بِسۡمِ ٱللَّهِ ٱلرَّحۡمَٰنِ ٱلرَّحِيمِ

Sudan University of Science and Technology

College of Graduate Studies

Modeling of Hydraulic control System of Power Intake Gates in Rosiers Hydropower Plant

نمذجة نظام التحكم الهيدروليكي لبوابات مدخل التوربينات بمحطة توليد كهرباء الروصيرص

A thesis Submitted in Partial Fulfillment for the
Requirements for The degree of the Master of Science
In
Mechatronics Engineering

By:

OMER ABDALAZIZ ALHAJ JADALLAH

Supervisor:

Dr. OBAI YOUNIS TAHA

December 2015

بسم الله الرحمن الرحيم

Sudan University of Science & Technology

College of Graduate Studies

جامعة السودان للعلوم والتكنولوجيا

كلية الدراسات العليا

كلية الدراسات العليا

Approval Page

Name of Candidate: Omer Abdelaziz ElHaj

Thesis title: Modling of Hydralic Control System of power intake gates in Rosiers Hydropower Plant

Approved by:

1. External Examiner

Name: Dr. Ali MOHAMED ALI SEORY

Signature: .... Date: 14/12/2015

2. Internal Examiner

Name: Dr. Ali Mohammed Hamdan Adam

Signature: .... Date: 14/12/2015

3. Supervisor

Name: Abai Younis Taha

Signature: .... Date: 14/12/2015

Sudan University of Science and Technology

College of Graduate Studies

Declaration

I, the signing here-under, declare that I'm the sole author of the (M.Sc.) thesis entitled Modeling of Hydraulic Control System of Power Intake Gates in Rosiers Hydropower Plant

which is an original intellectual work. Willingly, I assign the copy-right of this work to the College of Graduate Studies (CGS), Sudan University of Science & Technology (SUST). Accordingly, SUST has all the rights to publish this work for scientific purposes.

Candidate's name: Omer Abdalaziz Alhaj Jadallah

Candidate's signature: .... **Date:** 23/03/2016

إقرار

أنا الموقع أدناه أُقر بأنني المؤلف الوحيد لرسالة الماجستير المعنونة نمذجة نظام التحكم الهيدروليكي لبوابات مدخل التوربينات بمحطة توليد كهرباء الروصيرص

وهى منتج فكري أصيل . وباختياري أعطى حقوق طبع ونشر هذا العمل لكلية الدراسات العليا - جامعه السودان للعلوم والتكنولوجيا، عليه يحق للجامعه نشر هذا العمل للأغراض العلمية .

اسم الدارس : عمر عبد العزيز الحاج جاد الله

توقيع الدارس : .... **التاريخ :** ٢٣/٠٣/٢٠١٦

DEDICATION

To my mother,

To my father,

To my wife,

To my little son, Mohammed

ACKNOWLEDGMENT

In the name of Allah, Most Gracious, Most Merciful

First and foremost, praises and thanks to Allah, the Almighty, on whom ultimately we depend for the completion of this master's thesis. Only due to His blessings I could finish this thesis.

I would like to express my deep and sincere gratitude to my supervisor, Dr. Obai Younis Taha for providing invaluable guidance throughout this study. Acknowledgement is due to Sudan University of Science and Technology for giving me the opportunity of studying Mechatronis Engineering which has really broadened my horizon and opened new chances for me.

I also acknowledge my colleagues in Rosiers Hydro Power Plant, specially Engineer Bukhari Abdullah, the manager of electrical department and Engineer Mohammed Saeed for their encouragement and support.

Also thanks to be delivered to Mr. Klaus Marggraff, sales manager of HYTEC Company, for his valuable support.

Finally, I extend my acknowledgement and heartfelt love to parents, brothers, sisters, uncles and special acknowledgment to my uncle, Alfadil, and all who have been with me all the time to lift my spirits.

ABSTRACT

Hydro Power Plants and Power Plants in general, consist of various components that are controlled by various control systems. These systems are subjected, from time to time, to some changes in the operational conditions and improvement processes; hence some modifications in the parameters of these systems may be needed in order to adapt to the new environment. Modeling and simulation of these systems is a good choice to test these modifications in the systems parameters. Simulation is less time and effort consuming and cheaper compared with experimental testing and prototyping.

The aim of this thesis is to develop a physical model for the hydraulic control system of power intake gates, which is a major component in a hydropower plant, in Rosiers Hydropower Plant. The hydraulic control system components are modeled and analyzed using the physical modeling toolboxes inside the commercially available simulation software, MATLAB/Simulink. In order to verify the developed model, system variables such as raising and lowering time, position of power intake gate and oil pressure are measured on the real system and then compared with the simulation results. Also a graphical user interface was developed to ease interacting with the model and displaying the simulation results.

Simulation results showed very good agreement with the measurements. Furthermore, by the help of Graphical User Interface, effect of varying intake gate weight on raising and lowering time is investigated and analyzed.

The developed model can be used as based model to which a mechanical model of the power intake gate and physical model of electrical components such as motors can be integrated for deeper investigation behind the hydraulic system model. In addition, if a modification in the existing system is to be made, the effect can be tested over the model in advance.

مستخلص

تحتوى محطات التوليد المائى للكهرباء على عدة مكونات رئيسية يتم التحكم فيها باستخدام انظمة تحكم متعددة. هذه الانظمة قد تتعرض من فترة الى اخرى الى بعض التغييرات فى الظروف التشغيلية و بعض عمليات التحسين و هذا يتطلب تغيير بعض معاملات هذه الانظمة لأجل التكيف مع البيئة الجديدة. نمذجة و محاكاة هذه الأنظمة تعتبر خيارا جيدا لإختبار هذه التعديلات فى معاملات هذه الأنظمة وذلك كسبا للوقت و الجهد كما أن المحاكاة تعتبر و سيلة أرخص.

يهدف هذا البحث الى تطوير نموذج فيزيائى لنظام التحكم الهيدروليكى لبوابات مداخل التوربينات - و التى تعتبر مكونا رئيسيا فى محطات التوليد المائى للكهرباء- بمحطة توليد كهرباء الروصيرص. تم نمذجة وتحليل مكونات نظام التحكم الهيدروليكى باستخدام برنامج المحاكاة المتاح ماتلاب. و لأجل التحقق من النموذج المطور تم قياس المتغيرات فى النظام الحقيقى كزمن الرفع و النزول و ضغط الزيت ثم قورنت هذه القياسات مع نتائج المحاكاة. كما تم ايضا تطوير واجهة مستخدم رسومية لتسهيل التعامل مع النموذج وعرض نتائج المحاكاة.

أظهرت نتائج المحاكاة توافقا جيدا مع القياسات و اضافة الى ذلك، و بمساعدة واجهة المستخدم الرسومية، تم دراسة و تحليل أثر تغيير وزن بوابة مدخل التوربينة على زمن الرفع و النزول للبوابة.

النموذج المطور يمكن أن يستخدم كنموذج اساسي و الذى يمكن ان يضاف اليه نموذج ميكانيكى لبوابة مدخل التوربينة و نموذج فيزيائى لمكونات النظام الكهربائي كالموتورات و ذلك لدراسة ابعد و اعمق من نموذج النظام الهيدروليكى. اضافة الى ذلك اذا كان هنالك حوجة الى عمل تعديل فى النظام الموجود فإنه يمكن دراست اثر هذا التعديل مسبقا فى النموذج المطور.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES

LIST OF TABLES

TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

SYMBOLS

h_{AT} Orifice opening for the orifice_A_T block

h_{PA0} Initial opening for the orifice_P_A block

h_{PA} Orifice opening for the orifice_P_A block

h_{AT0} Initial opening for the orifice_A_T block

k Valve gain coefficient

k_{HP} Hagen-Poiseuille coefficient

k_{leak} Leakage coefficient

k_p Pilot ratio, k_p = AX / AA

K_p Penetration coefficient

q Flow rate

q_A Flow rate through port A into the cylinder

q_B Flow rate through port B from the cylinder

x_0 Initial distance between piston and cap A

x Piston position

x_E Distance the piston can travel to fully extend from initial position

x_R Distance the piston can travel to fully retract from initial position

p_{cr} Minimum pressure for turbulent flow

p_{crack} Valve cracking pressure

p_{max} Pressure needed to fully open the valve

p_{nom} Pump nominal pressure

A_A Area of the spool in the A chamber

A_B Area of the spool in the B chamber

A_{leak} Closed valve leakage area

A_{max} Fully open valve passage area

A_X Area of the pilot chamber

A(p) Instantaneous orifice passage area

C_D Flow discharge coefficient

D Pump displacement

D_H Instantaneous orifice hydraulic diameter

F Force developed by the cylinder

F_c Hard stop force

P Pressure differential

P_A Pressure at the cylinder port A

P_B Pressure at the cylinder port B

P_A, P_B Gauge pressures at the block terminals

P_e Equivalent pressure differential across the control member

P_P,P_T Gauge pressures at the block terminals

P_X Gauge pressure at the pilot terminal

Re_{cr} Critical Reynolds number

S Piston stroke

T Torque at the pump driving shaft

V Cylinder rod velocity

V_R, V_C Absolute velocities of cylinder rod and cylinder case, respectively

X Control member displacement from initial position

η_{mech} Pump mechanical efficiency

η_V Pump volumetric efficiency

ν Fluid kinematic viscosity

ν_{nom} Nominal fluid kinematic viscosity

ρ Fluid density

ρ_{nom} Nominal fluid density

ω Pump angular velocity

ω_{nom} Pump nominal angular velocity

ABBREVIATIONS

CCR : Central Control Room

DAC : Double Acting Cylinder

DC : Direct Current

DCV : Directional Control Valve

EXP : Experimental

FCV : Flow Control Valve

GUI : Graphical User Interface

HIL : Hardware-in-the loop

HMI : Human Machine Interface

ICG : Intake Control Gate

PLC : Programmable Logic Controller

PRV : Pressure Relief Valve

PS : Physical Signal

SAC : Single Acting Cylinder

SIM : Simulation

SOE : Sequence Of Events

UI : User Interface

CHAPTER ONE

INTRODUCTION AND LITERATURE REVIEW

1.1. General

Rosiers Hydropower Plant is hydroelectric dam on the Blue Nile, located in Ad Damazine, just upstream of the town of Roseirs in Blue Nile State in Sudan. Construction of the power plant was started after the construction of the dam body was finished in 1966. The power plant consists of 7 power generation units, with 40 Mw capacity of each and total installed capacity of 280 Mw (see figure 1.1).

The installation of the power generation units was implemented through four stages.

Stage one: Units 1 and 2 were installed in the year 1971 and unit 3 in the year 1972.

Stage two: Unit 4 was installed in the year 1979.

Stage three: Units 5 and 6 were installed in the year 1984.

Stage four: Unit 7 was installed in the year 1989.

