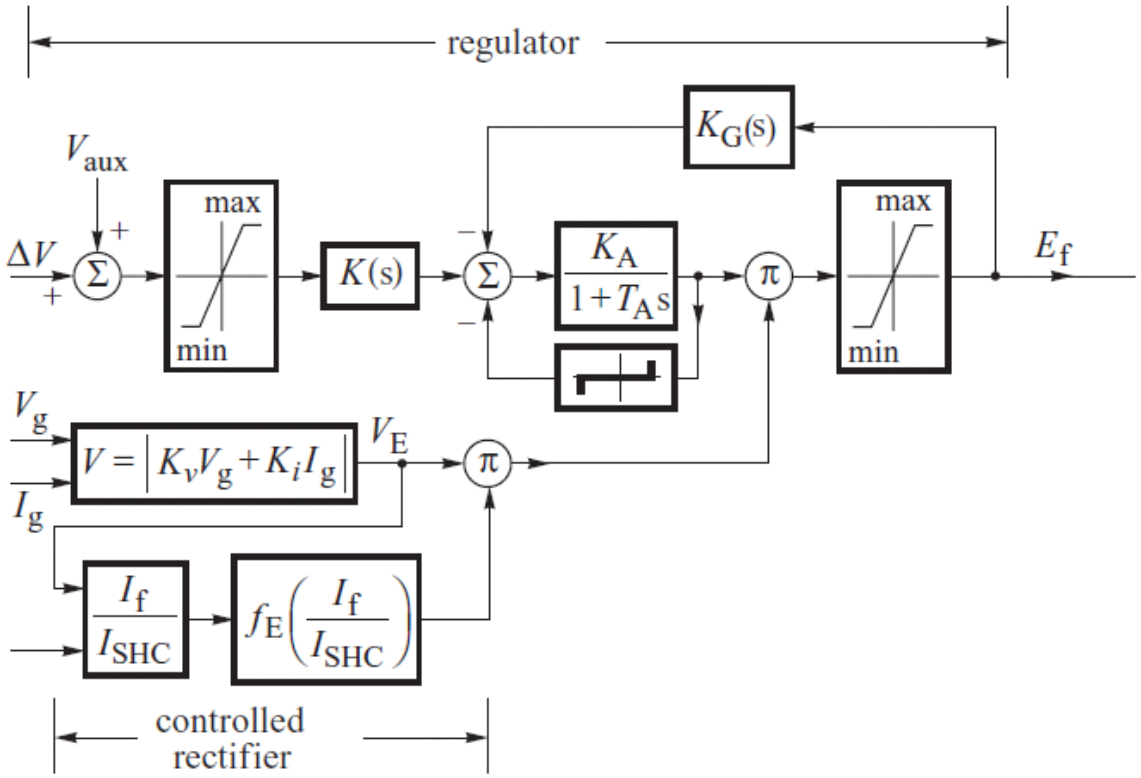


Fig. 6.11: Block diagram of the excitation system with a static exciter

In this figure the regulator and the stabilization element are shown in the upper part of the diagram while the static characteristic of the rectifier is shown in the bottom part of the diagram. The rectifier supply voltage V_E is proportional to both the generator armature voltage and armature current as determined by the constants K_v and K_i . The values of these constants depend on how the rectifier is fed.When K_i= 0 and K_v= 1 the system has no load compensation and corresponds to the rectifier being supplied directly from the terminals of the generator.

The way in which the main generator field current effects the rectifier output voltage is modelled in the same way as the uncontrolled rectifier. The regulator, together with the firing circuits, are modelled by a first-order transfer function with gain K_A and time constant T_A. If the systems does not contain cosinusoidal compensation of the voltage–firing angle dependency the gain $K_A$will not be constant and should be modelled as a cosine function of the regulator signal. The system is stabilized by the transfer function K(s) in the forward path and by feedback of the exciter output voltage through the block $K_G(s)$.

Although K(s)and $K_G(s)$ can be implemented using digital or analogue methods, digital AVRs are becoming common as they allow more sophisticated functions to be built into the AVR while only software changes are needed between different generators. The exciter output voltage E_f is given by the product of the supply voltage and the regulator output signal which represents the firing angle. If the influence of the generator field current on the rectifier output voltage is neglected then the exciter block diagram may be simplified to that consisting of the transfer function of the regulator and its stabilization element.

6.3 Power system stabilizer PSS

Power System Stabilizers are used for many years to add damping to the electromechanical oscillations. It will act through the generator excitation system which produces a component of electrical torque according to the speed deviation is generated (in addition to the damping torque). However it is easy to implement the PSS as its function mainly depends upon the modes of oscillation. i.e. whether it is local or inter-area mode. The highly efficient stabilizer which produces a damping torque over a wide range of frequencies where as less efficient stabilizer for a small range of frequencies only, which makes problem when the system changes its oscillation mode also change correspondingly.

Power System Stabilizer is used to provide additional modulation signal to the reference input of automatic voltage regulator. Due to this idea an electrical torque is produced in the generator proportional to the speed deviation. In earlier days PSS consists of lead block to adjust the input signal to give the correct phase.

General inputs to the PSS are generator shaft speed, terminal bus frequency, electrical or accelerating power. It consists of gain block, lead − lag block and a high pass filter which is called as a washout block in power system. The low frequency oscillations are observed in the power system since 1960s in the interconnected large scale power systems with weak tie-lines due to the insufficient damping provided for the system.

These oscillations will grow and deteriorate the system performance and leads to system instability. It also noted that the change in loading conditions which make the machine parameters also vary for different operating conditions so that the stabilizer should maintain the stability of the system even though the change occur in a complex manner. Therefore the PSS should be robustness to the changing machine parameters, loading and operating conditions etc.

Power System Stabilizer is the effective one for damping electromechanical oscillations especially in interconnected power system. There are two optimization problem related to PSS. One is selecting the optimum location of PSS and determining the optimum stabilizer parameters. When the stabilizers are correctly tuned the resulting damping action will be robust. The main advantage of PSS is a cost effective one when compared to the power electronics based FACTS controller when used for damping application. PSS have been used for over 20 years in the western part of United States of America Ontario Hydro[55]. In United Kingdom it was reported that PSS have been used for damping of oscillations when the power is transmitted for long distance with weak AC tie-lines like connecting Scotland and England. However PSS is not used under normal operating condition it will be service at abnormal or unusual condition which may occur sometimes. Therefore PSS is necessary to operate along with modern excitation systems to damp out the oscillations effectively and also to enhance the system stability.

The basic structure, modelling, and perfonnance of Speed-based stabilizers are illustrated by considering a thyristor excitation system. Fig. 6.12shows the block diagram of the excitation system, including the AVR and PSS. Since we are concemed with small-signal performance, stabilizer output limits and exciter output limits are not shown in the figure. The following is a brief description of the basis for the PSS configuration and considerations in the selection of the parameters.

The PSS representation in Fig. 6.12 consists of three blocks: a phase compensation block, a signal washout block, and a gain block. The phase compensation block provides the appropriate phase-lead characteristic to compensate for the phase lag between the exciter

input and the generator electrical (air-gap) torque. The figure shows a single first-order block. In practice, two or more first-order blocks may be used to achieve the desired phase compensation. In some cases, second-order blocks with complex roots have been used.

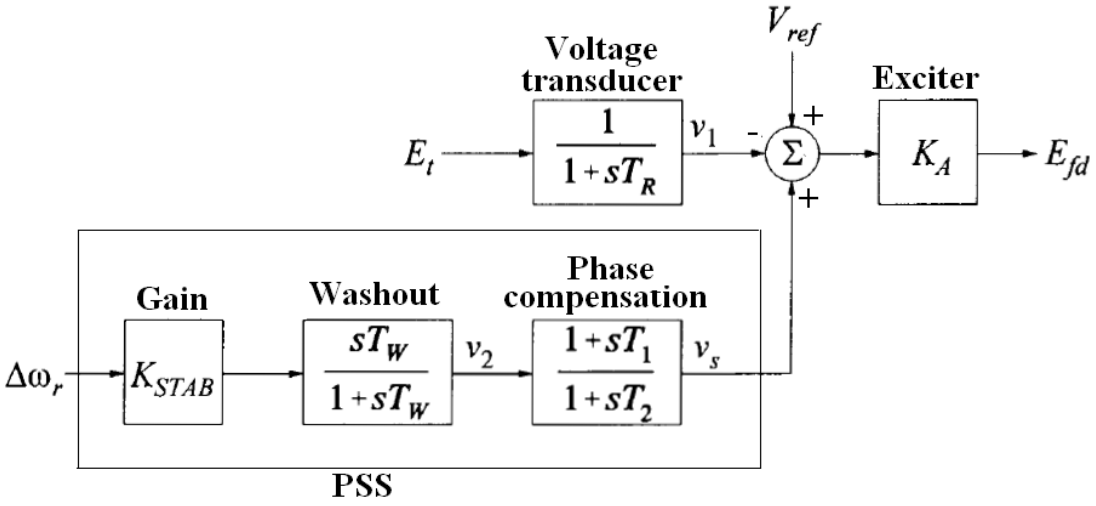


Fig. 6.12: Thyristor excitation system with AVR and Speed-based PSS

Normally, the frequency range of interest is 0.1 to 2.0 Hz, and the phase-lead network should provide compensation over this entire frequency range. The phase characteristic to be compensated changes with system conditions; therefore, a compromise is made and a characteristic acceptable for different system conditions is selected. Generally some undercompensation is desirable so that the PSS, in addition to significantly increasing the damping torque, results in a slight increase of the synchronizing torque.

The signal washout block serves as a high-pass filter, with the time constant T_w high enough to allow signals associated with oscillations in ω_r to pass unchanged. Without it, steady changes in speed would modify the terminal voltage. It allows the PSS to respond only to changes in speed. From the viewpoint of the washout function, the value of T_w is not critical and may be in the range of 1 to 20 seconds. The main consideration is that it be long enough to pass stabilizing signals at the frequencies of interest unchanged, but not so long that it leads to undesirable generator voltage excursions during system-islanding conditions.

The stabilizer gain K_{STAB} determines the amount of damping introduced by the PSS. Ideally, the gain should be set at a value corresponding to maximum damping; however, it is often limited by other considerations. In applying the PSS, care should be taken to ensure that the overall system stability is enhanced, notjust the small-signal stability.

7 INTERNET NETWORK MODEL/TRUETIME

Computer-based control systems and networked control systems are hybrid systems where continuous time-driven dynamics and discrete time-driven dynamics interact. TrueTime is a Matlab/Simulink-based simulator for networked and embedded control systems that has been developed at Lund University since 1999. The simulator software consists of a Simulink block library and a collection of MEX files. The kernel block simulates a real-time kernel by executing user-defined tasks and interrupt handlers. The various network blocks allow nodes (kernel blocks) to communicate over simulated wired or wireless networks. Fig. 7.1 shows TrueTime block library of Trutime 2.0 beta 6 2010 version.

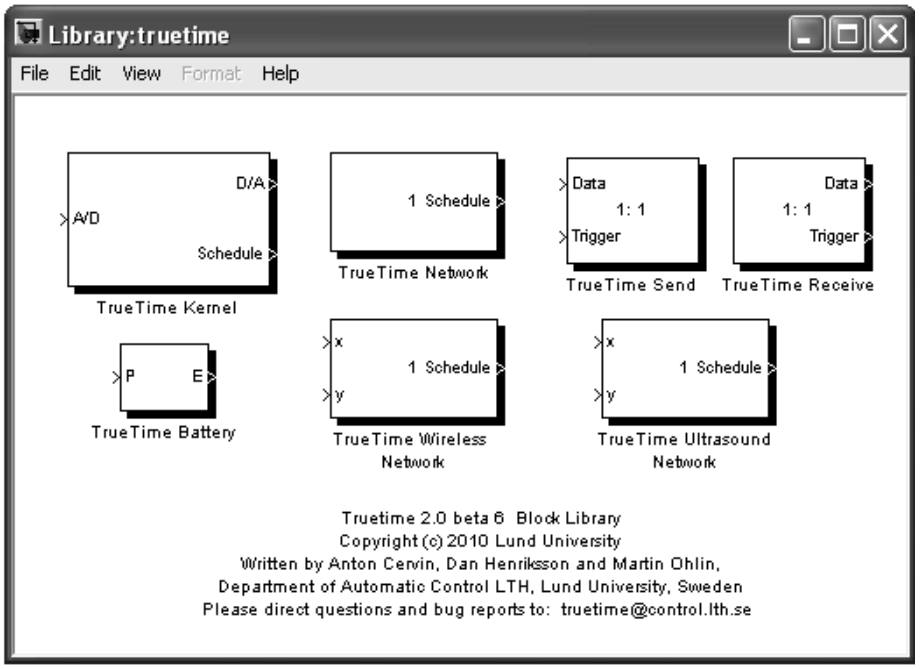


Fig.7.1: TrueTime block library

The TRUETIME blocks are connected with ordinary Simulink blocks to form a realtime control system. Before a simulation can be run, however, it is necessary to initialize kernel blocks and network blocks, and to create tasks, interrupt handlers, timers, events, monitors,.. etc for the simulation. The initialization code and the code that is executed during simulation may be written either as Matlab M-files or as C++ code

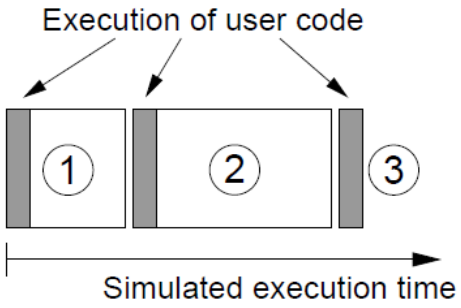


Fig. 7.2: Execution of user code and sequence of segments

The execution of tasks and interrupt handlers is defined by code functions. A code function is further divided into code segments according to the execution model shown in Fig. 7.2. All execution of user code is done in the beginning of each code segment. The execution time of each segment should be returned by the code function.

The standalone network blocks, named TrueTime Send and TrueTime Receive, as seen in Figure 1, can be used to send messages using the network blocks without using kernel blocks. This makes it possible to create TrueTime network simulations without having to initialize kernels.

The TrueTime Battery block acts as a power source for the battery enabled kernel blocks. It uses a simple integrator model so it can be both charged and recharged. The only one parameter in its block mask is the initial power. TrueTime Ultrasound Network and TrueTime Wireless Network blocks are used to simulate wireless networks

The TrueTime Network block simulates medium access and packet transmission (physical and medium access layer) in a local area network. The following subsections describe TrueTime Network and TrueTime Kernel blocks in some detail.

7.1 TrueTime kernel

The kernel block is a Simulink S-function that simulates a computer with a real-time kernel, A/D and D/A converters, a network interface, and external interrupt channels. The kernel executes user-defined tasks and interrupt handlers. Internally, the kernel maintains several data structures that are commonly found in a real-time kernel: a ready queue, a time queue, and records for tasks, interrupt handlers, monitors and timers that have been created for the simulation..

The block is configured by the block mask dialog as shown in Fig. 7.3. The main parameter is the name of the initialization function, because each kernel block has to be initialized at the start of the simulation. An optional argument for the initialization script can be set, also the battery option, the clock drift and the clock offset can be set.