Figure 1.1: 7 Generation units in Rosiers Hydropower Plant

Rosiers Hydropower plant consists of many control systems that are used to control the various components of power plant units.

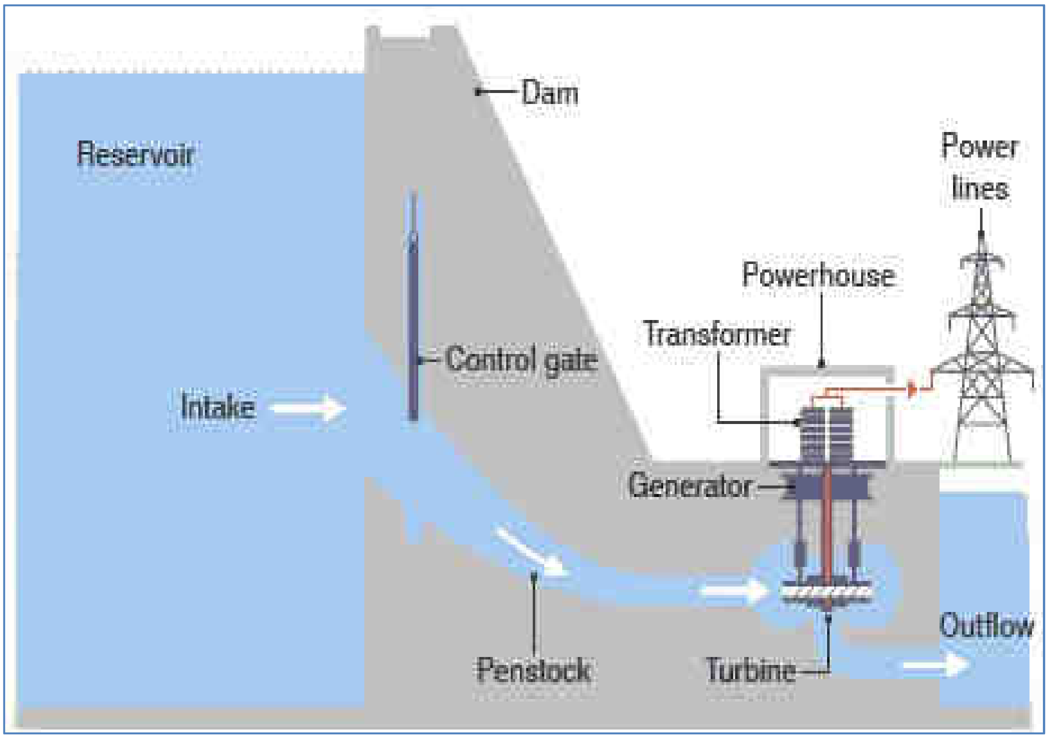


Figure 1.2: A typical Hydropower Plant

One of these components is the intake control gate or power intake gate. It is regarded as one of the important components in every hydropower plant. Hydro power plants use power intake gates on the dam to tap the water supply to meet power demand. As the gates open, the water flows down a pipe (penstock) to turbines inside the powerhouse (see figure 1.2).

Power Intake Gate is used also for the protection of the turbine and generator in case of over speed. So when the turbine speed reaches a certain limit (230 rpm), the intake gates must emergency be lowered down so as to prevent the turbine over speed. Turbine over speed event can lead to catastrophic failures that may generate unanticipated, uncontained liberation of turbine blades and components and may lead to severe damage in the penstock and resulting in water flooding the power house.

Each power generation unit, in Rosiers Hydropower Plant, has a power intake gate that allows water flowing to the turbine so as to generate electrical power. The seven intake gates are arranged side by side at a distance of 18 m.

Raising and lowering of the gates is carried out through a hydraulic control system. A new hydraulic control system, for power intake gates, was installed in December 2013 instead of the old one. This rehabilitation project was implemented among other rehabilitation projects so as to extend the life time of the power plant.

1.2. Motivation

The new Hydraulic control system, of power intake gates, is like a black box. So it is difficult to assess its current performance and also it is not allowed to make any modification in the parameters of the hydraulic system components, otherwise the warranty terms and conditions will be void.

Also in the future a new power intake gates will be installed and there may be a need for readjusting of the parameters of some hydraulic system components.

1.3. Scope of work

This study covered the development of model of the hydraulic system of power intake gates in Rosiers Hydropower Plant.

In addition to that, effects changes in any parameter of hydraulic systems components or in the gate weight can be analyzed in a more cost-saving and faster manner with the help of this model.

The purpose of this project is to create a physical model of a hydraulic control system, of power intake gates, and to evaluate the results from the simulation. The results will be confirmed by physical readings from the real system.

1.4. Literature Review

Hydraulics control systems are widely used and a well-known control systems. So in general it is easy to find research articles, notes and books about this subject. But for Power Intake gates specifically, it is difficult to find a typical model in hydro power plant and there may be no one found according to my humble survey.

Hydraulic control system of power intake gates doesn't differ too much, regarding the components and principle of work, from the most common hydraulic systems used in the control of hydraulic cranes, loaders and lifts, except that intake gates are relatively large loads. In intake gates the weights are counted in terms of tons while in loaders it is in terms of Kilograms.

So a number of studies on the performance of hydraulic control systems and its components have been made in recent past. Some of the relevant and significant studies related with the present work are discussed here.

Per-Willy Lauvli and Bjorn Victor Lund [1] presented in a study the results of modeling and simulation of a physical hydrostatic transmission with three different modeling tools; Simulink, SimHydraulics and SimulationX. The aim has been to get the simulations from the different models to be as similar as possible to the two measured pressures and the rotational speed of the load. The SimulationX model gave the best results compared with the measurements. The largest challenge has been to simulate the model in Simulink and to find the frictional losses in the hydraulic motor by performing different tests. The solver in Simulink could not solve the equations and it was difficult to find the tests for finding two of the friction parameters.

Boran Kilic [2] developed a dynamic model of the loader system of a backhoe-loader. Rigid bodies and joints in the loader mechanism and loader hydraulic system components are modeled and analyzed in the same environment using the physical modeling toolboxes, SimHydraulics and SimMechanics, available inside the commercially available simulation software, MATLAB/Simulink. Interaction between the bodies and response of the hydraulic system are obtained by co-operating the mechanical and hydraulic analyses. System variables such as pressure, flow and displacement are measured on a physical machine and then compared with the simulation results. Simulation results are consistent with the measurement results. The main result of this work is the ability to determine the dynamic loads on the joints and attachments of the backhoe-loader. In addition to that, prototyping time and costs can be highly reduced by implementing this model in the design process.

Weinan Cao et.al.[3] analyzed the necessity of research on Model and Simulation of ocean wave power generation platform of hydraulic lifting system. Aiming to the working condition and mechanism of self-elevating power-generating platform, the paper provided overall designing scheme and analyzed the working principle of the system .By modeling and simulation of hydraulic system based on AMESim software to set different parameter values for analysis of dynamic characteristics and stability of the system, which can help to find out the factors that influence the dynamic characteristics. This founding has a certain significant guiding meaning for the parameter optimization design of the jack up platform of hydraulic lifting system.

CHAPTER TWO

DESCRIPTION OF THE SYSTEM

The controlled movement of parts or a controlled application of force is a common requirement in the industries. These operations are performed mainly by using electrical machines or diesel, petrol and steam engines as a prime mover. These prime movers can provide various movements to the objects by using some mechanical attachments like screw jack, lever, rack and pinions etc. However, these are not the only prime movers. The enclosed fluids (liquids and gases) can also be used as prime movers to provide controlled motion and force to the objects or substances. The specially designed enclosed fluid systems can provide both linear as well as rotary motion. The high magnitude controlled force can also be applied by using these systems. This kind of enclosed fluid based systems using pressurized incompressible liquids as transmission media are called as hydraulic systems [4].

Fluid possesses a certain mechanical energy, if moved; the total energy of the fluid in motion consists of three parts:

- Potential energy (gravitational energy) which is a function of the height of the fluid column.
- Kinetic energy (hydrodynamic energy) which is a function of fluid velocity.
- Pressure energy (hydrostatic energy) which is a function of pressure.

The potential energy can be neglected completely in hydraulic systems since hydraulic circuits are never constructed with any great height. The kinetic energy is likewise negligible since the volume of fluid moving through the relatively thin pipes is small and the velocity is only a few meters per second or less. So the energy of a hydraulic fluid is derived mainly from its pressure which is the third part [5]. So a hydraulic system may be called as hydrostatic system.

A hydrostatic system uses fluid pressure to transmit power. Hydrostatics deals with the mechanics of still fluids and uses the theory of equilibrium conditions in fluid. The system creates high pressure, and through a transmission line and a control element, this pressure drives an actuator (linear or rotational). The pump used in hydrostatic systems is a positive displacement pump. The relative spatial position of this pump is arbitrary but should not be very large due to losses (must be less than 50 m). An example of pure hydrostatics is the transfer of force in hydraulics [6].

The hydrostatic system works on the principle of Pascal's law which says that the pressure in an enclosed fluid is uniform in all the directions. Pascal's law is illustrated in figure 2.1. The force given by fluid is given by the multiplication of pressure and area of cross section. As the pressure is same in all the direction, the smaller piston feels a smaller force and a large piston feels a large force. Therefore, a large force can be generated with smaller force input by using hydraulic systems [6].

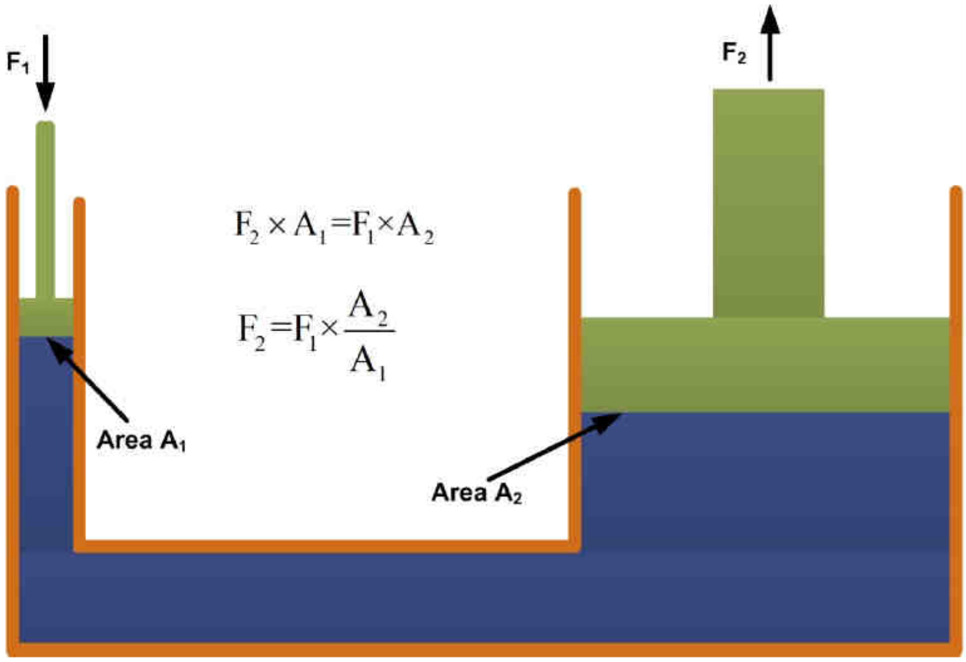


Figure 2.1: Principle of hydrostatic system

Hydrodynamic systems use fluid motion to transmit power. Power is transmitted by the kinetic energy of the fluid. Hydrodynamics deals with the mechanics of moving fluid and uses flow theory. The pump used in hydrodynamic systems is a non-positive displacement pump. The relative spatial position of the prime mover (e.g., turbine) is fixed. An example of pure hydrodynamics is the conversion of flow energy in turbines in hydroelectric power plants.

In oil hydraulics, we deal mostly with the fluid working in a confined system, that is, a hydrostatic system [6].

2.1. Basic Hydraulic System components

In hydraulic control systems, electrical energy or thermal energy is used to drive an electrical motor or combustion engine and then mechanical energy is generated.

This mechanical energy is then converted into hydraulic energy by means of hydraulic pump, processed in open or closed loops to drive linear or rotary actuators (cylinders or motors) by converting it back into mechanical energy to do the required work (see figure 2.2) .

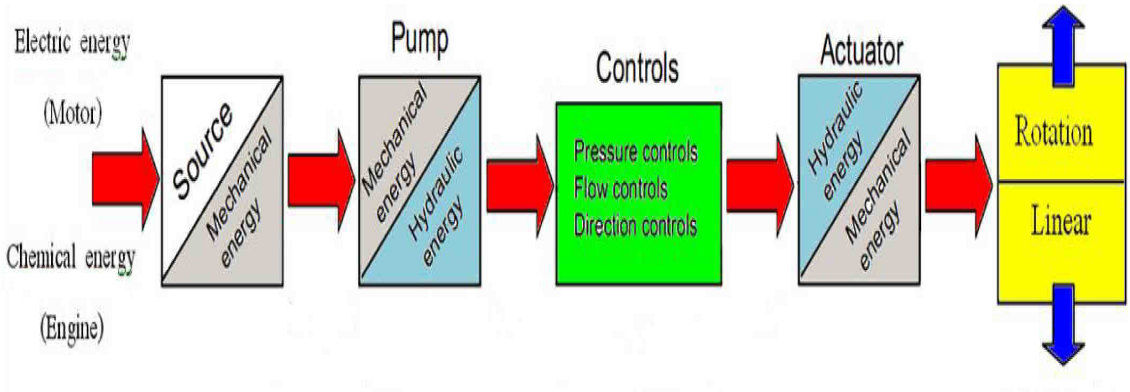


Figure 2.2: Energy conversion in hydraulic system

As an example figure 2.3 shows a simple circuit of a hydraulic system with basic components.

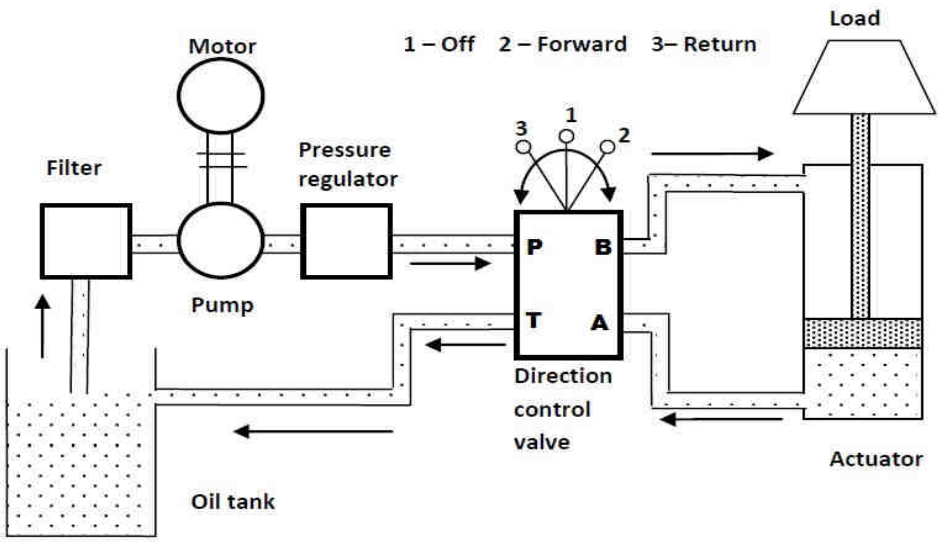


Figure 2.3: Simple hydraulic system

Functions of the components in the hydraulic circuit are as follows:

1. External power supply (motor) is used to provide the pump with the required mechanical power.
2. The hydraulic pump is used to force the fluid from the reservoir to the rest of the hydraulic circuit by converting mechanical energy into hydraulic energy.
3. Reservoir is used to hold the hydraulic liquid, usually hydraulic oil.
4. Piping system carries the hydraulic oil through various components of the hydraulic system.
5. Filters are used to remove any foreign particles so as keep the fluid system clean and efficient, as well as avoid damage to the actuator and valves.
6. Pressure regulator regulates (i.e., maintains) the required level of pressure in the hydraulic fluid.
7. Valves are used to control the direction, pressure and flow rate of a fluid flowing through the circuit.
8. The hydraulic actuator is a device used to convert the fluid power into mechanical power to do useful work. The actuator may be of the linear type (e.g., hydraulic cylinder) or rotary type (e.g., hydraulic motor) to provide linear or rotary motion, respectively.

Cylinder movement is controlled by a three-position change over a control valve.

1. When the piston of the valve is changed to upper position, the pressure port P is connected to port A and tank port T is connected to port B to allow oil in the upper side of the piston to return back to the tank, thus the load is raised.
2. When the position of the valve is changed to lower position, the pressure port P is connected to port B and tank port T is connected to port A to allow oil in the upper side of the piston to return back to the tank, thus the load is lowered.
3. When the valve is at center position, it locks the fluid into the cylinder, thereby holding it in position.

   In industry, a machine designer conveys the design of hydraulic systems using a circuit diagram. Figure 2.4 shows the components of the hydraulic system using symbols. The working fluid, which is the hydraulic oil, is stored in a reservoir. When the electric motor is switched ON, it runs a positive displacement pump that draws hydraulic oil through a filter and delivers at high pressure. The pressurized oil passes through the regulating valve and does work on actuator.

Oil from the other end of the actuator goes back to the tank via return line. To and fro motion of the cylinder is controlled using directional control valve [6].

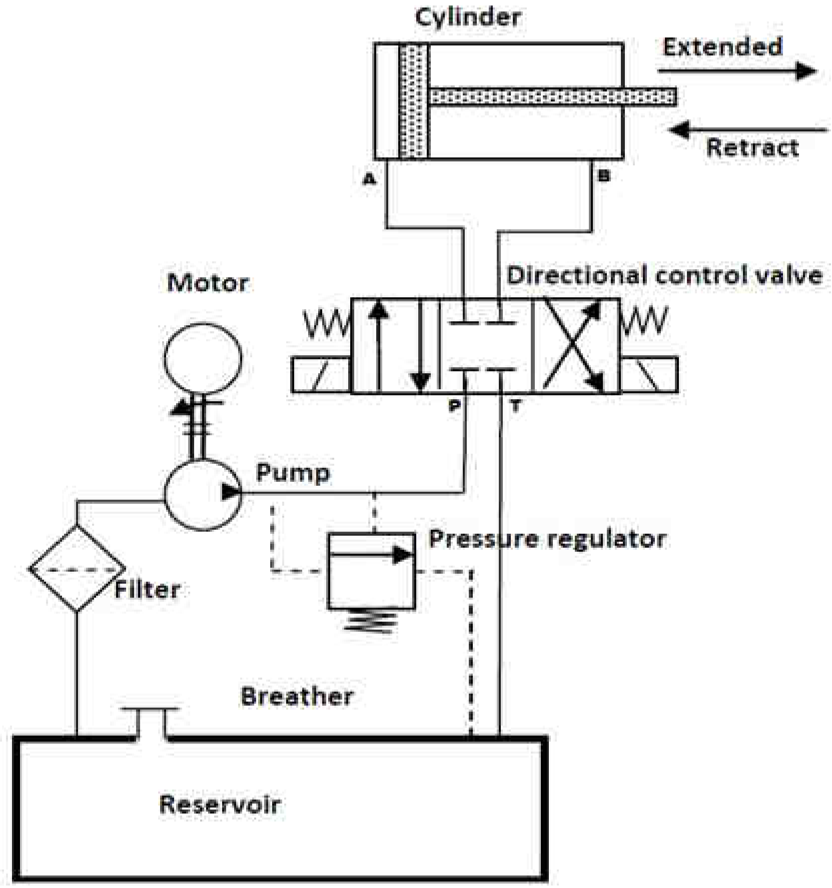


Figure 2.4: Hydraulic system components shown in symbols

The hydraulic system discussed above can be broken down into four main divisions that are analogous to the four main divisions in an electrical system.

1. The power device parallels the electrical generating station.
2. The control valves parallel the switches, resistors, timers, pressure switches, relays, etc.
3. The pipes in which the fluid power flows parallel the electrical lines.
4. The fluid power actuator (whether it is a linear or rotating, cylinder or hydraulic motor) parallels electrical motors and the solenoids.

So in general, hydraulic control system consists of the following components shown in block diagram as illustrated in figure 2.5:

1. The drive block which consists of a source of mechanical energy (electrical motors, combustion engines) connected to a Hydraulic pump
2. Control block which consists of valves so as to control:

- Direction and it is called directional control valves.
- Pressure and it is called pressure control valves.
- Flow and it is called flow control valves.

3. The output block or Actuators to do the useful work, whether it is linear (cylinder) or rotary (motor).
4. The machine block represents the load or the mechanical work to be done.