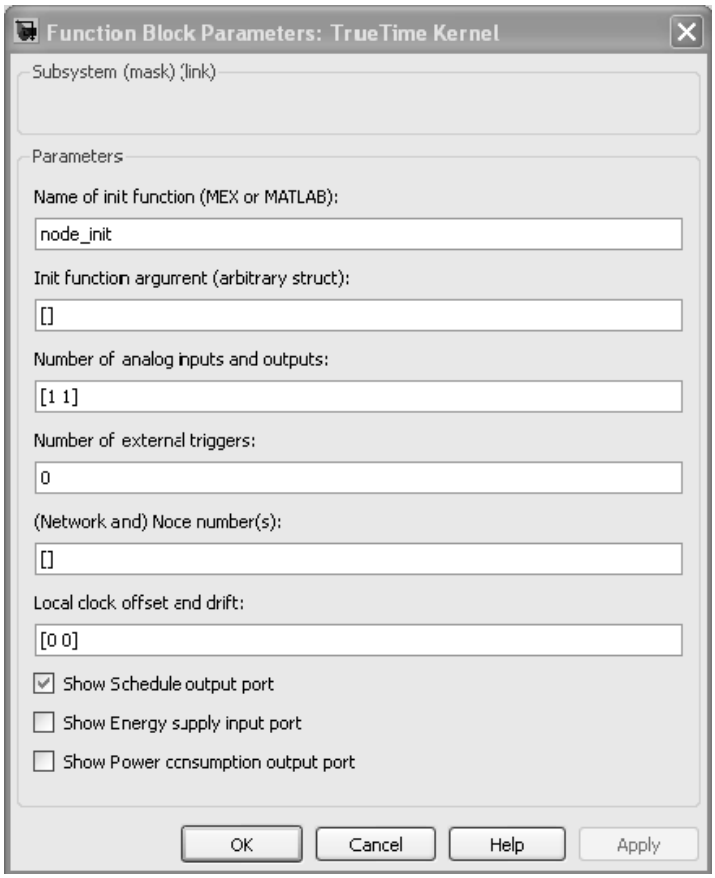


Fig.7.3: TrueTime Kernel block mask

Initialization of a TRUETIME kernel block involves specifying the number of inputs and outputs of the block, defining the scheduling policy, and creating tasks, interrupt handlers, events, monitors, etc for the simulation. This is done in an initialization script for each kernel block. The initialization script can (in the Matlab case) also take an optional parameter to limit the number of similar code functions. The TrueTime Kernel block mask include the following parameters:

- **Init function:** The name of the initialization script.
- **Init function argument:** An optional argument to the initialization script. This can be any Matlab struct.
- **Input and outputs number:** Specify number of inputs and outputs.
- **External triggers:** Specify number of external triggers used to excute task code.
- **Network number:** Specify the network nember which the kernel belong to.
- **Energy supply and Power consumption options:** Enable this check box if the kernel should depend on a power source.
- **Clock drift:** The time drift, 0.01 if the local time should run 1% faster than the nominal time (the actual simulation time).
- **Clock offset:** A constant time offset from the nominal time.

Initializing script code in Matlab of TrueTime Kernel Block can be as the following script example:

```
function node_inite
ttInitKernel('prioFP')
%
%
```

```
% Initializing code and specify task function parameters
%
%
ttCreatePeriodicTask('ctrl_task', starttime, period, 'ctrl_code', data)
```

Where node_inite is user defined initializing function name, which is defined in TrueTime Kernel block mask.

Function *ttInitKernel* Initializes the TrueTime Kernel. This function performs necessary initializations of the kernel block and must be called first of all in the initialization script. The priority function should be one of the following in the Matlab case: 'prioFP' (fixed-priority scheduling), 'prioDM' (deadline-monotonic scheduling), or 'prioEDF' (earliest-deadline-first scheduling, with support for constant bandwidth servers). A task associated with a CBS is scheduled according to the deadline of the CBS and not of the task.

Function *ttCreatePeriodicTask* is used to create a periodic task to run in the TrueTime Kernel . The periodicity is implemented internally by the kernel using a periodic timer. Its arguments are: '*ctrl_task'* the name of the task, must be a unique non-empty string, *starttime* specify release time for the first job of the periodic task, *period* is period of the task, '*ctrl_code*' is The code function of the task, where ctrl_code is a string (name of an m-file).

7.2 The TrueTime network

The TRUETIME network block simulates medium access and packet transmission in a local area network. When a node tries to transmit a message (using the primitive *ttSendMsg*), a triggering signal is sent to the network block on the corresponding input channel. When the simulated transmission of the message is finished, the network block sends a new triggering signal on the output channel corresponding to the receiving node. The transmitted message is put in a buffer at the receiving computer node. A message contains information about the sending and the receiving computer node, arbitrary user data (typically measurement signals or control signals), the length of the message, and optional real-time attributes such as a priority or a deadline.

Six simple models of networks are supported: CSMA/CD (e.g. Ethernet), CSMA/ AMP (e.g. CAN), Round Robin (e.g. Token Bus), FDMA, TDMA (e.g. TTP), and Switched Ethernet. The propagation delay is ignored, since it is typically very small in a local area network. Only packet-level simulation is supported, it is assumed that higher protocol levels in the kernel nodes have divided long messages into packets, etc.

The network block is configured through the block mask dialog, see Fig. 7.4. it is also possible to change some parameters on a per node basis by using the command *ttSetNetworkParameter*.

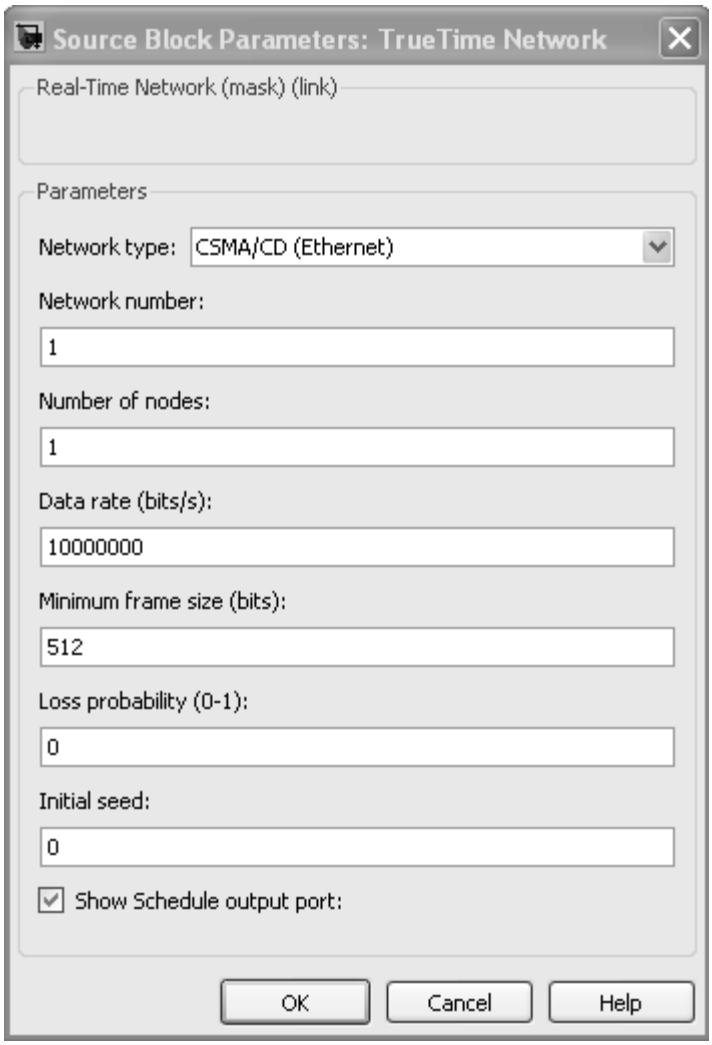


Fig. 7.4: TrueTime Network block mask

The following network parameters are common to all models:

- **Network number:** The number of the network block. The networks must be numbered from 1 and upwards. Wired and wireless networks are not allowed to use the same number.
- **Number of nodes:** The number of nodes that are connected to the network. This number will determine the size of the Snd, Rcv and Schedule input and outputs of the block.
- **Data rate (bits/s):** The speed of the network.
- **Minimum frame size (bits):** A message or frame shorter than this will be padded to give the minimum length. Denotes the minimum frame size, including any overhead introduced by the protocol. E.g., the minimum Ethernet frame size, including a 14-byte header and a 4-byte CRC, is 512 bits.
- **Pre-processing delay (s):** The time a message is delayed by the network interface on the sending end. This can be used to model, e.g., a slow serial connection between the computer and the network interface.
- **Post-processing delay (s):** The time a message is delayed by the network interface on the receiving end.

- **Loss probability (0–1):** The probability that a network message is lost during transmission. Lost messages will consume network bandwidth, but will never arrive at the destination.

Now we shall talk about CSMA/CD (Ethernet) and switched Ethernet network types in some details.

7.2.1 CSMA/CD (Ethernet)

CSMA/CD stands for Carrier Sense Multiple Access with Collision Detection. If the network is busy, the sender will wait until it occurs to be free. A collision will occur if a message is transmitted within 1 microsecond of another (this corresponds to the propagation delay in a 200 m cable; the actual number is not very important since collisions are only likely to occur when two or more nodes are waiting for the cable to be idle). When a collision occurs, the sender will back off for a time defined by:

$$t_{backoff} = \frac{minimum\ frame\ size}{data\ rate} * R \tag{7.1}$$

Where R (discrete uniform distribution) is:

$$R = rand(0.2^k - 1) \tag{7.2}$$

And k is the number of collisions in a row (but maximum 10, there is no upper limit on the number of retransmissions). Note that for CSMA/CD, minimum frame size cannot be 0.

After waiting, the node will attempt to retransmit. For an example when two nodes are waiting for a third node to finish its transmission, they will first collide with probability 1, then with probability 1/2 (k= 1), then 1/4 (k = 2), and so on.

7.2.2 Switched Ethernet

In Switched Ethernet, each node in the network has its own, full-duplex connection to a central switch. Compared to an ordinary Ethernet, there will never be any collisions on the network segments in a Switched Ethernet. The switch stores the received messages in a buffer and then forwards them to the correct destination nodes. This common scheme is known as store and forward.

If many messages in the switch are destined for the same node, they are transmitted in FIFO order. There can be either one queue that holds all the messages in the switch, or one queue for each output segment. In case of heavy traffic and long message queues, the switch may run out of memory. As it is illustrated in Fig. 7.5, the following options are associated with the Switched Ethernet:

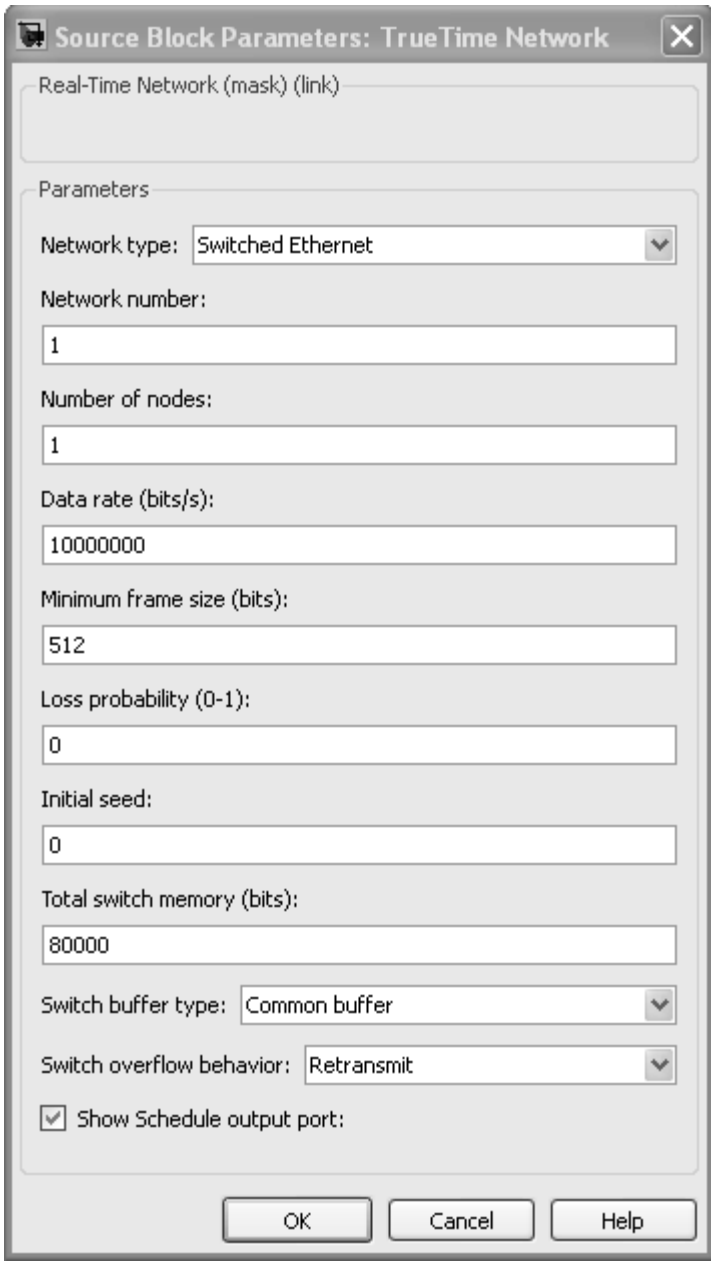


Fig. 7.5: Switched Ethernet block mask parameters.

- **Total switch memory (bits):** This is the total amount of memory available for storing messages in the switch. An amount of memory equal to the length of the message is allocated when the message has been fully received in the switch. The same memory is deallocated when the complete message has reached its final destination node.
- **Switch buffer type:** This setting describes how the memory is allocated in the switch. Common buffer means that all messages are stored in a single FIFO queue and share the same memory area. Symmetric output buffers means that the memory is divided into n equal parts, one for each output segment connected to the switch. When one output queue runs out of memory, no more messages can be stored in that particular queue.
- **Switch overflow behavior**: This option describes what happens when the switch has run out of memory. When the complete message has been received in the switch, it is deleted. Retransmit means that the switch then informs the sending node that it should

try to retransmit the message. Drop means that no notification is given, the message is simply deleted.

Appendex C contains tables of TRUETIME commands using for writing initializing scripts, task code functions and interrupt handler code functions.

8 SIMULATION AND RESULTS

8.1 Three phase synchronous generator steady-state model

Fig. 8.1 shows the implemntation of three phase synchronous generator steady-state model illusterated in chapter 4.2. The generator parameters are illustrated in Appendex A.

The suitable initial values of flux linkages, speed and osilator were adjusted to avoid transient and starting conditions in simulation.

Voltages v_a, v_b and v_c applied to stator windings are three balanced phases which are:

$$v_a = v_m \cos(\omega t) \tag{8.1}$$

$$v_b = v_m \cos(\omega t - \frac{2\pi}{3}) \tag{8.2}$$

$$v_c = v_m \cos(\omega t + \frac{2\pi}{3}) \tag{8.3}$$

In this case the operation conditions are simiral to conditions in which the generator is connected directly to infinite network with neglected ohmic and inductance resistant of the network.

Fig. 8.2 and Fig. 8.3 show simulation 1 results of implementing mechanical torque of 0.8 pu in the begining of simulation, 0 pu at time 3 s, 1 pu at time 6 s and finally 0.5 pu at time 9 s until simulation end.

The field voltage implemented during simulation is constant at 1 pu.

Fig. 8.5 and Fig. 8.5 show simulation 2 results of changing the field voltage implemented to field winding by 20% steps, that's mean implementing field voltage of 1 pu in the begining of simulation, 0.8 pu at time 3 s, 1 pu at time 6 s and finally 1.2 pu at time 9 s until simulation end.

The mechanical torque implemented during simulation is constant at 0.8 pu.