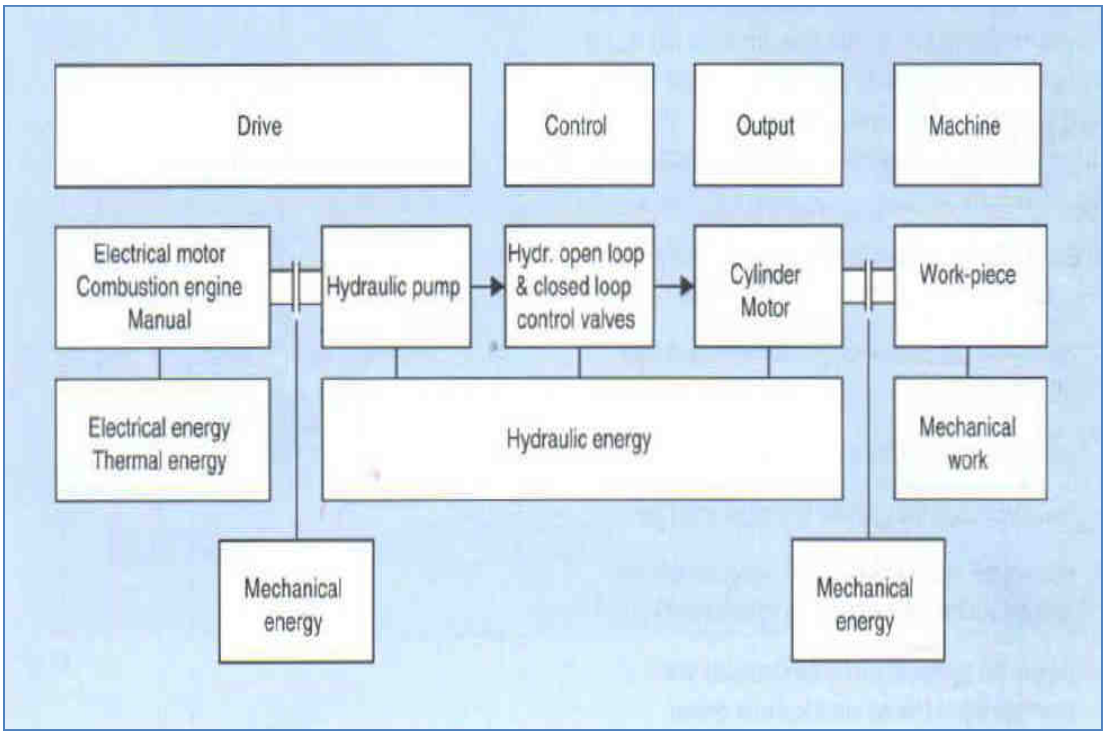


Figure2.5: Block diagram of hydraulic system [7].

2.2. Control System Description

Roseirs Hydro Power Plant consists of seven power generation units. Each unit has its own power intake gate that allows water flowing to the turbine so as to generate power. The seven intake gates are arranged side by side at a distance of 18 m.

The control system of the power intake gates consists of two parts:

1. The electrical control system.
2. The hydraulic control system.

The electrical control system is used to send commands to the hydraulic control system and receive feedback signals from it. The hydraulic control system is the target of modeling and simulation in this study.

2.2.1.Description The electrical control system

The electrical control system of power intake gate is a local control panel consists of a PLC controller from Siemens Company, touch panel used as (HMI) Human Machine Interface and push buttons for local operation (see figure 2.6 and 2.7). The electrical control system is linked also to Power Plant control via hardwired signals for remote operation from CCR (see figures 2.8 and 2.9) [8].

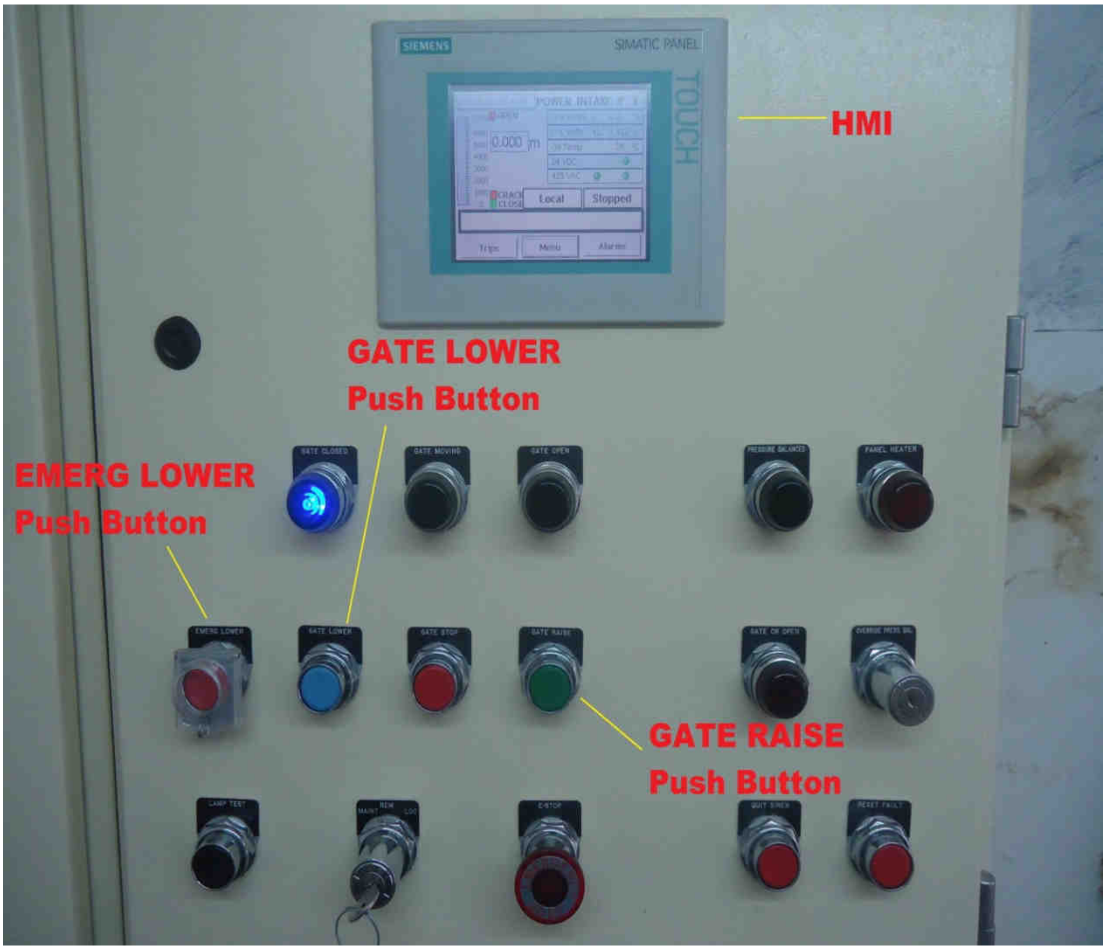


Figure 2.6: Local Control Panel

Figure 2.7: Inside Local Control Panel

Figure 2.8: CCR in Rosiers Hydropower plant

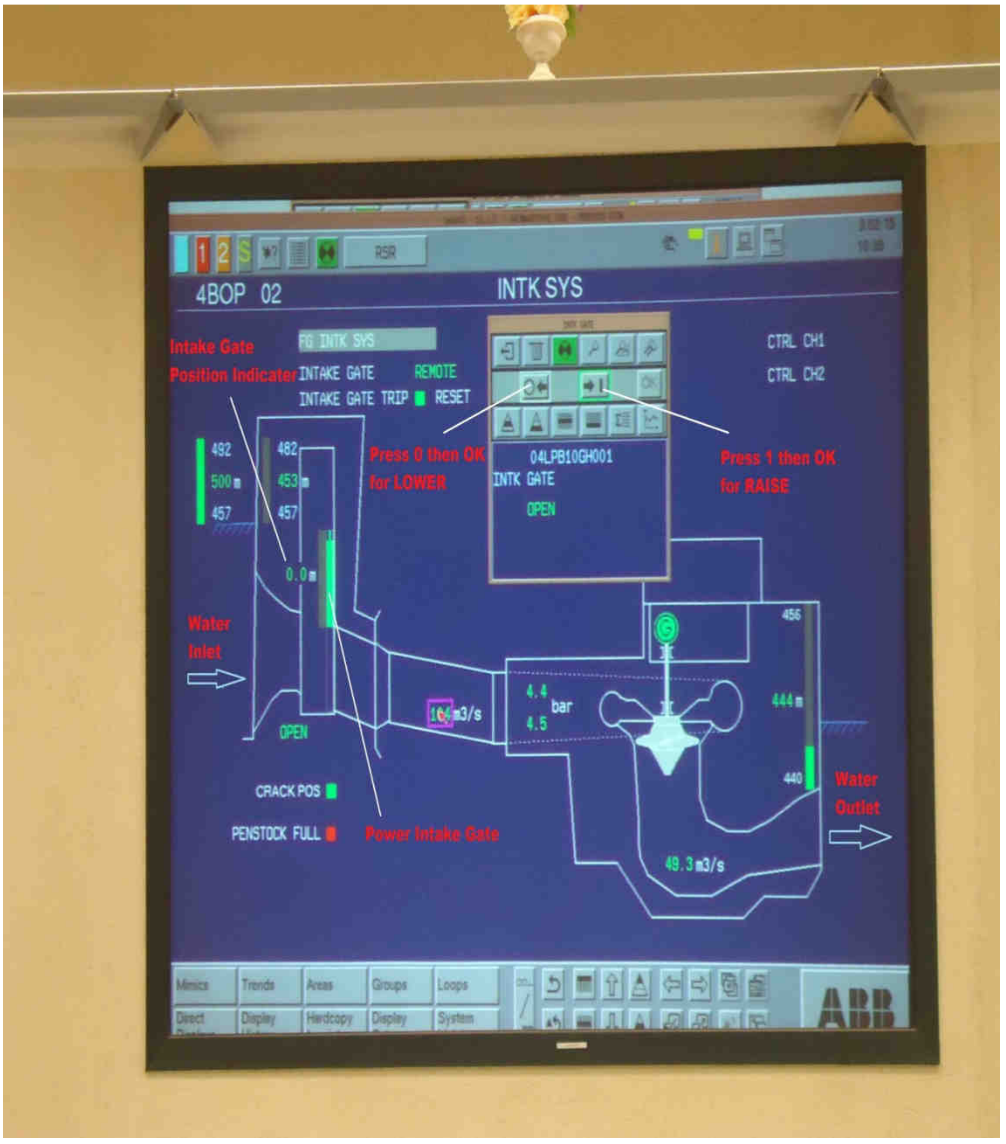


Figure 2.9: Remote operation of Intake Gate from CCR

2.2.2.Description of the hydraulic control system

Raising and lowering of the gates is carried out through a hydraulic control system. The intake gates are divided into two different groups, based on the weight of the gates, raising and lowering time.

Group 1 consists of the gates for units from 1 to 3.

Group 2 consists of the gates for units from 4 to 5.

Main parameters of the gates:

1. Gates from 1 to 3:

   Weight of the gate:100 tons.

2. Gates from 4 to 7:

   Weight of the gate: 36 tons.

Clear opening for all intake gates:

Height:7.10 m.

Width: 5.80 m.

The hydraulic control system consists, as mentioned earlier, of the drive element, control element and actuator.

The drive element in hydraulic control system of power intake gates consists of the electrical motor and hydraulic pump.

The control element consists of the check valves, pilot operated check valves, directional control valves, pressure relief valves and flow control valves.

The actuator type used is a linear actuator which is a double acting cylinder.

So a brief description of the above mentioned and other components in the hydraulic circuit diagram in figure 2.10 will be introduced.

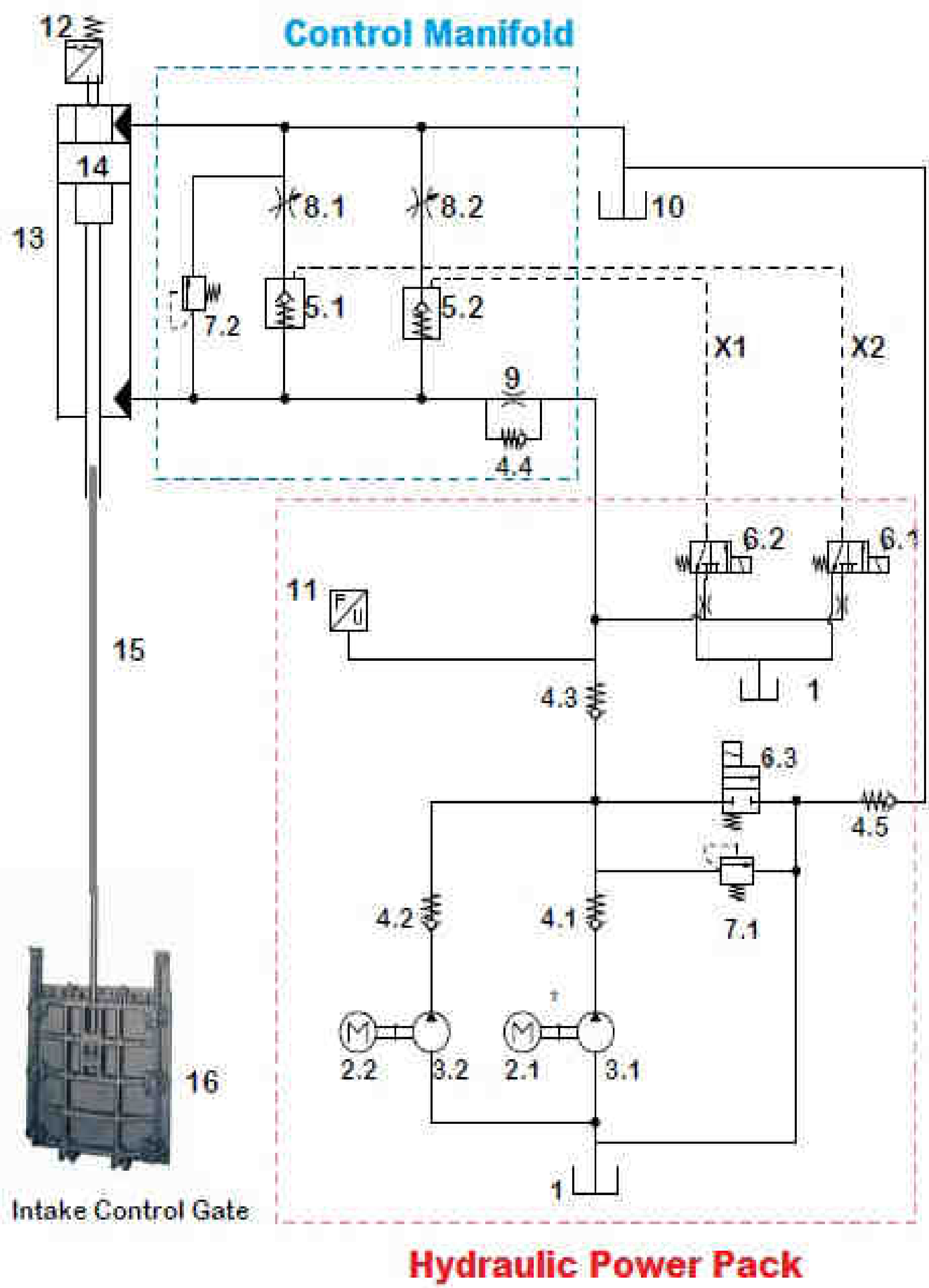


Figure 2.10: Hydraulic circuit diagram of Rosiers power intake gates

Table 2.1: Description of items in the hydraulic circuit diagram

Item No.	Quantity	Description
1	1	Main Tank
2	2	Electrical motor
3	2	Hydraulic pump
4	5	Check valve or Non Return Valve
5	2	Pilot Operated Check Valve
6	3	Directional Control Valve, solenoid operated
7	2	Pressure relief valve (PRV)
8	2	Flow Control Valve
9	1	Fixed Orifice
10	1	Overhead Tank
11	1	Pressure sensor
12	1	Position sensor
13	1	Double Acting Cylinder (DAC)
14	1	Cylinder piston
15	1	Interconnecting rod
16	1	Intake gate

The hydraulic system components are mounted on two locations as in figure 2.10:

- The hydraulic power pack.
- The control manifold mounted on the cylinder bottom.

2.2.2.1. Main oil Tank

It is used to supply the hydraulic system with the required quantity of oil. Also the hydraulic tank performs important functions such:

- Dissipating heat through its walls.
- Conditioning the fluid by helping settle the contaminants.
- Relief of air from the systems.
- Providing mounting support for the hydraulic pumps, electric motors and other components (see figure 2.11).

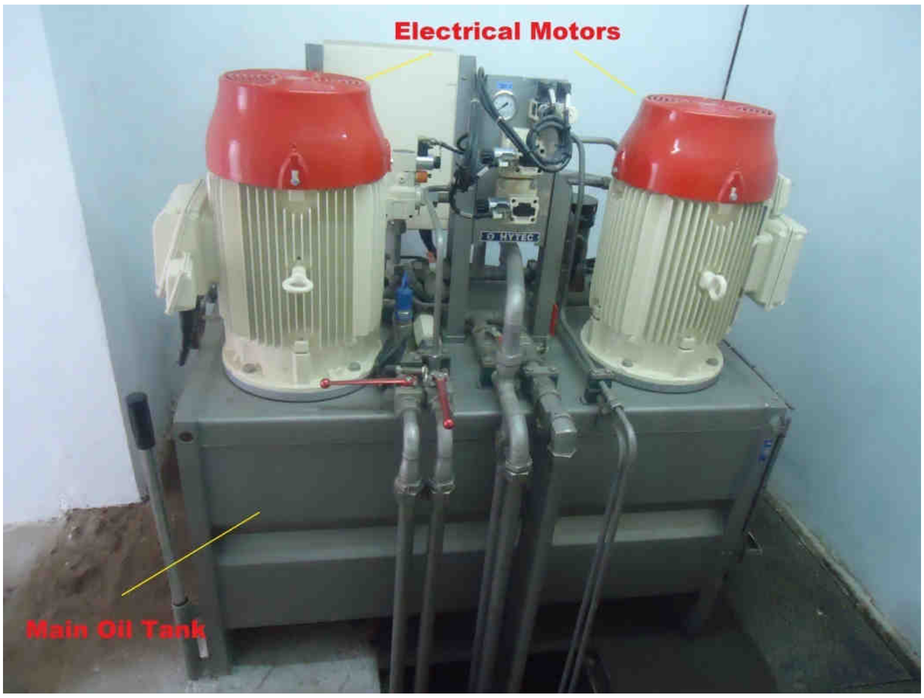


Figure 2.11: Hydraulic Power Pack

2.2.2.2. Electrical motor

It is used to provide the hydraulic pump with the required mechanical energy (see figure 2.11).

2.2.2.3. Hydraulic pump

A positive displacement hydraulic pump is a device used for converting mechanical energy into hydraulic energy. It is driven by a prime mover such as an electric motor. It basically performs two functions. First, it creates a partial vacuum at the pump inlet port. This vacuum enables atmospheric pressure to force the fluid from the reservoir into the pump. Second, the mechanical action of the pump traps this fluid within the pumping cavities transports it through the pump and forces it into the hydraulic system. It is important to note that pumps create flow but not pressure. Pressure is created by the resistance of flow.

The type of positive displacement pump used in this application is the internal gear pump, which is illustrated in figure 2.12. They consist of two gears: An external gear and an internal gear. The crescent placed in between these acts as a seal between the suction and discharge (see figure 2.12). When a pump operates, the external gear drives the internal gear and both gears rotate in the same direction. The fluid fills the cavities formed by the rotating teeth and the stationary crescent. Both the gears transport the fluid through the pump. The crescent seals the low-pressure pump inlet from the high-pressure pump outlet. The fluid volume is directly proportional to the degree of separation and these units may be reversed without difficulty. The major use for this type of pump occurs when a through shaft is necessary, as in an automatic transmission. These pumps have a higher pressure capability than external gear pumps [6].

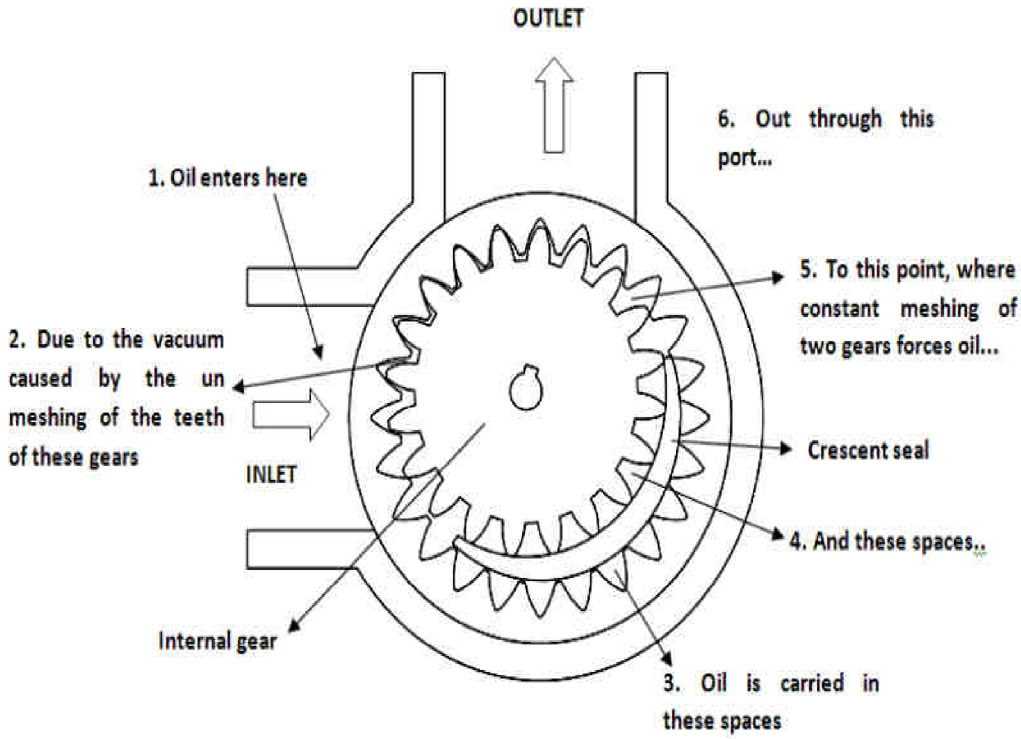


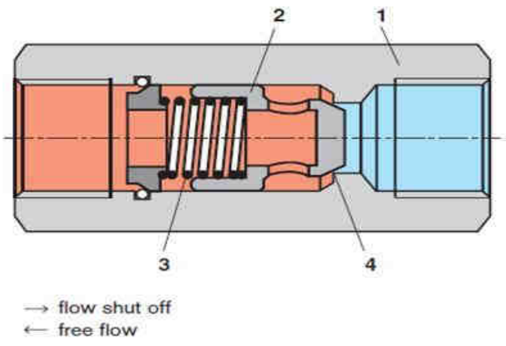
Figure 2.12: Internal Gear pump

The electrical analogy of hydraulic pump, regarding its output, is a source of DC or constant oil flow.