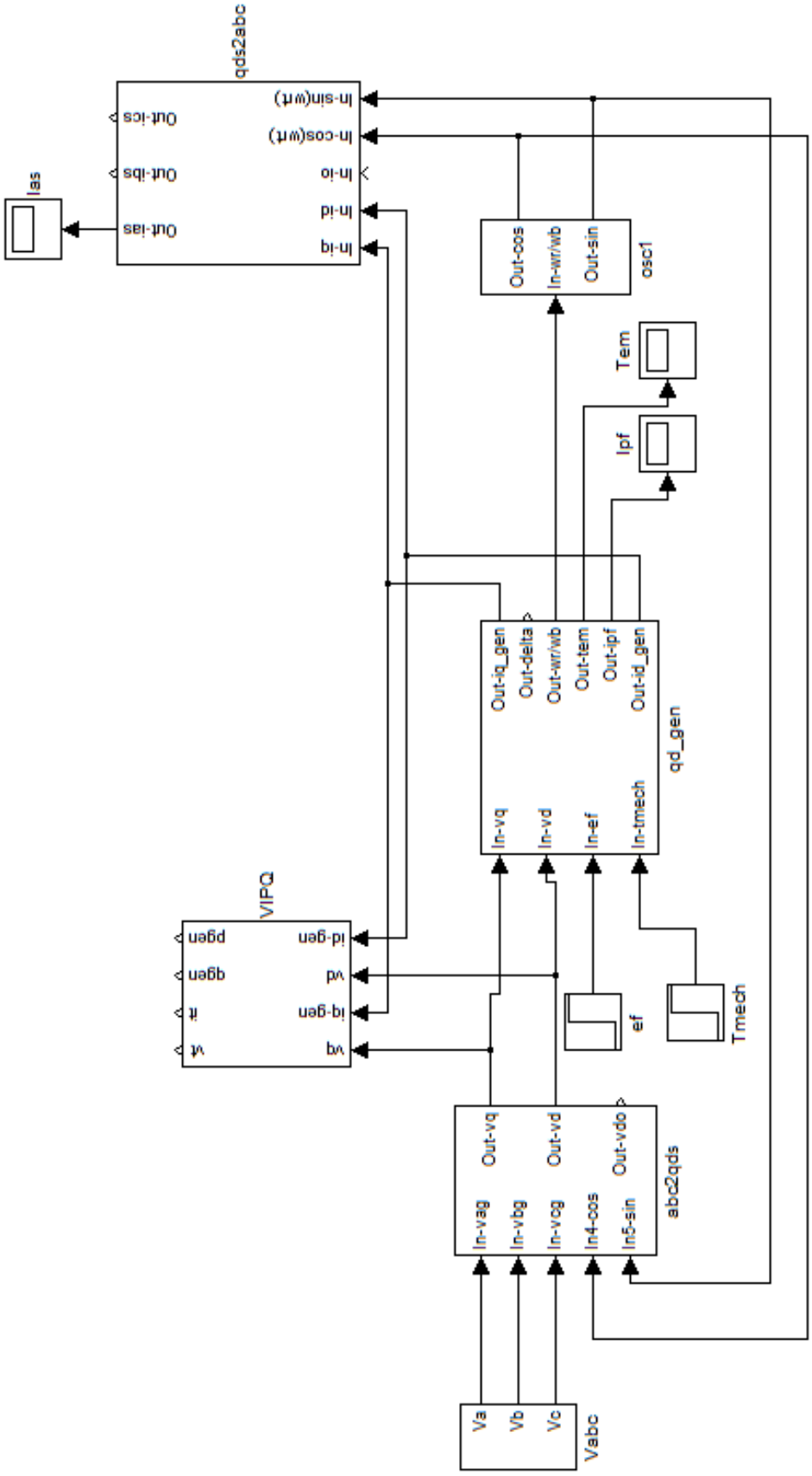


Fig. 8.1: Three phase synchronous generator model.

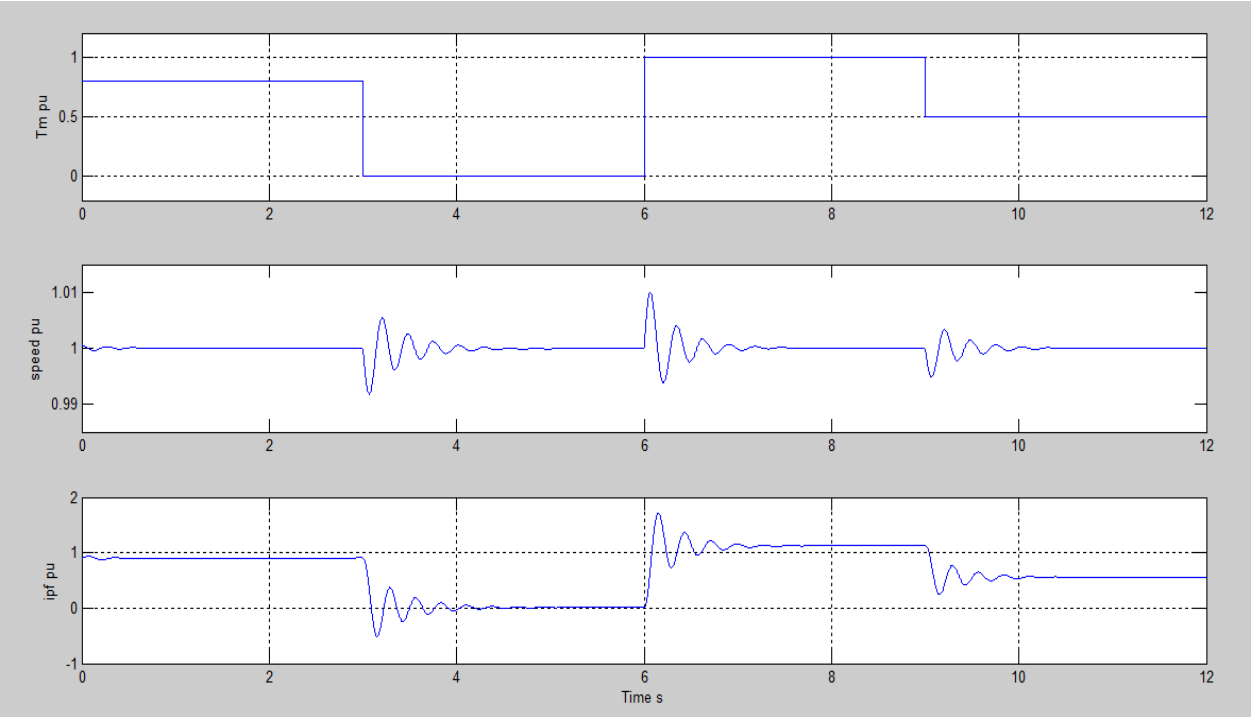


Fig. 8.2: Torque, speed and field current results of simulation 1 in pu.

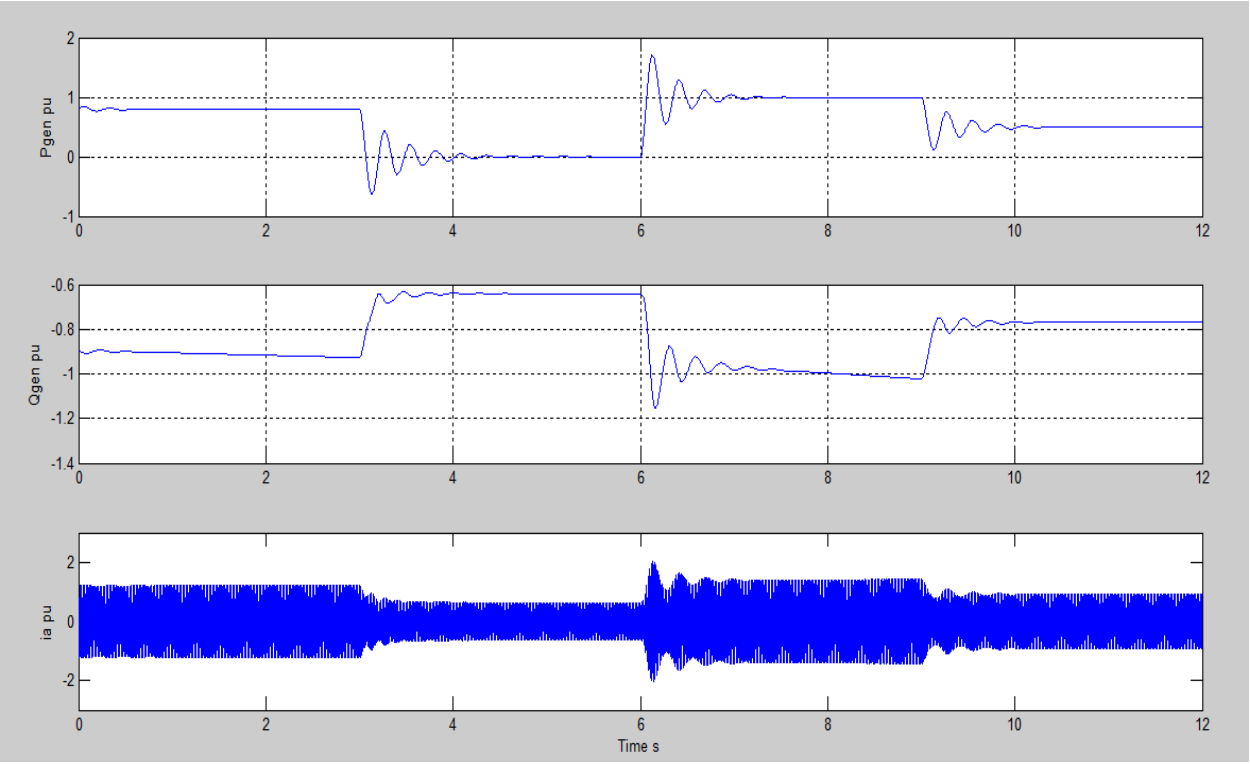


Fig. 8.3: Active power, reactive power and phase a current results of simulation 1 in pu.

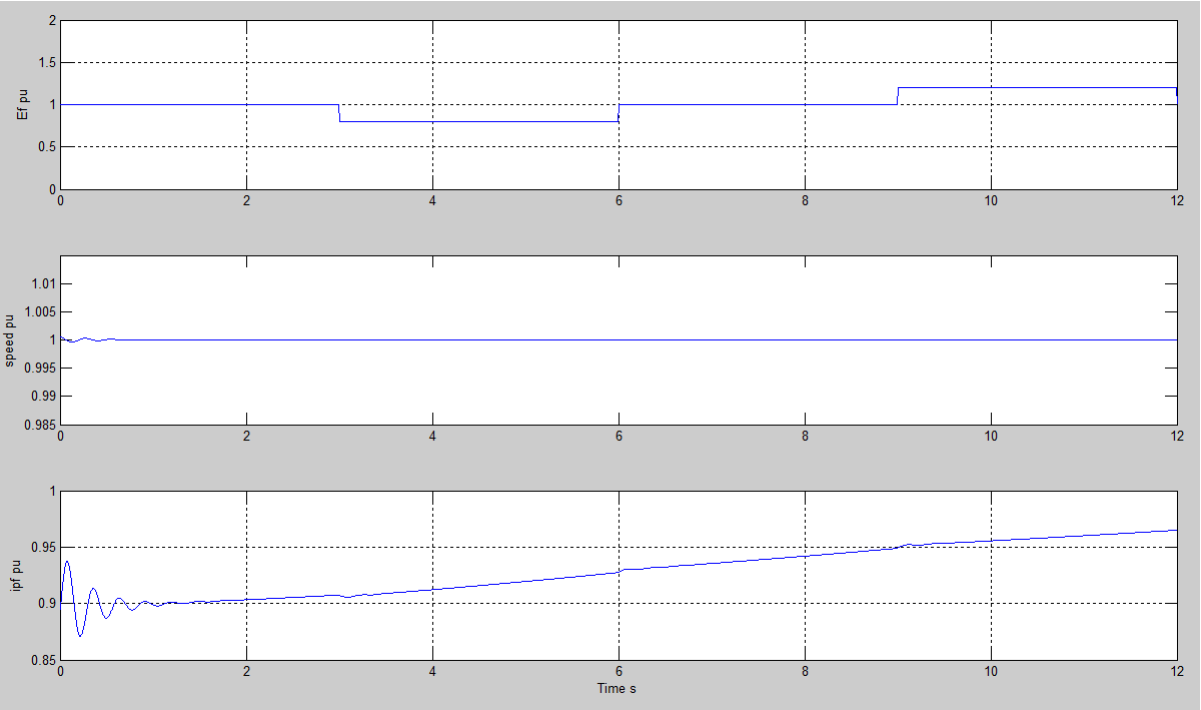


Fig. 8.4: Torque, speed and field current results of simulation 2 in pu.

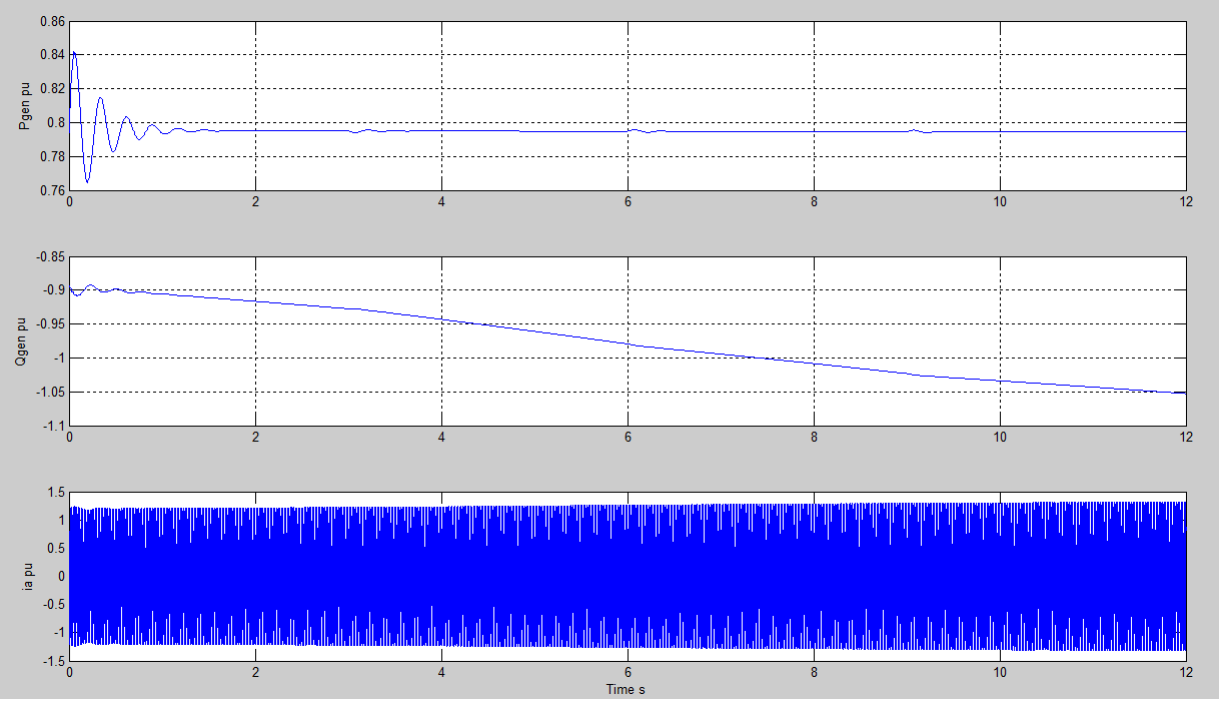


Fig. 8.5: Active power, reactive power and phase a current results of simulation 2 in pu.

8.2 Three phase synchronous generator transient model/ hydro power plant model

In this section we will show results of implementing three phase synchronous generetor transient model discussed in chapter 4.3, with excitation system and hydrualic turbine.

Parameters of generetor, exciter and turbine illustrated in Appendex A. Fig. 8.6 shows overall diagram of the hydro power planet, while Fig. 8.7, Fig. 8.8 and Fig. 8.9 show inside the Synchronous Generator, Excitation system and Hydrualic Turbine blocks respectively.