2.2.2.4. Non return valve

Non return valve is regarded as a directional control valve. It is used in hydraulic systems to stop flow in one direction and allow flow in the opposite direction. They are known also as check valves (see figure: 2.13).

These valves basically comprise housing (1) and hardened piston (2), which is pushed onto the seal seat (4) by means of spring (3) [7].



(a) Cross section

(b) In the real system

Figure 2.13: Check valve or non-return valve

The check valve can be viewed as a hydraulic version of a diode in an electronic circuit, which allow the flow of the current in a direction and block the flow in the other [9].

2.2.2.5. Pilot operated check valve

As opposed to the simple non return valve, pilot operated check valve can be opened in the direction of closure; these valves are used for example:

- To isolate working circuits under pressure
- To prevent the load from dropping, if a line should break.
- To prevent creep movements of hydraulically loaded actuators.

  As shown in figure 2.14, the free flow is from A to B in the normal operation, so fluid pressure acts on area A1of the main poppet (1) and lifts it from its seat against spring force (3). No flow is allowed in the direction from B to A, which corresponds with the function of the normal check valve (see figure 2.14) [7].

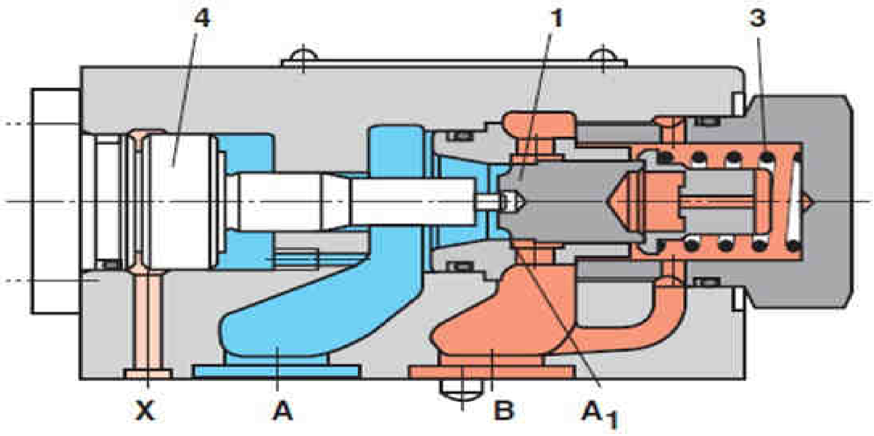


Figure 2.14: pilot operated check valve cross section

When a pilot pressure is applied at port X on piston (4), the piston (4) is pushed to the right and thus opened the main poppet (1) once a specified pressure has been reached.

Also the electrical analogy of pilot operated check does exist. It can be viewed as a hydraulic transistor which has a piston controlled by a low-current signal, moves a poppet which affects the current through another section of pipe.

2.2.2.6. Directional control valve

One of the most important considerations in any fluid power system is control. If control components are not properly selected, the entire system does not function as required. In fluid power, controlling elements are called valves.

There are three types of valves and one of them is the directional control valves.

A valve is a device that receives an external signal (mechanical, fluid pilot signal, electrical or electronics) to release, stop or redirect the fluid that flows through it. The function of a DCV is to control the direction of fluid flow in any hydraulic system. A DCV does this by changing the position of internal movable parts. To be more specific, a DCV is mainly required for the following purposes:

- To start, stop, accelerate, decelerate and change the direction of motion of a hydraulic actuator.
- To permit the free flow from the pump to the reservoir at low pressure when the pump's delivery is not needed into the system.
- To vent the relief valve by either electrical or mechanical control.
- To isolate certain branch of a circuit [5].

The designation of directional control valve refers to the number of service ports (not including control ports) and the number of spool position.

3/2 way Directional Control Valve: this type of DCV with control element directly operated by mechanical acting device.

In the initial position the seat element which is a ball (1) is pushed to the left onto the seat (3) by spring (2) (see figure: 2.15) [7].

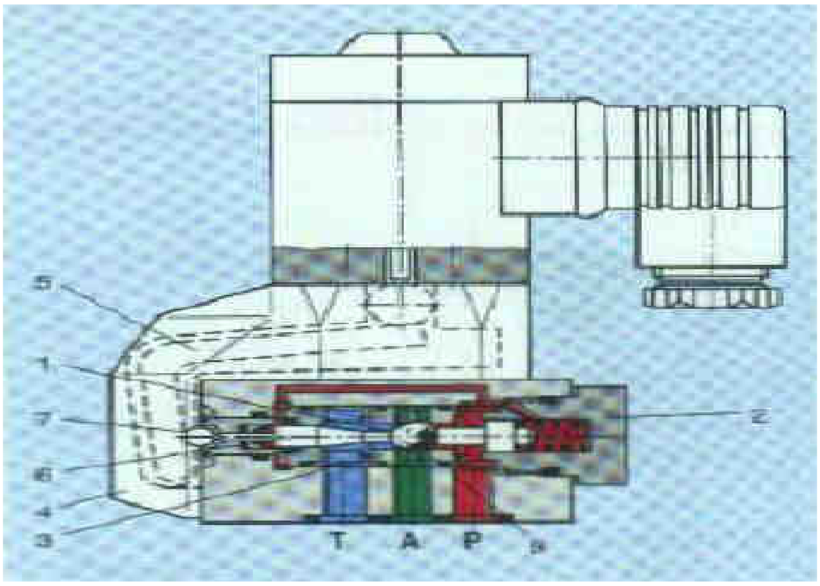
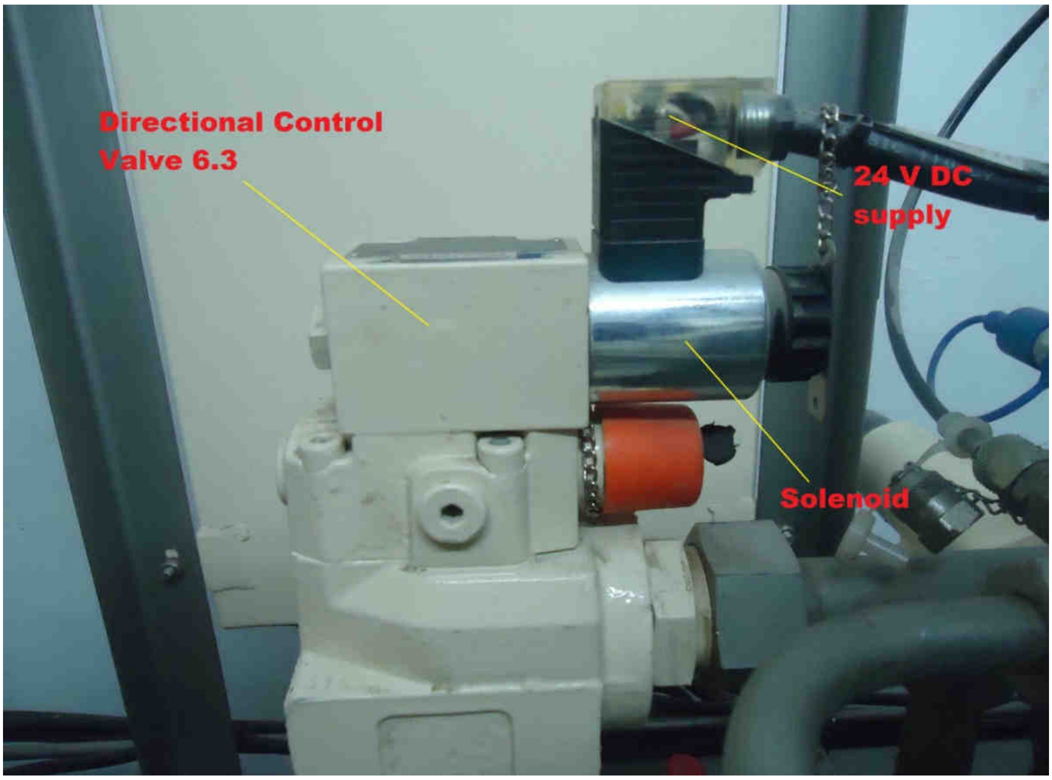
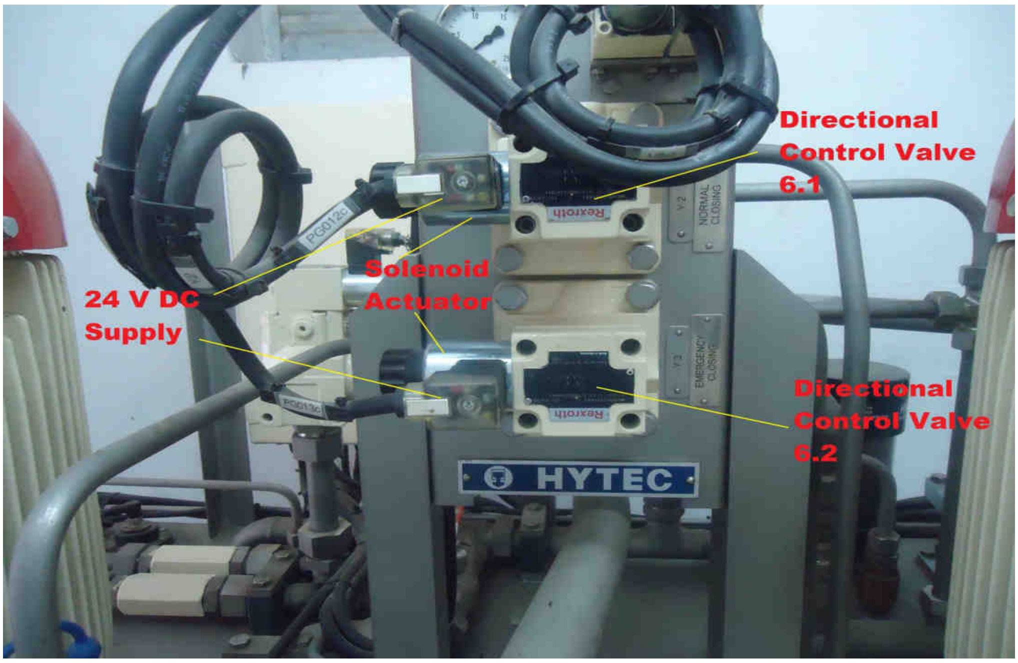


Figure 2.15: Cross section for directional control valve

There are three directional control valves, for raising, normal lowering and emergency lowering (see figure 2.16).