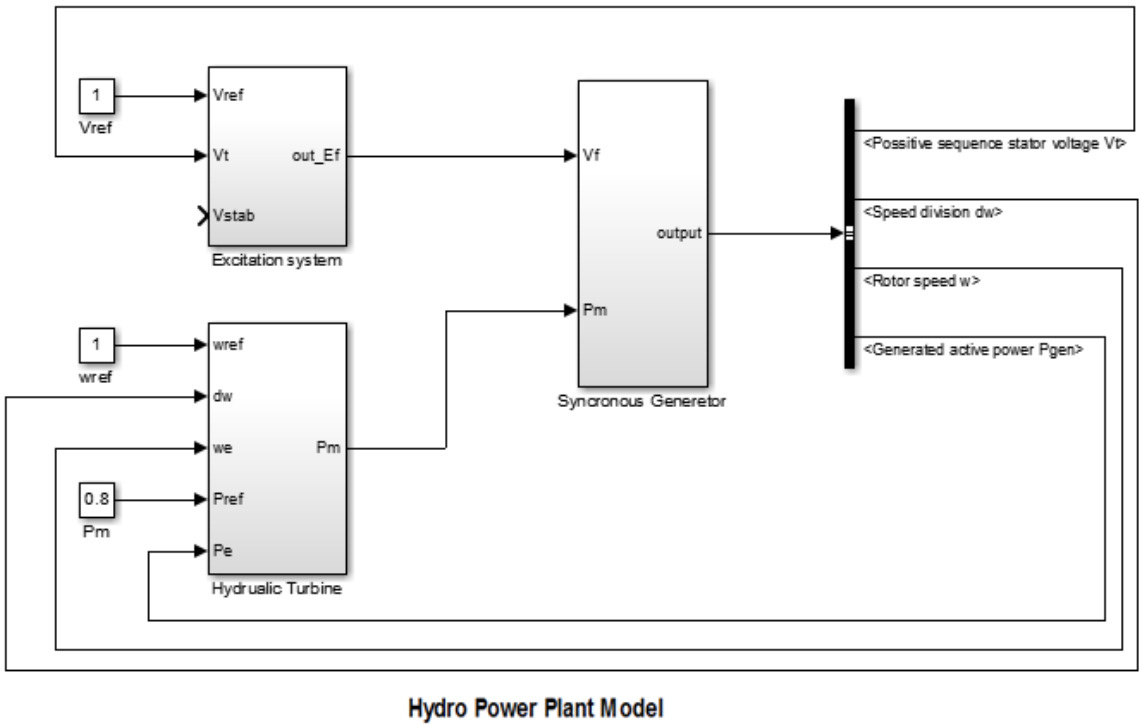


Fig. 8.6: Overall diagram of hydro power plant

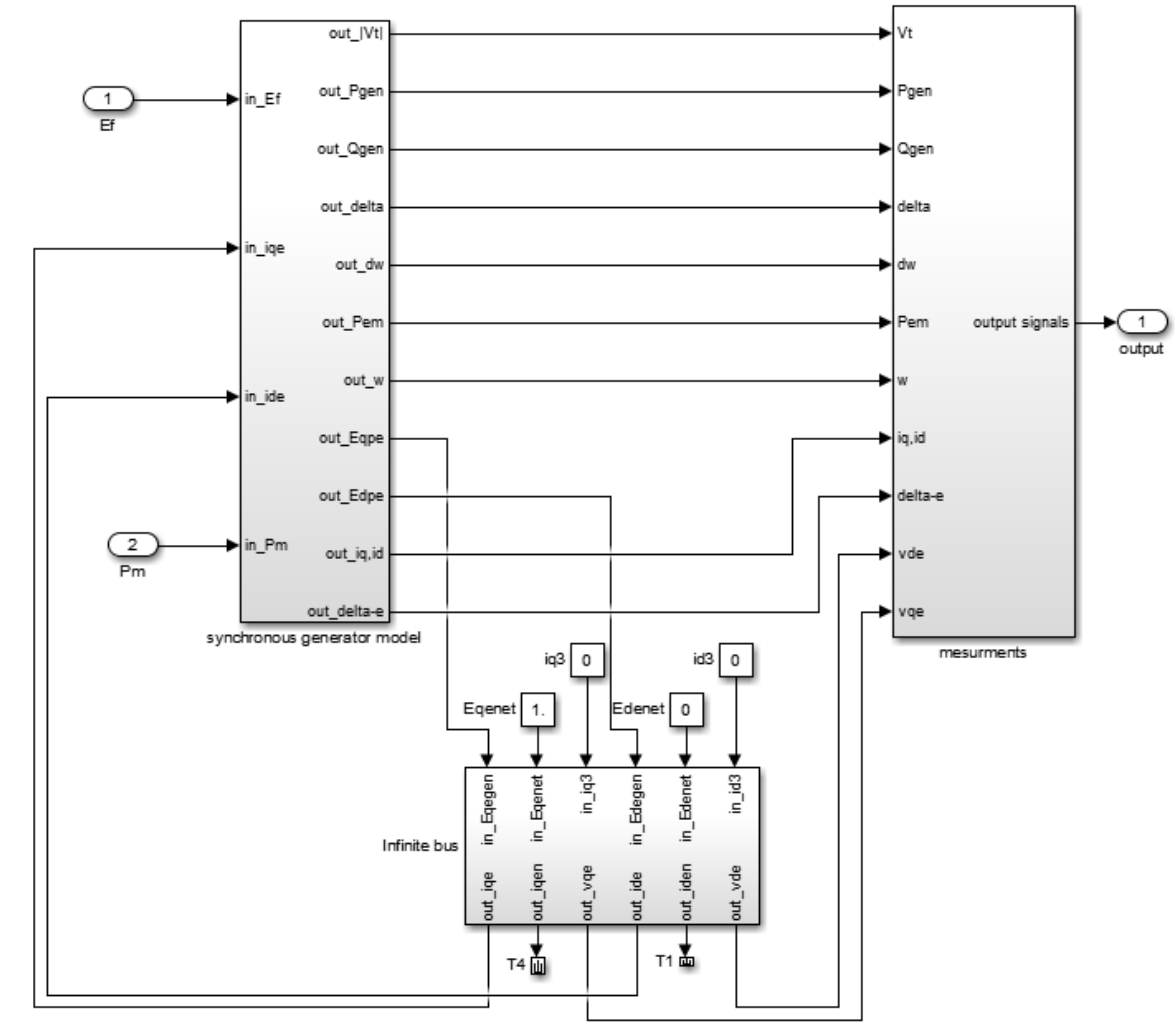
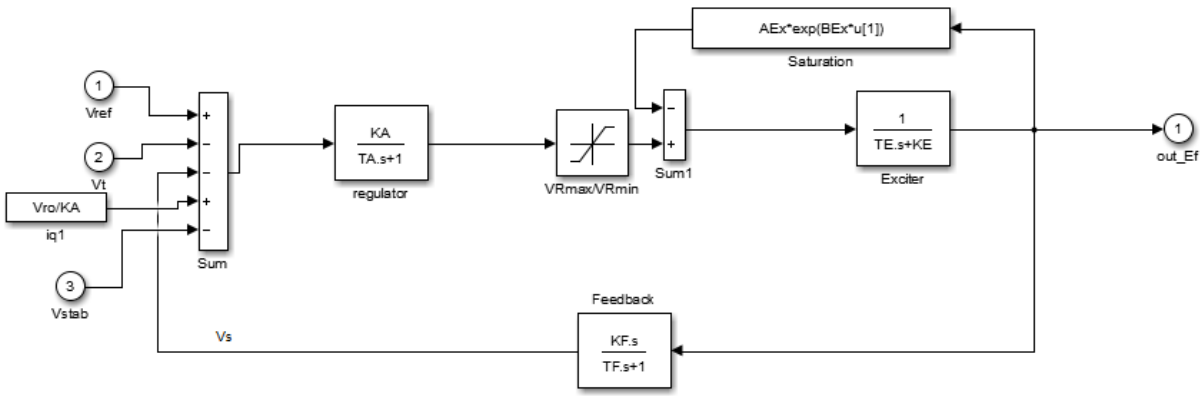


Fig. 8.7: Inside Synchronous Generator block



Excitation System

Fig. 8.8: Inside Excitation System Block

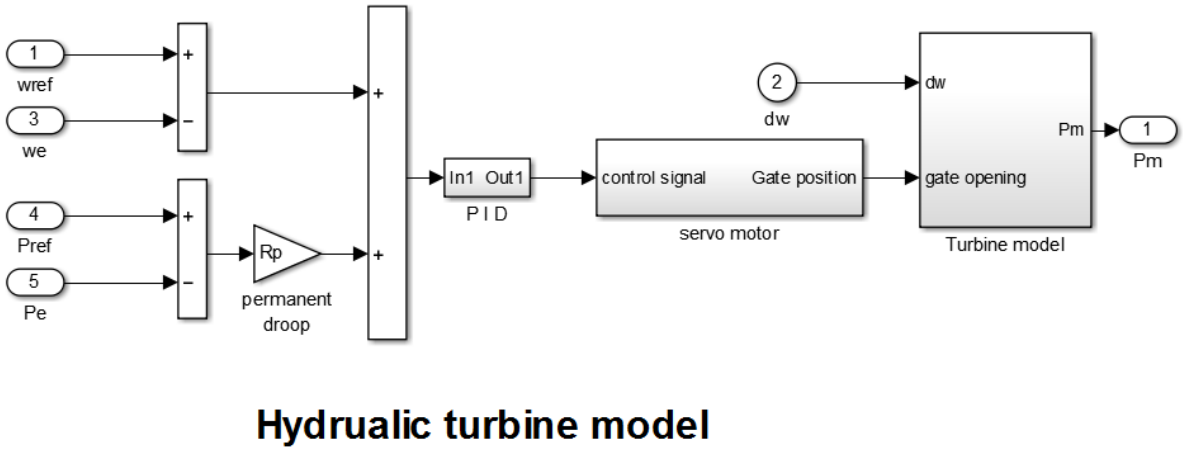


Fig. 8.9: Inside hydrualic turbine Block

Fig. 8.10 shows inside the synchronous generator model subsystem.

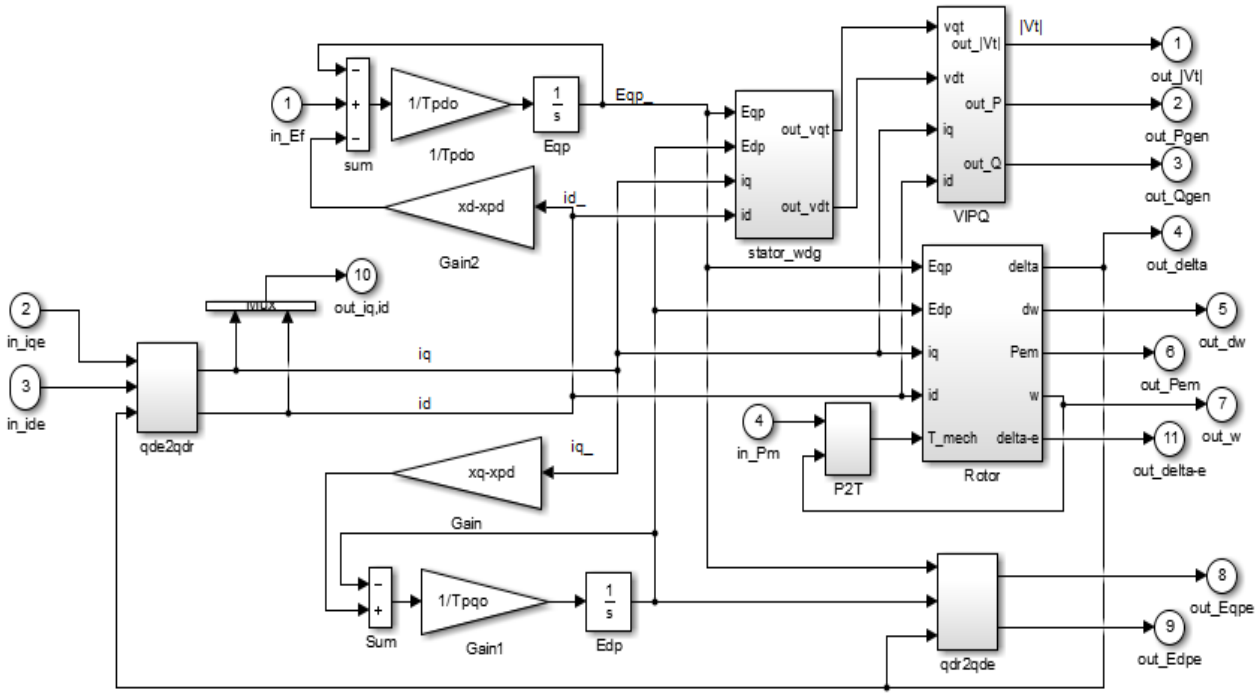


Fig. 8.10: Inside the *synchronous generator model* subsystem

Fig. 8.11 shows the hydraulic turbine time response for changing in reference power input from 0.8 pu to 0.5 pu at time 10 s.

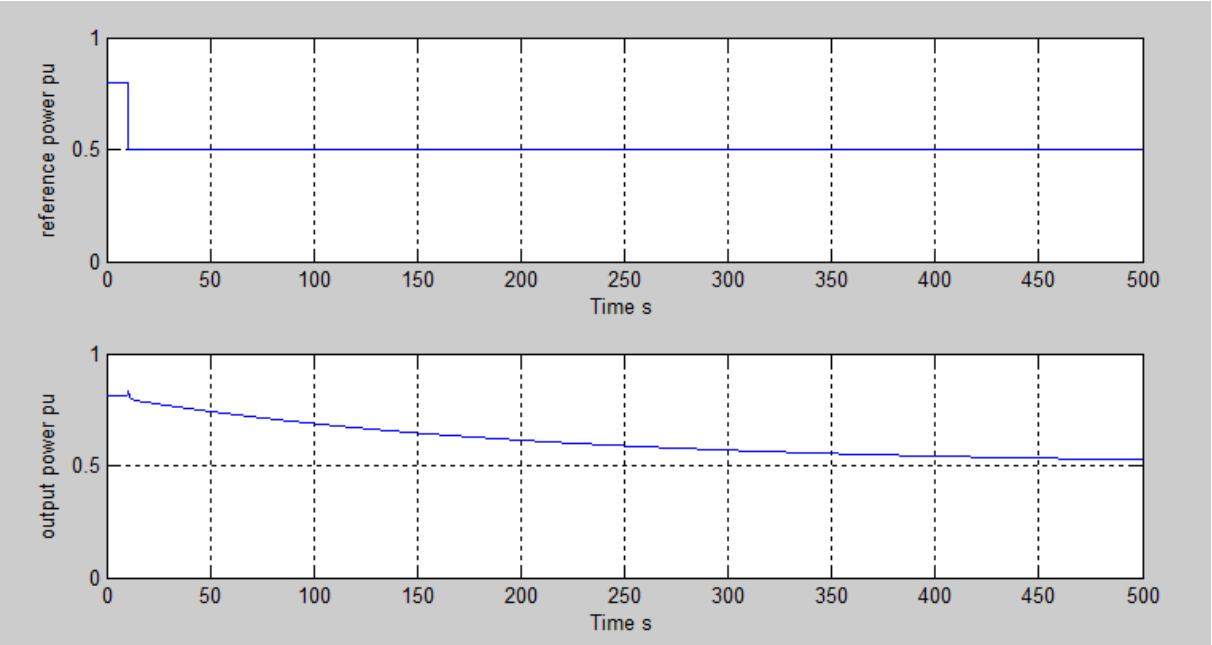


Fig. 8.11: Turbine response

The following figures (Fig. 8.12, Fig. 8.13 and Fig.8.14) show the transient response results of implementing three phase fault on the generator terminals at time 5s, the fault duration is 0.1s.

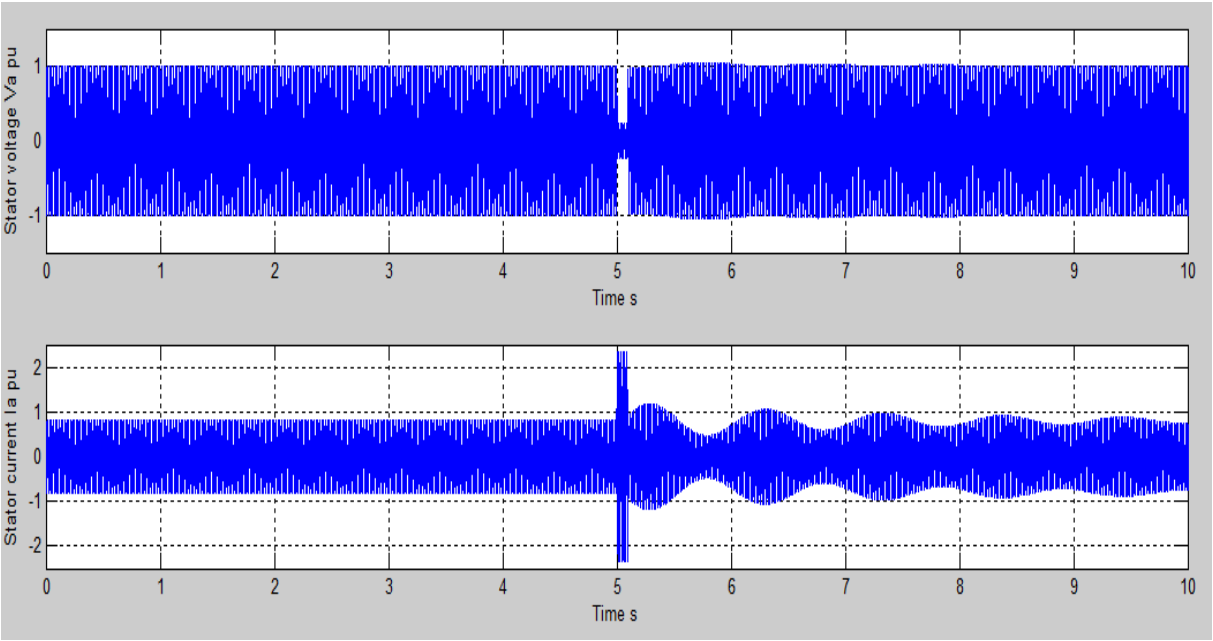


Fig. 8.12: Stator voltage Va and current Ia with fault at time 5s

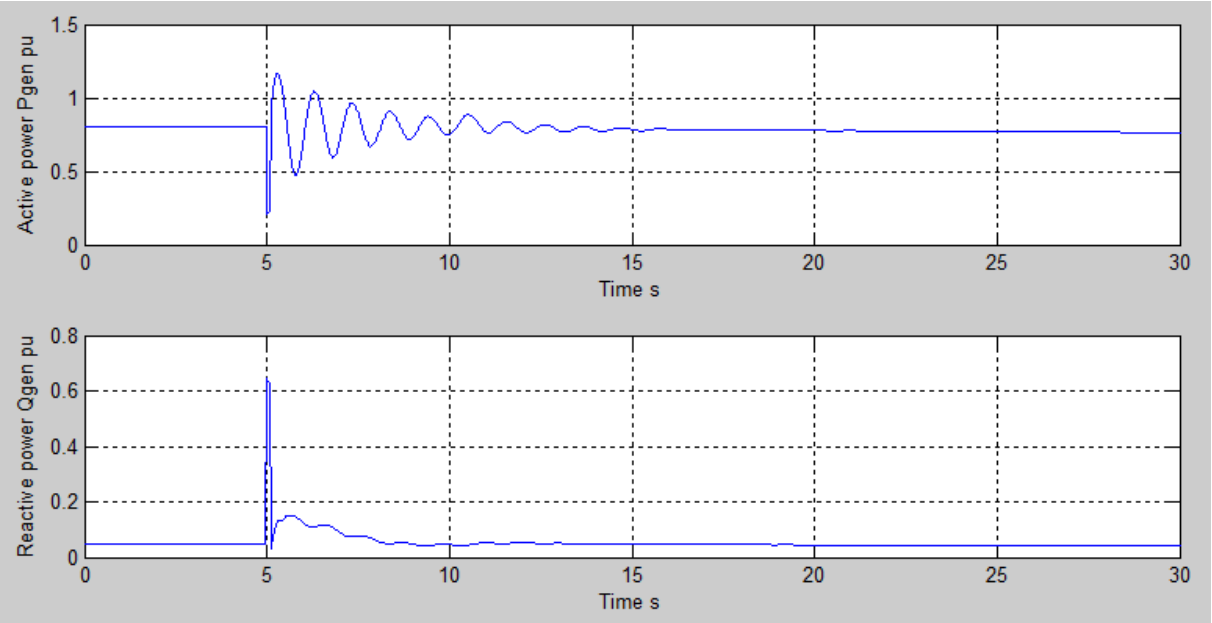


Fig. 8.13: Active power Pgen and reactive power Qgen with fault at time 5s

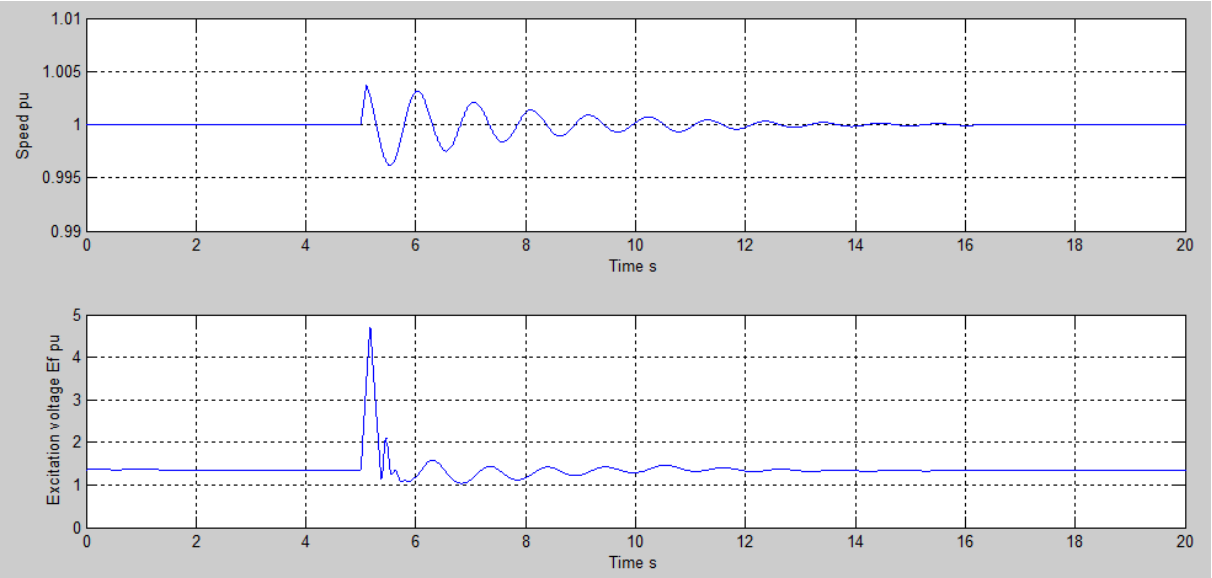


Fig. 8.14: Speed and excitation voltage with fault at time 5s

Now we will show the effect of using power system stabilizer PSS on damping the oscillation occured after fault. The stabilizer parameters are shown in Appendix A.

Fig. 8.15 and Fig 8.16 show show simulation results of speed, excitation voltage, active power and reactive power with using of PSS. In this case the generator will settle in about 4 s after fault (Fig.8.15 and Fig. 8.16), compairing with about 10s in the case without using of PSS (Fig 8.13 and Fig. 8.14).

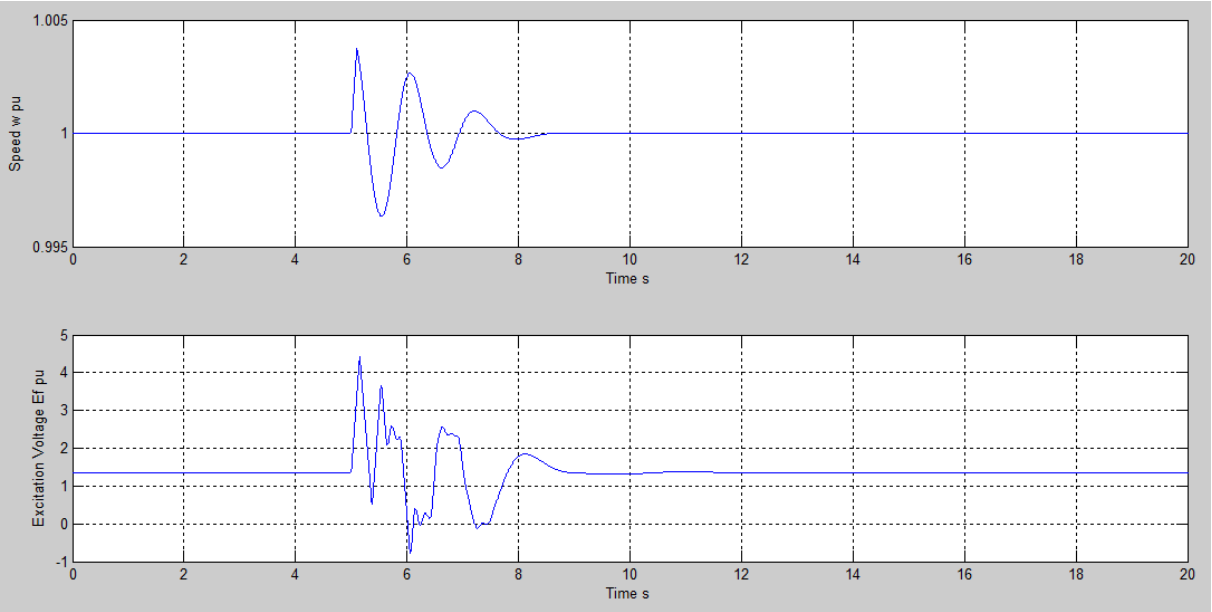


Fig. 8.15: Speed and excitation voltage results with using of PSS. (fault at time 5s)

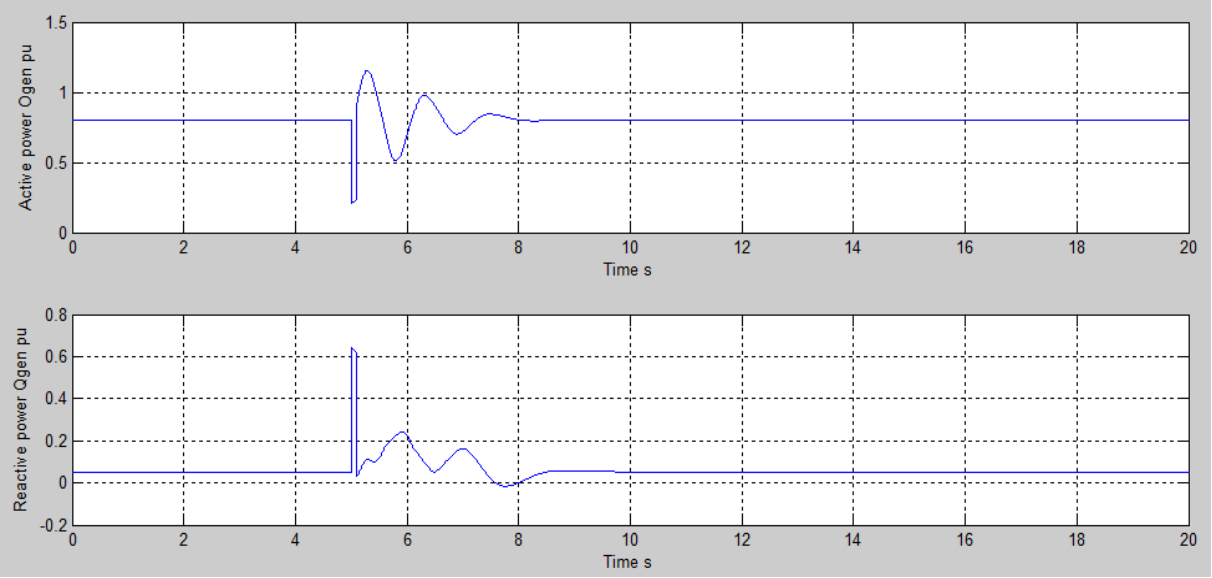


Fig. 8.16: Active power Pgen and reactive power Qgen results with using of PSS.(fault at time 5s)

8.3 Data aqcuisation system of power plant over internet network

Fig. 8.17 shows overall diagram of hydro power plant with data aqcuisation system using internet network.

TrueTime Simulink library, illustrated in chapter 7, was used to simulate the internet network. Network type Switched Ethernet was used in this simulation. Parameters of network illustrated in Appendex A. Fig. 8.18 shows inside the network block, which is consists of three nodes.

1- **Data aqcuisition node**: It recieves all signale from the plant, converts it to digital signals and sends it to actuator and database nodes.
2- **Actuator node**: It convert control signals to analog form and sends it to the plant, control signals in this case are reference values of turbine mechanical power, speed and field voltage.
3- **Database node**: It recieves signals from Data aqcuisition node, and store them in database server that they can be restored and analysed in any time.

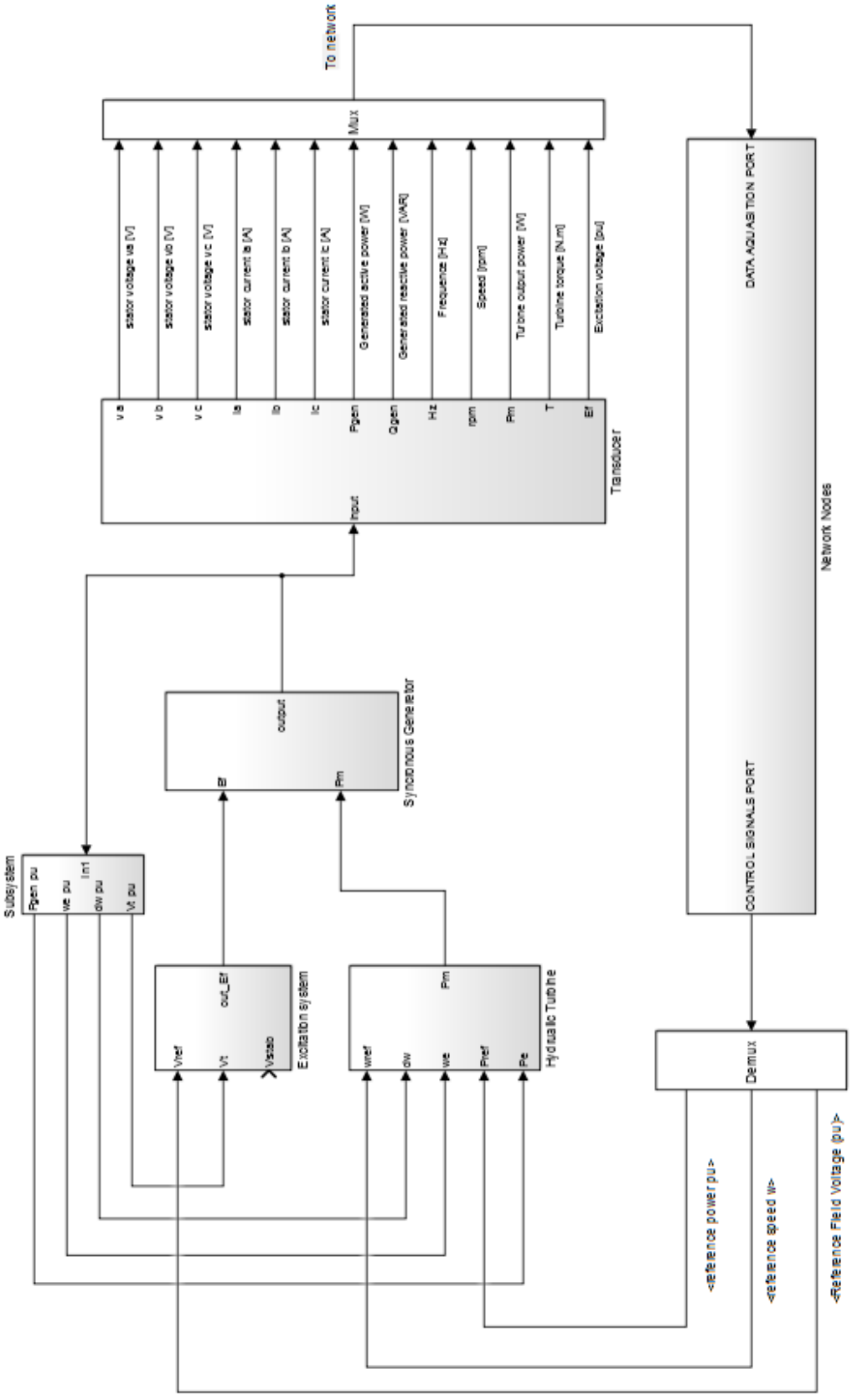


Fig. 8.17: Data Aqcuisition system of hydro power plant using internet network.