(a)DCV for Raising



(b) DCV for lowering

Figure 2.16: Directional control valvesin the real system

In the initial position, the connection from P to A is opened, port T is closed. The spool position of the valve is changed by force from solenoid supplied by 24 V dc (electrical signal). The force affects the seat element (1) by means of a lever (5) supported by bearings, ball (7) and operating plunger (6) in the housing (4). The ball is pushed to the right against spring (2) and pushed on its seat (8). Port P is now closed and the connection from A to T is opened [7].

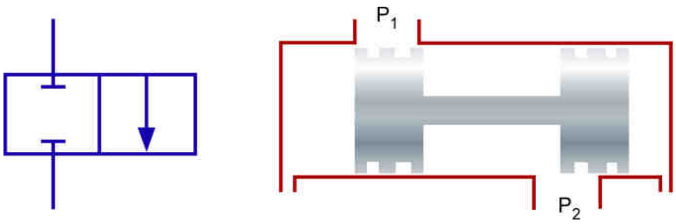


Figure 2.17: 2/2 directional control valve symbols

The electrical version of the directional control valve is simply electric switch which controls the flow direction of the current in electric circuits.

2.2.2.7. Pressure relief valve

Pressure relief valve is a type of pressure control valves. It is used in hydraulic systems to limit the system pressure to a specific level, if this preset level is reached, the pressure relief valve is activated and feeds the excess oil (difference between pump and actuator flow) from the system back to the tank (see figure: 2.18) [7].

There are two PRVs in the system, one is found on hydraulic power pack and the other on the control manifold (see figure: 2.22)

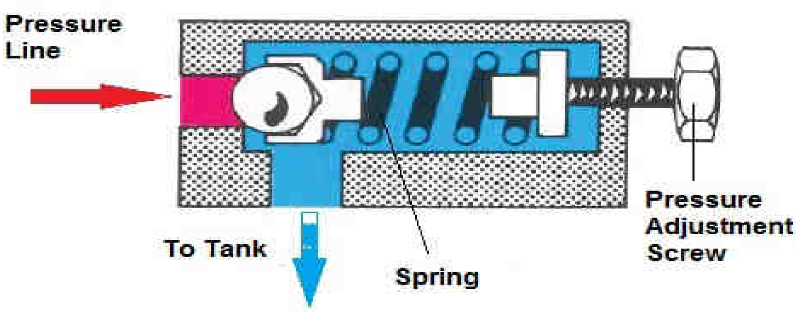


Figure 2.18: Pressure relief valve

A relief valve is similar to a fuse in an electrical system. When circuit amperage stays below the fuse amperage, all is well. When circuit amperage tries to exceed fuse amperage, the fuse blows and disables the circuit. Both devices protect the system from excess pressure/current by keeping it below a preset level. The difference is that when an electrical fuse blows, it must be reset or replaced by maintenance personnel before the machine cycles again. This requirement alerts electrician's about a possible problem before restarting the machine. Without the protection of a fuse, the electrical circuit would finally overheat and start a fire [6].

2.2.2.8. Flow Control Valve

In practice, the speed of actuator is very important in terms of the desired output and needs to be controlled. The speed of actuator can be controlled by regulating the fluid flow. A FCV can regulate the flow or pressure of the fluid (see figure 2.19) [7].

There are two FCVs, one for the normal lowering and the other for emergency lowering, located on the control manifold (see figure 2.22).

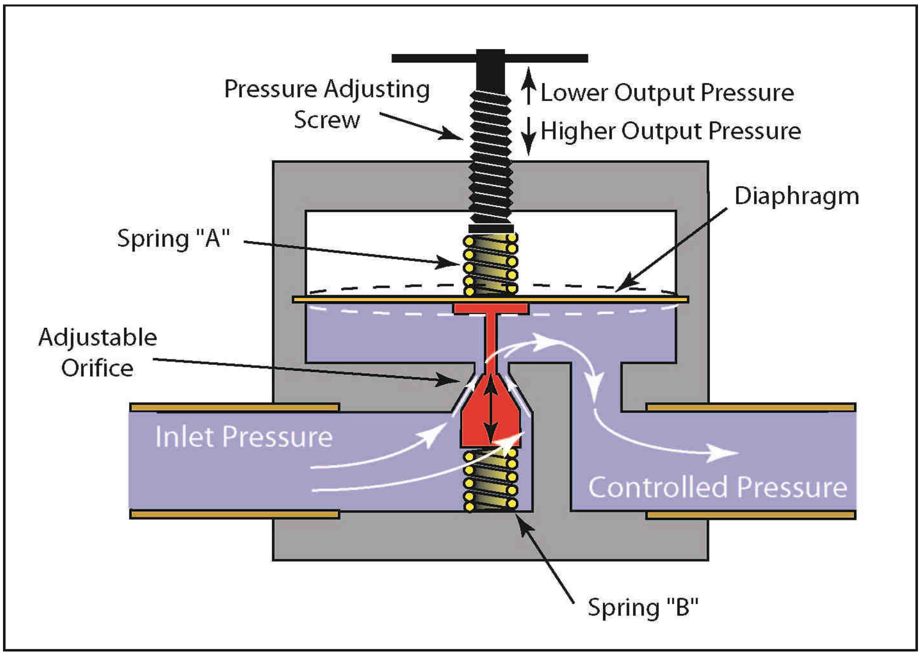


Figure 2.19: Simple flow control valve

The fluid flow is controlled by varying area of the valve opening through which fluid passes. The fluid flow can be decreased by reducing the area of the valve opening and it can be increased by increasing the area of the valve opening.

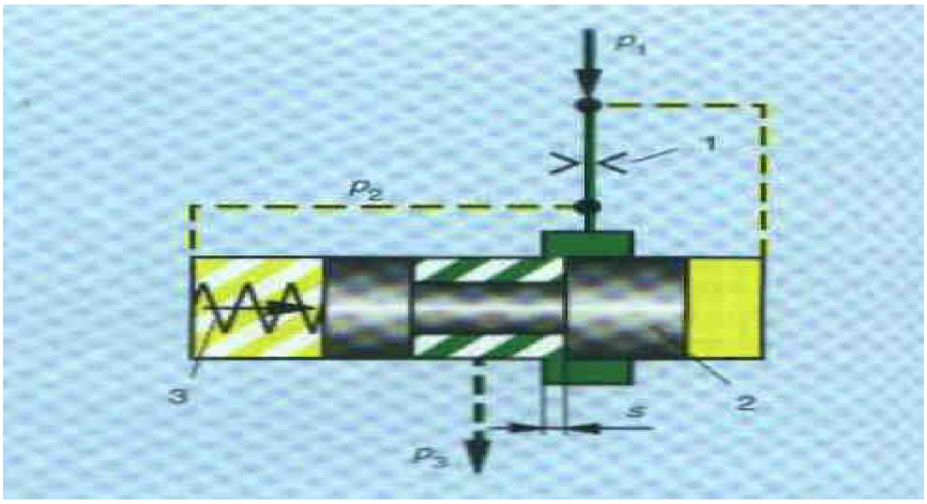


Figure 2.20: Pressure compensated FCV

The type of FCV, used to control the gate movement, keeps a set flow constant regardless of the pressure drop variation due to the load. This is achieved in that in addition to adjustable orifice (1) (measuring orifice), an additional moving orifice (2) is built into the system, which operates as a control orifice (pressure compensator) and at the same time as a comparator element in the closed loop control circuit (see figure 2.20).

In the static condition, the compensating spool (2) is biased fully open by the compensator spring (3). As soon as flows occurs, there will be a pressure drop across the valve and pressure upstream of the measuring orifice (1) tends to close the valve but this is opposed by the spring assisted by pressure from downstream of the measuring orifice. The compensator spool adopts a balanced position with a consequential pressure drop over the compensating orifice formed by the partially closed spool. A rise in supply pressure tends to close the spool and the increased pressure drop across the compensating orifice balances the increase in supply pressure. If the load pressure rises, the compensating orifice opens, again maintaining the pressure drop over the metering orifice at a set value [7].

2.2.2.9. Fixed orifice

Fixed orifice is a type of flow control valve (see figure 2.21). The purpose of this orifice is to prevent a sudden drop of the power intake gate should the hydraulic pipe between the cylinder and the main tank fail. So this is a fail-safe function [8].