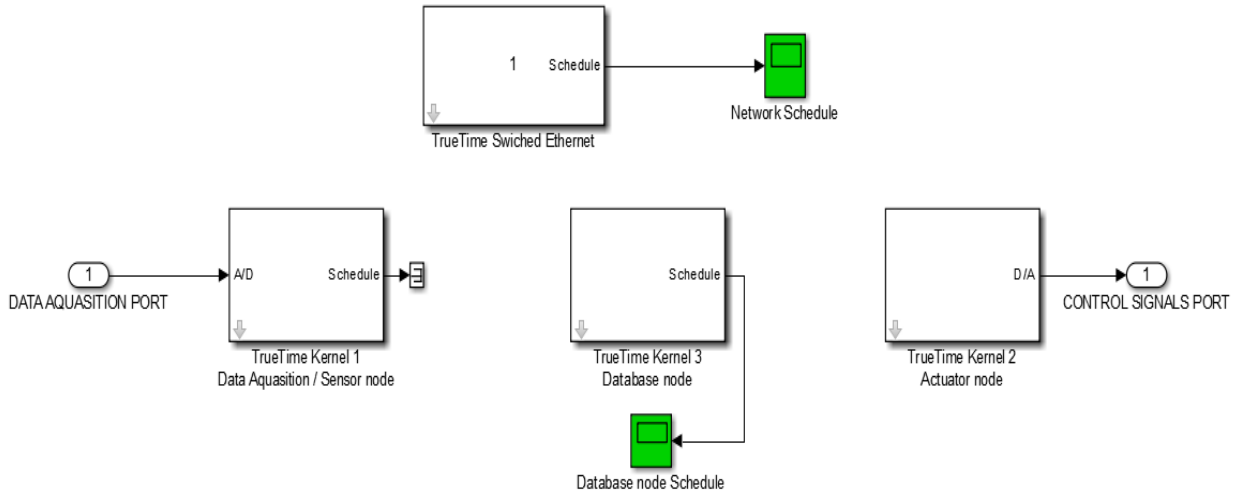


Fig.8.18: Inside internet subsystem.

While the used frequency for generator is 60 Hz, so using sampling time of 0.002s and less for data aqcuasition will be acceptable. In this simulation the used sampling time is 0.001s, which mean about 16 samples during each cycle of the 60 Hz signals.

Fig. 8.19, Fig. 8.20 and Fig. 8.21 show the stored data of stator voltage va, stator current ia and frequency respectivly, the results are shown before, during and after the fault.

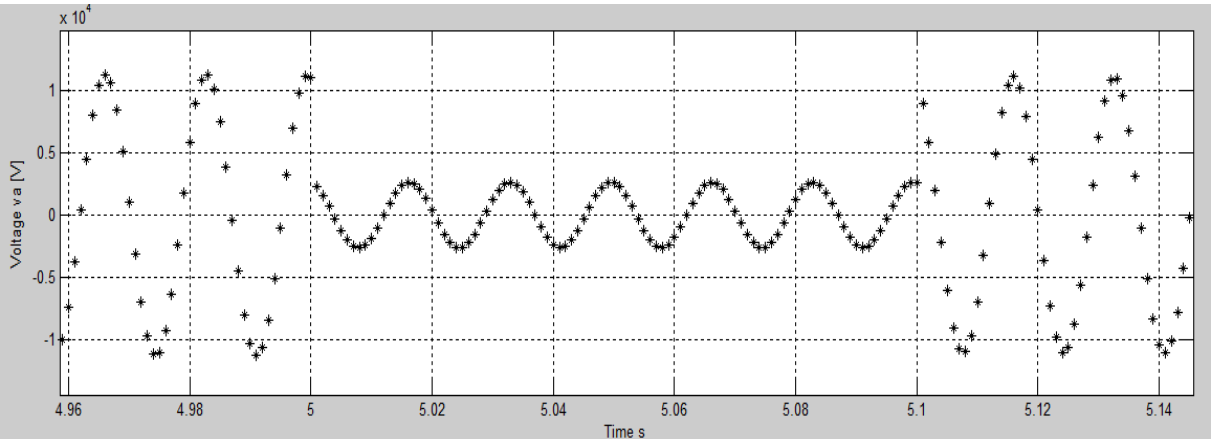


Fig. 8.19: Data stored in database of stator voltage va [V] before, during and after the fault

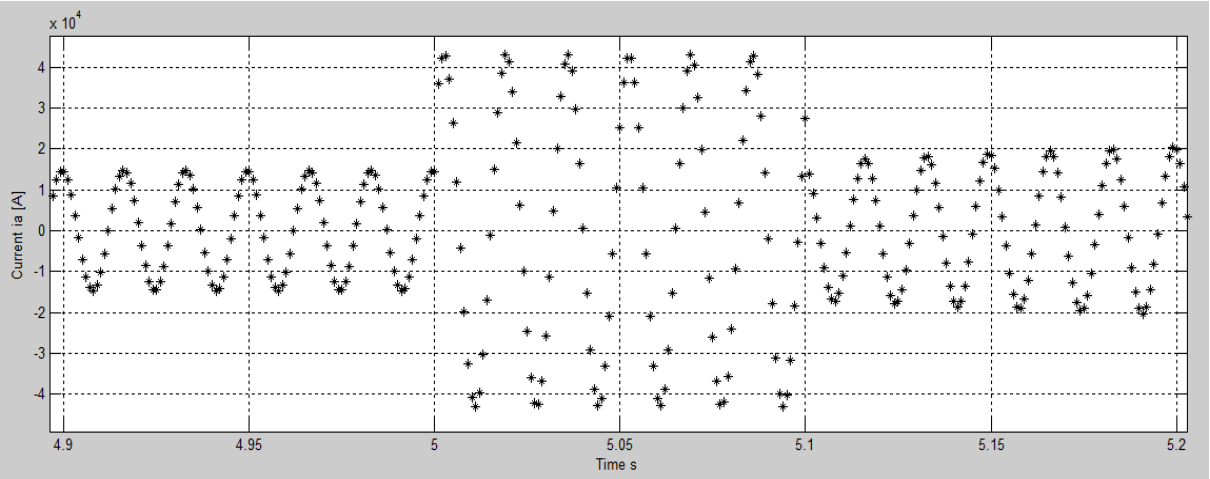


Fig. 8.20: Data stored in database of stator current ia [A] before, during and after the fault

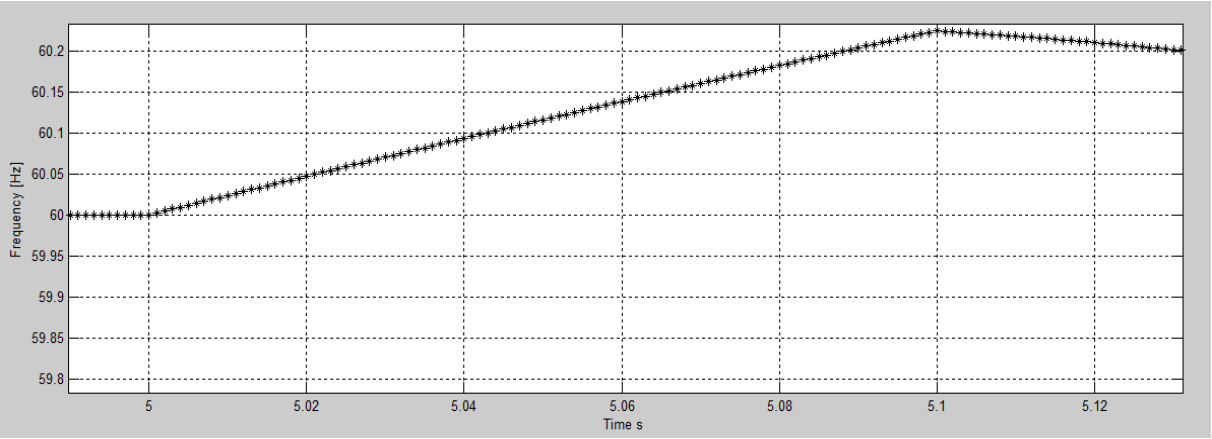


Fig. 8.21: Data stored in database of frequency [Hz] before, during and after the fault

8.4 Data aqcuisation and control system of power plant over internet network

Fig. 8.22 show over all diagram of the data acquisition and control system of plant. Two control processes were performed over the network.

First process is PID control of the gate servo motor of the turbine, the discrete PID control theory is explained in Appendix (B).

The second control process is the power system stabilizer PSS of the excitation system.

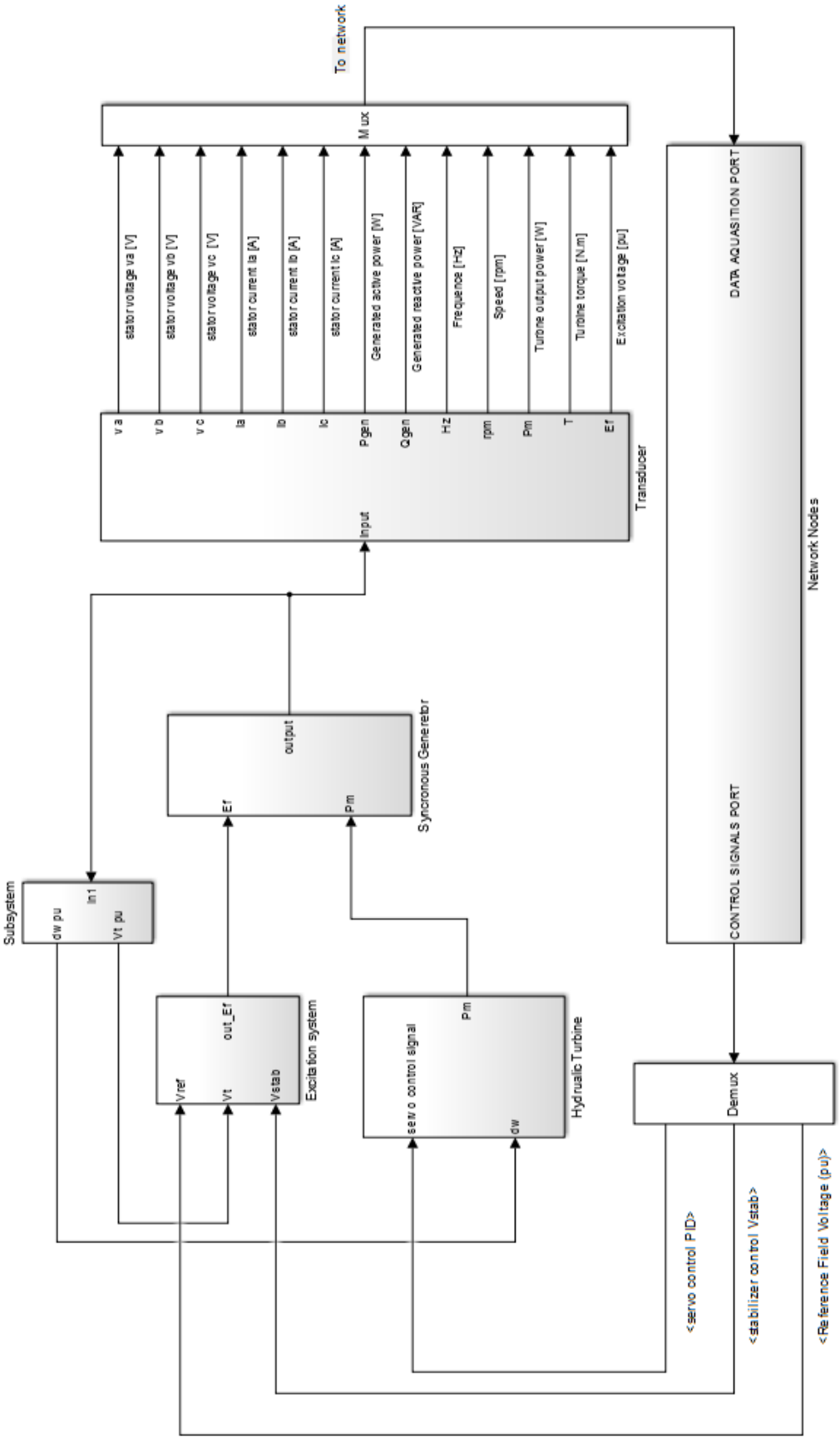


Fig. 8.22: Over all diagram of the data acqusition and control system of the plant

The control processes were performed in the new added node to the netwok, *Controller node*, which is illustrated in Fig. 8.23.

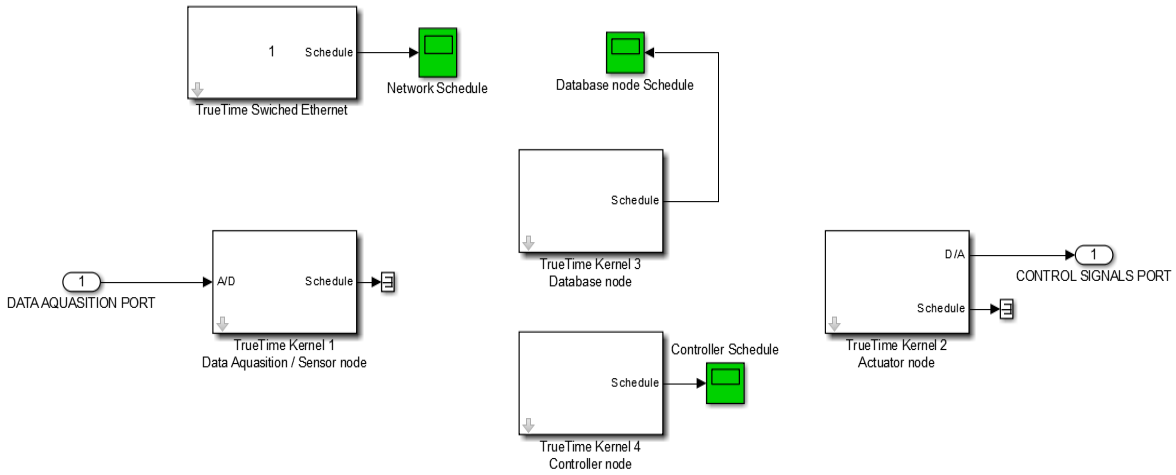


Fig. 8.23: Inside the network subsystem of the data acquisition and control system

Controller node (node 4) recieves the required data from data acquisition node (node 1) and perform the control processes then it sends the resulting control signals to actuator node (node 2) which, in turn, converts signals to analog form and sends it to the plant.

Fig. 8.24 and Fig. 8.25 show the network output control signals during and after the fualt, where Fig. 8.24 shows the network PID control signal of the hydrualic turbine gate servomotor, while Fig. 8.25 shows the network power system stabilizer control signal of the excitation system.

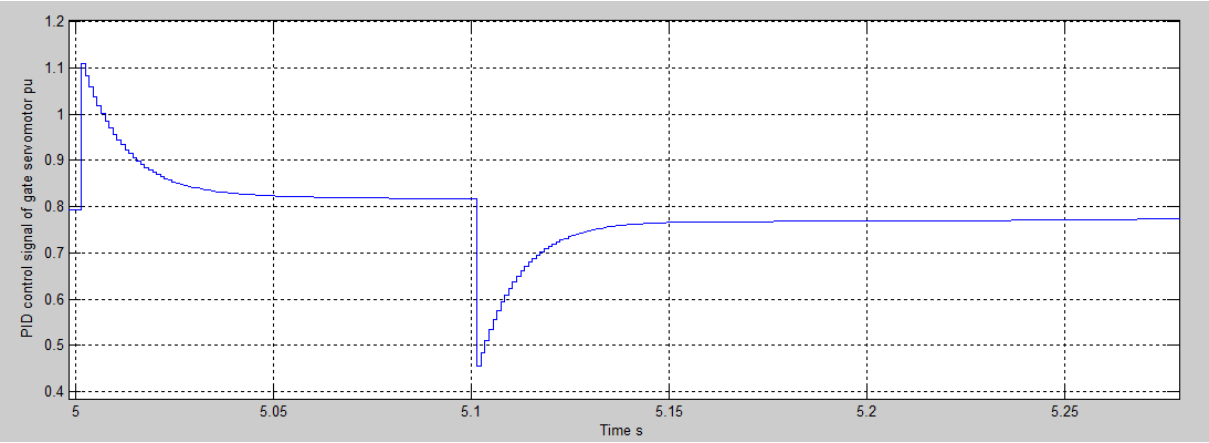


Fig. 8.24: Network PID control signal of gate servomotor of the turbine

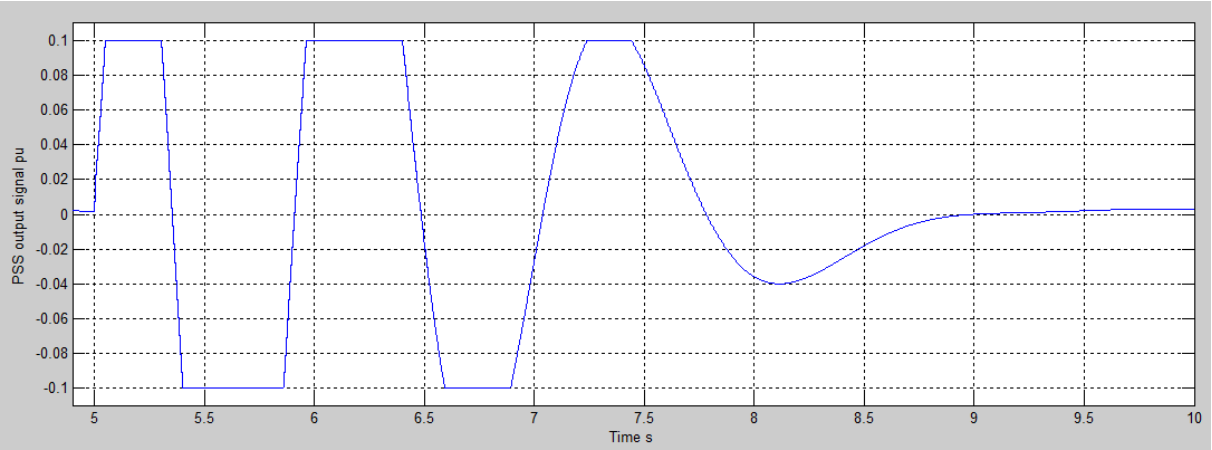


Fig. 8.25: Network PSS control signal

9 CONCLUSION AND FUTURE WORK

9.1 Conclusion

In this work, a data acquisition and control system of hydroelectric power plant using internet techniques model was developed. The modeled plant consists of hydraulic turbine and synchronous generator with excitation system.

Two synchronous generator models, steady state model and transient model, were simulated using qd rotating reference frame. The modeled generator is connected to infinite bus (infinte grid), Thevenin's equivalent circuit was considered to represent the infinite bus and generator connection. A transient stability study during three phase fault at generator terminals was performed, and the effect of using power system stabilizer on the system stability was discussed.

TRUETIME Matlab library was used to simulate the internet network and to build up a data acquisition and control system of the hydropower plant. TrueTime simulink library was developed by the Department of Automatic Control, Faculty of Engineering, Lund University in Sweden. Switched Ethernet network was used to acquire data from the plant and to send back the control signals.

Four simulation projects were applied; the first project is steady state model of synchronous generator. In this project, the results of steady state operation of synchronous generator during small changes of input power and excitation voltage were shown. The considered generator in first project is connected directly to infinite grid.

In the second project, the transient model of synchronous generator with excitation system model and hydraulic turbine model were applied. The simulated excitation system is DC exciter type excitation system with considering saturation characteristic. Non-linear hydraulic turbine model was applied with PID control of the gate servomotor. The generator transient status after implementing three phase fault on generator terminals was discussed in this project. Also the effect of using power system stabilizer on the system stability was discussed in this project.

In the third project, switched Ethernet network type of TrueTime library network models was used to build up a data acquisition system of the hydropower power plant system model developed and discussed by second project. Results of acquired transient status data during and after the fault when using sampling time of 0.001s of the DACs were shown. In this project three reference values can be adjusted via the network, these reference values are the reference speed value, reference power of the hydraulic turbine value and reference excitation voltage value.

In addition to adjust reference values in the fourth project, two control processes over the network were performed, the control processes are the governor control system of the hydraulic turbine process, and the power system stabilizer process of the excitation system. To perform the control processes, a controller node was added to the switched Ethernet network nodes. In this project, results of using networked control system to control the hydraulic turbine and the excitation system were shown.

9.2 Future Work

The aim of this thesis is to study the using of internet techniques and networked control in electrical power systems. The chosen part of the power system to study was a hydroelectric plant with synchronous generator, this part of the power system was favored because of the importance of green energy generated by hydraulic turbines around the world. So a simulation of hydroelectric plant with synchronous generator was performed.

Future work shall focus on reliability and stability of networked control systems, by studying the effects of network delay, data losing and control processing time on response of controlled system. Also a data analyzing system would be performed for maintenance schedules, fault prediction, faults diagnostics, design purposes and other needs. To study the synchronous generator transient response during big faults, a sub-transient model including damper winding representation would be developed.

APPENDIX (A) – SIMULATED SYSTEM PARAMETERS

The following tables contain values of all parameters used in simulation

Tab. A.1: Synchronous generator parameters used in steady-state simulation

Parameter	Value
S	900 MVA
V	18 kv
F	60 Hz
Poles	4
r_s	0.0032 pu
x_{ls}	0.093 pu
x_d	1.63 pu
x_q	1.56 pu
x'_d	0.174 pu
x''_d	0.123 pu
x''_q	0.124 pu
T'_{do}	4.3 s
T''_{do}	0.032 s
T''_d	0.023 s
T''_q	0.023 s
T''_{qo}	0.066 s

Tab. A.2: Synchronous generator parameters used in transient simulation

Parameter	Value
S	310 MVA
V	13.8 kv
F	60 Hz
Poles	56
r_s	0.008 pu
x_{ls}	0.11 pu
x_d	1.14 pu
x_q	0.63 pu
x'_d	0.33 pu
x'_q	0.63 pu
x''_d	0.25 pu
x''_q	0.33 pu
T'_{do}	6.6 s
T''_{do}	0.05 s
T'_{qo}	2.2 s
T''_{qo}	0.1 s

Tab. A.3: Excitation system parameters

Parameter	Value
T_A	0.06 s
K_A	50
T_E	0.052 s
K_E	-0.0465
K_F	0.0832
T_F	1 s
VRmax	1
VRmin	-1
A_{ex}	0.33 pu
B_{ex}	0.63 pu

Tab. A.4: Power system stabilizer parameters

Parameter	Value
T_A	0.06 s
K_A	50
T_E	0.052 s
K_E	-0.0465
K_F	0.0832
T_F	1 s
VRmax	1

Tab. A.5: Infinite bus, load and fualt parameters

Parameter	Value
Infinite bus impedance	0.008+j0.3 pu
Fault impedance	0.005+j0.05 pu
Load impedance	12-j6 pu

Tab. A.6: Turbine parameters

Parameter		Value
PID	Rp	0.05
	Kp	1.163
	Ki	0.105
	Kd	0.105
	td	0.01 s
Servo	ka	3.3
	ta	0.071 s
Turbine	Gmax	0.99
	Gmin	0.1
	beta	0
	Tw	2.7 s

Tab. A.7: Network parameters

Parameter	Value
Type	Swiched Ethernet
Data rate	10 Mbs
Minimum frame size	80 bits
Total switch memory	80000 bits
Switcg buffer type	Common buffer
Switch overflow behavior	Retransmit
Loss probability	0

APPENDIX (B) – DISCRETE PID CONTROLLER

B.1 Discrete PID method

If a digital implementation of the PID controller is adopted, then the control laws have to be discretised. This can be done with any of the available discretisation method (Åström and Wittenmark, 1997). For the sake of clarity and for future reference (see Chapter 8), an example is shown hereafter. Considering the continuous time expression of a PID controller in ideal form:

$$u(t) = K_p(e(t) + \frac{1}{T_i}\int_0^t e(\tau)d\tau + T_d\frac{de(t)}{dt}) \tag{B.1}$$

Where $u(t)$ is output signal, $e(t)$ is input signal, K_p is the PID proportional gain, T_i is the integral time constant and T_d is the devirative time constant.

Defining a sampling time Δt, The integral term in last equation can be approximated by using backward finite differences as:

$$\int_0^{t_k} e(\tau)d\tau = \sum_{i=1}^{k} e(t_i)\Delta t \tag{B.2}$$

Where $e(t_i)$ is the error of the continuous time system at the ith sampling instant. By applying the backward finite differences also to the derivative term it results:

$$\frac{de(t_k)}{dt} = \frac{e(t_k)-e(t_{k-1})}{\Delta t} \tag{B.3}$$

Then, the discrete time control law becomes:

$$u(t_k) = K_p\left(e(t_k) + \frac{\Delta t}{T_i}\sum_{i=1}^{k} e(t_i) + \frac{T_d}{\Delta t}(e(t_k) - e(t_{k-1}))\right) \tag{B.4}$$

In this way, the value of the control variable is determined directly. Alternatively, the control variable at time instant t_k can be calculated based on its value at the previous time instant $u(t_{k-1})$. By subtracting the expression of $u(t_{k-1})$ from that of $u(t_k)$, we obtain:

$$u(t_k) = u(t_{k-1}) +$$

$$K_p\left[\left(1 + \frac{\Delta t}{T_i} + \frac{T_d}{\Delta t}\right)e(t_k) + \left(-1 - \frac{2T_d}{\Delta t}\right)e(t_{k-1}) + \frac{T_d}{\Delta t}e(t_{k-2})\right] \tag{B.5}$$

For an obvious reason, the control algorithm (A.5) is called incremental algorithm or velocity algorithm, while that expressed in (A.4) is called positional algorithm.

Expression (1.37) can be rewritten more compactly as:

$$u(t_k) - u(t_{k-1}) = K_1e(t_k) + K_2e(t_{k-1}) + K_3e(t_{k-2}) \tag{B.6}$$

Where:

$$K_1 = K_p\left(1 + \frac{\Delta t}{T_i} + \frac{T_d}{\Delta t}\right) \quad \text{(B.7)}$$

$$K_2 = -K_p\left(1 + \frac{2T_d}{\Delta t}\right) \quad \text{(B.8)}$$

$$K_3 = K_p\frac{T_d}{\Delta t} \quad \text{(B.9)}$$

By defining z^{-1} as the backward shift operator, that means:

$$z^{-1}u(t_k) = u(t_{k-1}) \quad \text{(B.10)}$$

Then the discretised PID controller in velocity form can be expressed as:

$$C(z^{-1}) = \frac{K_1 + K_2 z^{-1} + K_3 z^{-1}}{1 - z^{-1}} \quad \text{(B.11)}$$

Where K_1, K_2 and K_3 can be viewed as the tuning parameters.

B.2 Typical modifications of the basic PID control law:

The expressions (A.1) and (A.4) of a PID controller given in the previous subsection are actually not adopted in practical cases because of a few problems that can be solved with suitable modifications of the basic control law. These are analysed in this susection.

B.2.1 Derivative Action modification

From Expressions (A.1) and (A.4) it appears that the controller transfer function is not proper and therefore it can not be implemented in practice. This problem is evidently caused by the derivative action. Indeed, the highfrequency gain of the pure derivative action is responsible for the amplification of the measurement noise in the manipulated variable. The amplification effect is more evident when the frequency of the noise is high. In practical cases, a (very) noisy control variable signal might cause a damage of the actuator. The problems outlined above can be solved by filtering the derivative action with (at least) a first-order low-pass filter. The filter time constant should be selected in order to filter suitably the noise and to avoid to influence significantly the dominant dynamics of the PID controller. Typically the filter time constant is $\frac{T_d}{N}$, where N generally assumes a value between 1 and 33, although in the majority of the practical cases its setting falls between 8 and 16.

Another issue related to the derivative action that has to be considered is the so-called derivative kick. In fact, when an abrupt (stepwise) change of the set-point signal occurs, the derivative action is very large and this results in a spike in the control variable signal, which is undesirable. A simple solution to avoid this problem is to apply the derivative term to the process output only instead of the control error. In this case the ideal (not filtered) derivative action becomes:

$$u(t) = -K_d\frac{dy(t)}{dt} \quad \text{(B.12)}$$

Where y(t) is the feedback signal and r(t) that is reference signal, that is:

$$e(t) = e(t) - y(t) \tag{B.13}$$

B.2.2 Set-point Weighting

A typical problem with the design of a feedback controller is to achieve at the same time a high performance both in the set-point following and in the load disturbance rejection performance. Roughly speaking, a fast load disturbance rejection is achieved with a high-gain controller, which gives an oscillatory set-point step response on the other side. This problem can be approached by designing a two-degree-of-freedom control architecture, namely, a combined feedforward/feedback control law.

In the context of PID control this can be achieved by weighting the set-point signal for the proportional action, that is, to define the proportional action as follows:

$$u(t) = K_p(\beta r(t) - y(t)) \tag{B.14}$$

Where the value of β is between 0 and 1.

B.2.3 General ISA–PID Control Law

If all the modifications of the basic control law previously addressed are considered, the following general PID control law can be derived:

$$u(t) = K_p\left(\beta r(t) - y(t) + \frac{1}{T_i}\int_0^t e(\tau)d\tau + T_d\left(\frac{d(\gamma r(t) - y_f(t)}{dt}\right)\right) \tag{B.15}$$

$$\frac{T_d}{N}\frac{dy_f(t)}{dt} = y(t) - y_f(t) \tag{B.16}$$

Where the value of γ is between 0 and 1. Its value is usually either 0 if the derivative action is entirely applied to the process output, or 1 if the derivative action is entirely applied to the control error.

APPENDIX (C) – TRUETIME LIBRARY COMMANDS

The available TRUETIME commands are summarized in the following tables. The commands are categorized according to their intended use (I; initialization script, T; task code function, and H; interrupt handler code function).

Tab. C.1 Commands used to create and initialize TRUETIME objects, and to control the simulation.

Command	Description
ttInitKernel	Initialize the kernel, specifying the scheduling policy.
ttGetInitArg	(C++ only) Retrieve the init argument from the block dialogue.
ttCreateTask	Create an aperiodic task.
ttCreatePeriodicTask	Create a periodic task.
ttCreateLog	Create a log structure and specify data to log.
ttCreateHandler	Create an interrupt handler.
ttCreateMonitor	Create a monitor (mutex) for protection of shared data.
ttCreateEvent	Create an event (condition variable).
ttCreateMailbox	Create a mailbox for inter-task communication.
ttCreateSempahore	Create a counting semaphore.
ttCreateCBS	Create a soft or hard constant bandwidth server.
ttNoSchedule	Switch off the schedule plot for a specific task or interrupt handler.
ttNonPreemptible	Make a task non-preemptible.
ttAttachTrigger	Handler Attach an interrupt handler to an external trigger.
ttAttachNetworkHandler	Attach an interrupt handler to a network interface.
ttAttachDLHandler	Attach a deadline overrun handler to a task.
ttAttachWCETHandler	Attach a worst-case execution time overrun interrupt handler to a task.
ttAttachHook	(C++ only) Attach a kernel scheduling hook to a task.
ttAttachCBS	Attach a task to a constant bandwidth server.
ttAbortSimulation	Abort the simulation.

Tab. C.2 Commands used to set and get task or kernel attributes

Command	Description
ttSetPeriod	Set the period of a periodic task.
ttSetDeadline	Set the relative deadline of a task.
ttSetPriority	Set the priority of a task.
ttSetWCET	Set the worst-case execution (maximum budget) time of a task.
ttSetData	Update the local memory data structure of a task.
ttSetAbsDeadline	Set the absolute deadline of the current job.
ttSetBudget	Set the execution time budget of the current job.
ttSetUserData	Set arbitrary kernel user data (C++ only).
ttGetPeriod	Get the period of a periodic task.
ttGetDeadline	Get the relative deadline of a task.
ttGetPriority	Get the priority of a task.
ttGetWCET	Get the worst-case execution time of a task.
ttGetData	Retrieve the local memory data structure of a task.
ttGetRelease	Get the release time of the current job.
ttGetAbsDeadline	Get the absolute deadline the current job.
ttGetBudget	Get the execution-time budget of the current job.
ttSetUserData	Set the kernel user data (C++ only).
ttGetUserData	Get the kernel user data (C++ only).
ttSetCBSParameters	Set the parameters of a constant bandwidth server

Tab C.3 Real-time primitives commands

Command	Description
ttCreateJob	Create a job of a task.
ttKillJob	Kill the running (ready) job of a task, if any.
ttEnterMonitor	Enter a monitor (blocking).
ttExitMonitor	Exit a monitor.
ttWait	Wait for an event (blocking).
ttNotify	Notify the highest-priority task waiting for an event.
ttNotifyAll	Notify all tasks waiting for an event.
ttLogStart	Start a timing measurement for a log.
ttLogStop	Stop a timing measurement and save in the log.
ttLogNow	Log the current time.
ttLogValue	Log a scalar value.
ttPost	Post a message to a mailbox (blocking).
ttFetch	Fetch a message from a mailbox (blocking).
ttRetrieve	Read the actual message fetched from a mailbox.
ttTake	Take a semaphore.
ttGive	Give a semaphore.
ttCreateTimer	Create a one-shot timer and associate an interrupt handler with the timer.
ttCreatePeriodicTimer	Create a periodic timer and associate an interrupt handler with the timer.
ttRemoveTimer	Remove a specific timer.
ttCurrentTime	Get and/or set the current time in the simulation on a per node basis.
ttSleepUntil	Sleep until a certain point in time.
ttSleep	Sleep for a certain amount of time.
ttAnalogIn	Read a value from an analog input channel.
ttAnalogOut	Write a value to an analog output channel.
ttSetNextSegment	Set the next segment to be executed in the code function (to implement loops and branches).
ttGetInvoker	Get the name of the external trigger, network interface, or overrun timer that triggered running handler.
ttCallBlockSystem	Call a Simulink block diagram from within a code function.
ttSendMsg	Send a message over a TRUETIME network (non-blocking).
ttGetMsg	Get a message that has been received over a TRUETIME network (non-blocking).
ttDiscardUnsentMessages	Delete any unsent messages.
ttSetNetworkParameter	Set a specific network parameter on a per node basis.
ttSetKernelParameter	Set a specific kernel parameter on a per node basis.

BIBLIOGRAPHY

[1] CHEE-MUN ONG. *Dynamic Simulation of Electric Machinery Using Matlab/Simulink*, Prentice hall PTR, 1998, ISBN: 0-13-723785-5.

[2] HASE, Y. *Handbook of Power System Engineering*, John Wiley & Sons, 2007. ISBN: 139978-0-470-02724-4 (HB)

[3] KLEMPNER, G., ISIDOR KERSZENBAUM I. *Handbook of Large Turbo-Generator Operation and Maintenance*, John Wiley & Sons, 2008. ISBN: 978-0470-16767-0.

[4] SKALICKÝ, J. *Electrické Servopohony*, Vysoké Učení Technické V Brně.

[5] SKALICKÝ, J.*Control Theory*, Faculty of Electrical Engineering and Communication, Brno University of Technology.

[6] Sattouf, M. Simulation Model of Hydro Power Plant Using Matlab/Simulink, *Int. Journal of Engineering Research and Applications*, ISSN : 2248-9622, Vol. 4, Issue 1(Version 2), January 2014, pp.295-301.

[7] GOPI, E.S. *Algorithm Collections for Digital Signal Processing Applications Using Matlab*, Springer, 2007. ISBN: 978-1-4020-6410-4.

[8] PAO, Y.C. *Engineering Analysis Interactive Methods and Programs with FORTRAN, QuickBASIC, MATLAB, and Mathematica*, CRC press LLC, 2001. ISBN: 0-8493-2016-X.

[9] KARRIS, S.T. *Introduction to Simulink with Engineering Applications*, Orchaed Publications, 2006. ISBN: 978-0-9744239-8-2.

[10] KARRIS, S.T. *Signals and Systems, second edition*, Orchaed Publications, 2003.ISBN: 0-9709511-8-3.

[11] KARRIS, S.T. *Signals and System with Matlab Computing and Simulink Modeling, third edition*, Orchaed Publications, 2007.ISBN: 0-9744239-9-8.

[12] SHAMPINE, L.F., GLADWELL, I. *Solving ODEs with Matlab, Instructor's Manual*, L.F. Shampine, I. Gladwell & S. Thompson, 2002.

[13] SHAMPINE, L.F., GLADWELL, I., THOMPSON, S. *Solving ODEs with Matlab,* L. F. Shampine, I. Gladwell, S. Thompson, 2003, Cambridge University Press. ISBN: 978-0-511-07707-4.

[14] WILSON, H.B., TURCOTTE, L.H., HALPERN, D. *Advanced Mathematics and Mechanics Applications Using Matlab, third edition,* Chapman &Hall/CRC, 2003. ISBN: 1-58488-262-X.

[15] HUNT, B.R., LIPSMAN, R.L., ROSENBERG, J.M., A Guide to MATLAB for Beginners and Experienced Users, B. Hunt, R. Lipsman, J. Rosenberg, K. Coombes, J. Osborn, G. Stuck, 2001, Cambridge University Press. ISBN: 978-0-511-07792-0.

[16] WANG, L., GUOLI, M., WANG, J. Implementing for Networked Motion Control System with Large-Capacity Data Acquisition, *Advanced in Control Engineering and Information Science*, Elsevier Ltd, 2011.Procedia Engineering 15 (2011) 28 – 32.

[17] VISIOLI, A. *Practical PID Control,* Springer-Verlag London Limited, 2006. ISBN: 1-84628-585-2.

[18] KUNDUR, P. *Power System Stability and Control*, McGraw-Hill, 1994. ISBN: 0-07-035958-X.

[19] TENORIO, L.A.L. *Hydro Turbine and Governor Modelling, Electric – Hydraulic Interaction*, Master of science thesis, Department of Electric Power Engineering, Norwegian University of Science and Technology, 2011.

[20] NAGHIZADEH, R.A., JAZEBI, Z., VAHIDI, B. Modeling Hydro Power Plants and Tuning Hydro Governors as an Educational Guideline, *International Review on Modelling and Simulations (I.RE.MO.S.),* Vol. 5, N. 4 pp. 1780-1790, August 2012. ISSN: 1974-9821 Praise Worthy Prize S.r.l., 2012.

[21] SINGH, G., CHAUHAN, D.S. Simplified Modeling of Hydraulic Governor-Turbine for Stable Operation under Operating Conditions, *International Journal of Engineering Research and Applications*, Vol. 1, Issue 3, pp.478-482. ISSN: 2248-9622.

[22] GRIGSBY, L.L. *Power System Stability and Control, Third Edition*, Taylor & Francis Group, LLC, 2012, CRC Press. ISBN: 978-1-4398-8321-1.

[23] MACHOWSKI, J., BIALEK, J.W., BUMBY, J.R. *Power System dynamics, Stability and Control, Second Edition*, John Wiley & Sons 2008. ISBN: 978-0-470-72558-0.

[24] Working Group on Prime Mover and Energy Supply Models for System Dynamic Performance Studies HYDRAULIC TURBINE AND TURBINE CONTROL MODELS FOR SYSTEM PYNAMIC STUDIES, *Transactions on Power Systems*, Vol. 7, NO. 1, pp. 167-179, February 1992. ISSN: 0885-8950. IEEE 1992.

[25] KORITAROV, V., GUZOWSKI, L., FELTES, J., KAZACHKOV, Y., LAM, B., GRANDE-MORAN, C., TROUILLE, B., DONALEK, P. *Review of Existing Hydroelectric Turbine-Governor Simulation Models*, Report prepared for U.S. Department of Energy – Wind and Water Power Technologies Office, 2013.

[26] KISHOR, N., SAINI, R.P., SINGH, S.P. A review on hydropower plant models and control, *Renewable and Sustainable Energy Reviews*, 11 (2007) 776–796, Elsevier 2006.

[27] BOLDEA, I. *The Electric Generators Handbook SYNCHRONOUS GENERATORS*, Taylor & Francis Group, LLC, 2006,. CRC Press.

[28] AWAD, M.L. *Modeling of Synchronous Machines for System Studies*, Doctoral Thesis, Department of Electrical and Cornputer Engineering, University of Toronto, 1999.

[29] SKALICKÝ, J. *Power electronics*, Brno University, 2002.

[30] LEONHARD, W. *Control of Electrical Drives, Second Edition*, Springer-Verlag Berlin Heidelberg, 1996. ISBN: 978-3-642-97648-3.

[31] ASTROM, K.J., WITTENMARK, B. *Computer-Controlled Systems Theory and Design, Third Edition,* Prentice Hall, 1997.

[32] HRISTU-VARSAKELIS, D., LEVINE, W. *Handbook of Networked and Embedded Control Systems*, Birkhauser Boston 2005. ISBN: 10 0-8176-3239-5.

[33] WANG, F.Y., LIU, D. *Networked Control Systems Theory and Applications*, Springer-Verlag London Limited 2008. ISBN: 9781848002142.

[34] DU, D.Z., RAGHAVENDRA,C. *Distributed Network Systems From Concepts to Implementations*, Springer Science + Business Media 2005. ISBN: 0-387-23840-9.

[35] AUSTERLITZ, H. *Data Acquisition Tecliniques Using PCs, Second Edition*, Elsevier Science (USA) 2003, 1991. ISBN: 0-12-068377-6.

[36] DORT, R.C., BISHOP, R.H. *Modern Control Systems Solution Manual, Eleventh Edition*, Pearson Education, Upper Saddle River 2008. ISBN: 9780132270298.

[37] Wostenkuhler, G. *Data Acquisition Systems (DAS) in General*, John Wiley & Sons, Ltd, 2005.

[38] MO MEASUREMENT COMPUTING. *Data Acquisition handbook, A Reference For DAQ And Analog & Digital Signal Conditioning, Third Edition,* Measurement Computing Corporation 2004-2012.

[39] KUAMR, s., MOHAMMAD, s. *PC Based Data Acquisition System*, Diploma Thesis, Electronics & Communication Department, National Institute of Technology, Rourkela, India.

[40] KALAITZAKIS, K., KOUTROULIS, E., VLACHOS, V. Development of a data acquisition system for remote monitoring of renewable energy systems, *Measurement* 34 (2003) pp.75–83. Elsevier 2003.

[41] NIEVA, T. *Remote Data Acquisition Of Embedded Systems Using Internet Technologies: A Role-Based Generic System Specification*, Doctoral Thesis N. 2388 (2001), Présentéé Au Departement D'informatique, École Polytechnique Fédérale De Lausanne, 2001.

[42] OGATA, K. *Discrete Time Control Systems, Second Edition*, Prentice Hall 1995. ISBN: 0133286428.

[43] IEEE-SA Standard Board, *IEEE Standard for Ethernet,* IEEE Std 802.3-2012.

[44] IEEE Computer Society, *IEEE Standard for Ethernet, Amendment 1: Physical Layer Specifications and Management Parameters for Extended Ethernet Passive Optical Networks,* IEEE Std 802.3bk-2013.

[45] SONG, E.Y., LEE, K. Understanding IEEE 1451 Networked Smart Transducer Interface Standard, *IEEE Instrumentation & Measurement Magazine,* April 2008, ISSN: 1094-6969. © iStockphoto.com.

[46] JOHNSON, R., LEE, K., WICZER, J., WOODS, S. *Smart Transducer Interface Standard - IEEE 1451*, Sensors Expo, Philadelphia, October 2, 2001.

[47] LEE, K. *IEEE 1451:* A Standard in Support of Smart Transducer Networking*, IEEE Instrumentation and Measurement,* Technology Conference, Baltimore, MD USA, May 1-4,2000, pp. 525-528. IEEE 2000. ISBN: 0-7803-5890-2.

[48] Burch,j. *Overview of IEEE 1451 and IEEE 1588 for Distributed Measurement and Control Applications*, Agilent Technologies.

[49] IEEE Standards Board, *IEEE Recommended Practice for Excitation System Models for Power System Stability Studies*, IEEE Std 421.5-1992.

[50] OONSIVILAI, A., MARUNGSRI, B. Optimal PID Tuning of Power System Stabilizer for Multi-machine Power System using Particle Swarm Optimization, *12th WSEAS International Conference on CIRCUITS*, Heraklion, Greece, July 22-24, 2008, pp.345-350. ISBN: 978-960-6766-82-4.

[51] ZEA, A. A. *Power System Stabilizers for The Synchronous Generator Tuning and Performance Evaluation*, Master of Science Thesis, Department Of Energy and Environment Division Of Electric Power Engineering Chalmers University Of Technology, Goteborg, Sweden 2013.

[52] KOTHARI, M. L., NANDA, J., BHATTACHARYA, K. Discrete mode power system stabilisers, *IEE PROCEEDINGS-C*. Vol. 140, No. 6, NOVEMBER I993, pp. 523-531. IEE 1993.

[53] IEEE Standards Board, American National Standards Institute, *IEEE Guide: Test Procedures for Synchronous Machines*, IEEE Std 115-1995(R2002), The Institute of Electrical and Electronics Engineers, Inc.1996.

[54] SWAIN, A. S. R., SWAIN, A. K., MOHAPATRA, A. *Design of Power System Stabilizer,* Diploma Thesis*,* Department of Electrical Engineering National Institute of Technology, Rourkela MAY 2012.

[55] RAJAA VIKHRAM, G. Y., LATHA, S. Design of Power System Stabilizer for Power System Damping Improvement with Multiple Design Requirements, *International Journal of Soft Computing and Engineering (IJSCE)*, Volume-2, Issue-5, November 2012, pp.43-48. ISSN: 2231-2307.

[56] IEEE Power Engineering Society, *IEEE Recommended Practice for Excitation System Models for Power System Stability Studies*, IEEE Std 421.5, 2005. The Institute of Electrical and Electronics Engineers, Inc. 2006.

[57] HAMMER, A. *Analysis of IEEE Power System Stabilizer Models,* Master of Science Thesis, Department of Electric Power Engineering, Norwegian University of Science and Technology, June 2011.

[58] CERVIN, A., HENRIKSSON, D., LINCOLN, B., EKER, J., ÅRZÉN, K. E. How Does Control Timing Affect Performance? Analysis and Simulation of Timing Using Jitterbug and TrueTime, *IEEE Control Systems Magazine*, June 2003, pp.16-30. ISSN: 0272-1708. IEEE 2003.

[59] PENG, D., ZHANG, H., HUANG, C., LIN, J., LI, H. Study of Immune PID Networked Control System Based on TrueTime, *Journal Of Networks*, VOL. 6, NO. 6, June 2011, pp.912-915. Academy Publisher 2011.

[60] CERVIN, A., HENRIKSSON, D., OHLIN, M. *TRUETIME 2.0 beta—Reference Manual*, Department of Automatic Control, Lund University, June 2010.

[61] CERVIN, A., HENRIKSSON, D., OHLIN, M. *TRUETIME 1.5 beta—Reference Manual*, Department of Automatic Control, Lund University, January, 2007.

[62] CHOO, Y. C., MUTTAQI, K. M., NEGNEVITSKY, M. Modeling Of Hydraulic Governor-Turbine For Contrlo Stabilisation, *ANZIAM J. 49 (EMAC2007)* pp. C681-C698, 2008.

[63] BERAN, J. Performance Analysis In Ip-Based Industrial Communication Networks, Doctoral Thesis, Department of Control and Instrumentation, Faculty of Electrical Engineering and Communication, Brno University of Technology, Brno 2010.

[64] SATTOUF, M. Simulation of Three Phase Synchronous Machine Using Matlab/Simulink, *XI. International Conference on Low Voltage Electrical Machines,* UVEE FEEC BUT. 2011. ISBN: 978-80-214-3975- 7.

[65] SATTOUF, M. Simulation of Three- phase Induction Motor Using MATLAB SIMULINK, *Low Voltage Electrical Machines 2010*. Šlapanice: UVEE FEEC BUT, 2010. ISBN: 978-80-214-4178- 1.

[66] SATTOUF, M. Direct Torque Control For Induction Motors Drives, *IEEE Workshop Králíky 2010*. kraliky, 2010. ISBN: 978-80-214-4139- 2.

[67] Demiroren, A., Zeynelgil, H. L. Modelling and simulation of synchronous machine transient analysis using SIMULINK, *International Journal of Electrical Engineering Education 39/4*, pp. 337-346.

[68] ZHANG, W. *Stability analysis of networked control systems*, Doctoral thesis, Case Western Reserve University, August, 2001.

[69] CERVIN, A., HENRIKSSON, D., ANDERSSON, M., ÅRZÉN, K. E. TRUETIME: Simulation Of Networked Computer Control Systems, *Preprints of the 2nd IFAC Conf. on Analysis and Design of Hybrid Systems* (Alghero, Italy), 7-9 June 2006, pp. 272-273.

[70] AUBRUN, C., SAUTER, D., YAMÉ, J. Fault Diagnosis of Networked Control Systems, *Int. J. Appl. Math. Comput. Sci.,* 2008, Vol. 18, No. 4, pp. 525–537.

[71] TrueTime: Simulation of Networked and Embedded Control Systems, Matlab/Simulink Library, Lund University, avialable at: http://www.control.lth.se/truetime/.

[72] ODOM, W. *Cisco CCENT/CCNA ICDN1 100-101 Official Cert Guide, Academic Edition*, Cisco Press, Pearson Education 2013. ISBN: 978-1-58714-485-1.