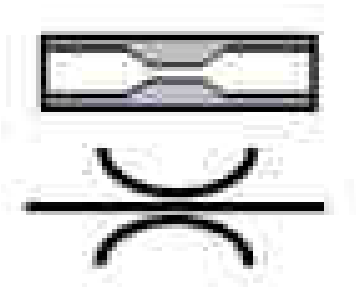


Figure 2.21: Fixed orifice

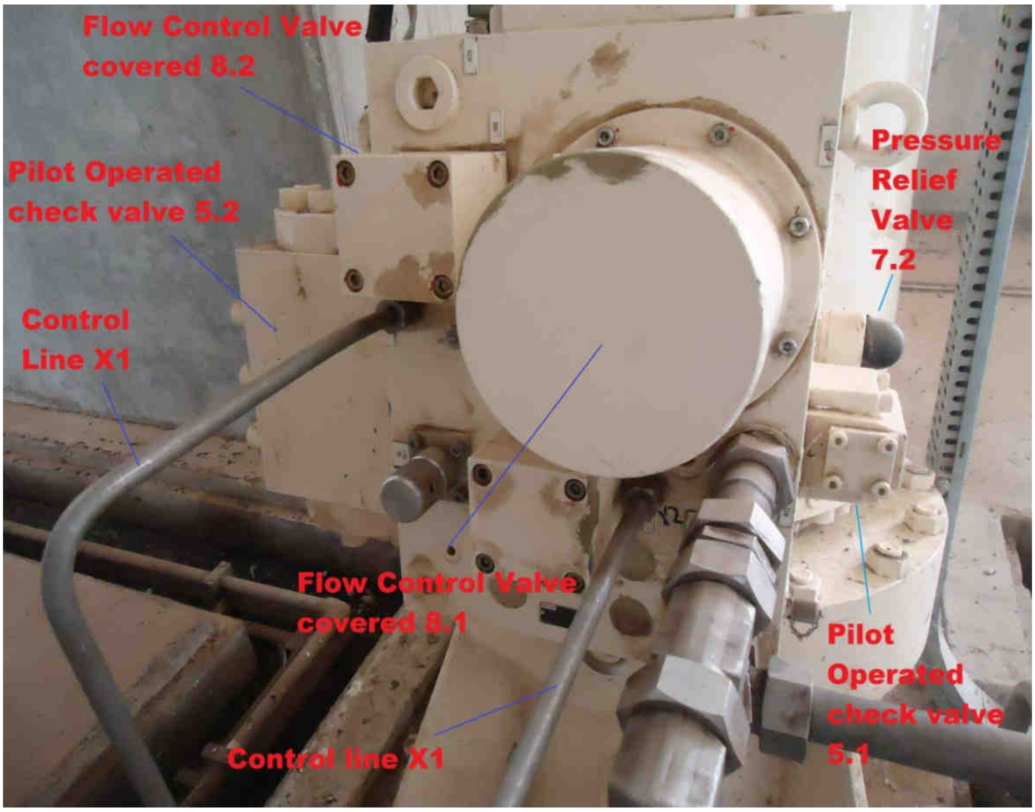


Figure 2.22: Control Manifold mounted on cylinder bottom

2.2.2.10. Overhead Tank

It is used to provide the additional oil to fill the piston side of the cylinder when lowering the gate. This additional quantity of oil is drawn so as to compensate for the missing quantity of oil due to the volume of cylinder rod when gate at closed position [8].It is mounted near the top of the cylinder (see figure 2.23)

Figure 2.23: Double Acting Cylinder and overhead oil tank

2.2.2.11. Pressure Sensor

The hydraulic system is equipped with a pressure sensor used to measure the working pressure of the hydraulic system and fed it back to the PLC control systems. The pressure sensor is located on the hydraulic power pack.

2.2.2.12. Position Sensor

The hydraulic system is equipped with a position sensor to measure the position of the intake gate.

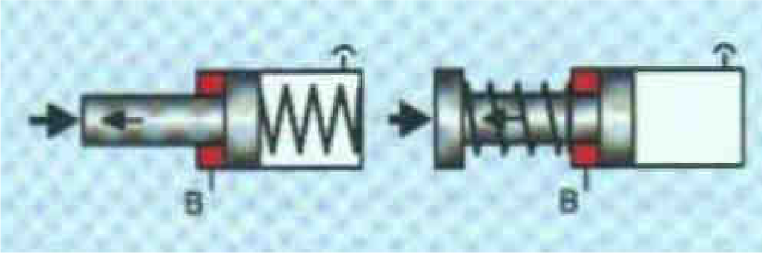
2.2.2.13. Double acting Cylinder

Nowadays the hydraulic cylinder is indispensable unit in any hydraulic system for converting the hydraulic energy into mechanical energy. The cylinder is the link between the hydraulic system and the machine or the load (see figure 2:24).

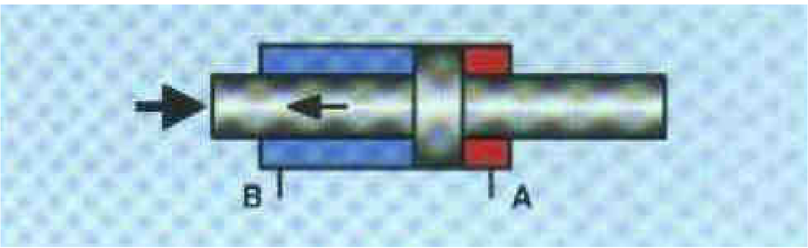
Due to their function, hydraulic cylinders can be categorized into two groups:

- Single Acting Cylinders.
- Double Acting Cylinders.

SAC can only exert force in one direction and the piston is then returned by a spring. DAC can exert force in both directions and it is the type used in this application (see figures: 2.23 and 2.24).



(a) Single Acting Cylinder



(b) Double Acting Cylinder

Figure 2.24: Hydraulic Cylinders

The components of double acting cylinder are shown in figure 2.25.

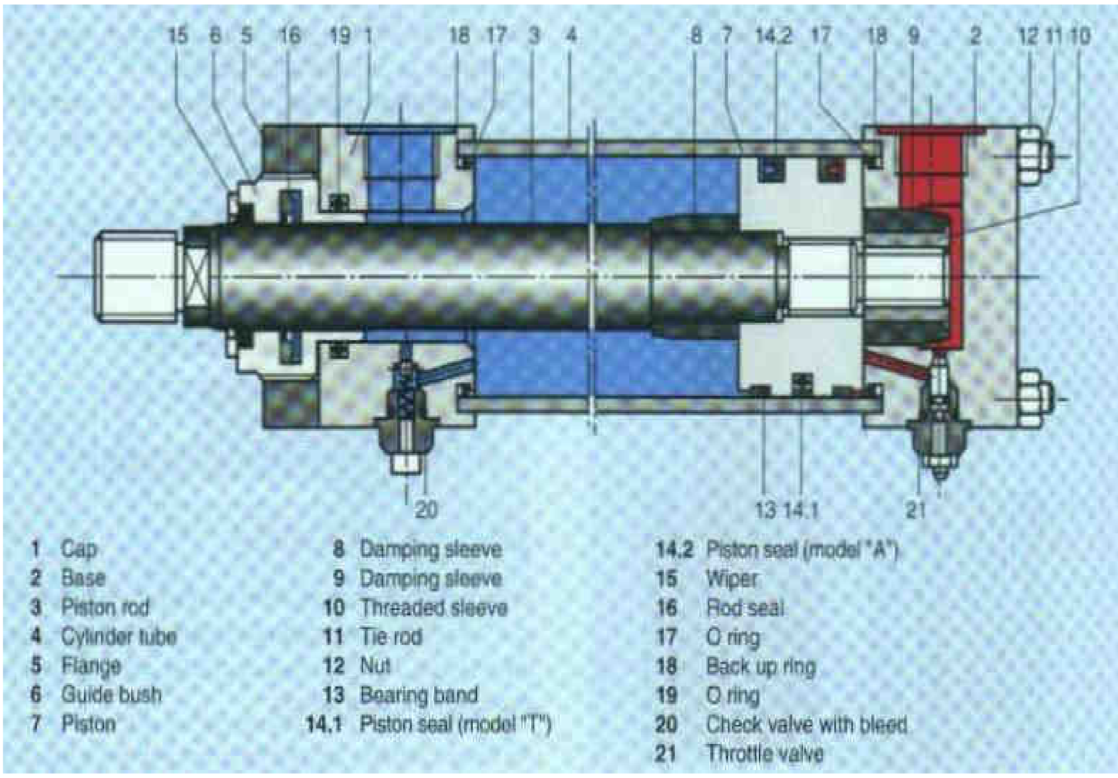


Figure 2.25: Components of double acting cylinder

The double acting cylinder is also equipped with a damping device called the cushion. The cushion decelerates the cylinder rod near the end of the stroke by restricting the flow rate leaving the cylinder chamber. The figure shows a typical design of a cylinder cushion (see figure 2.26).

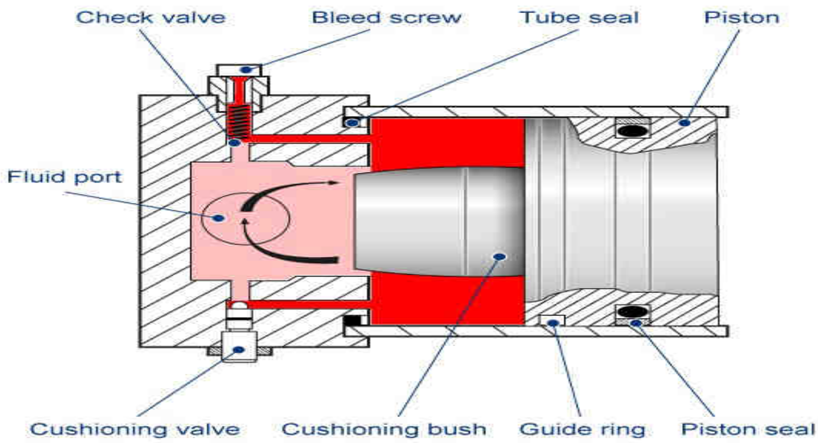


Figure 2.26: Cylinder Cushion

As the piston moves toward the cap (to the left in the figure), the cushioning bush enters the chamber in the cap and creates an additional resistance to the fluid leaving the chamber. The bush profile determines the desired deceleration. Near the end of the stroke, the fluid flows through the gap between the bush and the cap and through the cushioning valve with constant cross-sectional area. The check valve located between the chambers allows free flow to the piston chamber to ease the piston breakaway from the end position [10].

2.2.3. Description of Hydraulic Control System Functioning

There are three main functions of the hydraulic control system:

- Raising of power intake gate.
- Normal lowering of power intake gate.
- Emergency Lowering of power intake gate.

Raising time of the power intake gate depends on the amount of oil pumped below the piston in the hydraulic cylinder.

Lowering time of the power intake gate depends on the amount of oil discharged from below the piston and it is controlled by opening of the normal lowering flow control valve which is a pressure compensator type. So regardless of the gate weight, for a fixed opening of the FCV, the amount of oil discharged will be the same and the lowering time is also same.

Emergency Lowering time of the power intake gate also depends on the amount of oil discharged from below the piston and is controlled also by a flow control valve which is a variable orifice but without pressure compensator and hence the amount of oil flow depends on the pressure drop across the orifice which depends on the gate weight. So for a fixed opening of the FCV the amount of oil discharged is not the same and so the emergency lowering time.

The following functional description of the hydraulic control system of power intake gates should be read together with the hydraulic circuit diagram in figure 2.10.

2.2.3.1. Raising of the power intake gate

Raising time is 24 minutes for gates 1-4 and 12 minutes for gates 5-7.

From the closed position of the gate (bottom end position)

For gates 1-3 one electric motor will be operated.

For gates 4-7 both electric motors will be operated.

On pressing the gate raise pushbutton on the electrical control panel or pressing 1 then OK from the remote central control room, electric motor (items 2.1 or 2.2) will operate and drive the hydraulic pump (items 3.1 or 3.2). The pump will suck the oil from the main oil tank (item 1) and return it back to the tank for the starting period of the motor. After a time delay of approx. 5 s, the solenoid of the 2/2 way directional valve DCV1 (item 6.3) will be energized and reverse its position. Then the oil pressure will build and will be supplied under pressure to the rod side chamber of the cylinder (below the piston (item 13)) via check valves (items 4.1 or 4.2, 4.3 and 4.4). The oil displaced from the full bore side of the cylinder (above the piston), will be forced into the overhead tank (item 10) integrated onto the cylinder and the excess oil is forced back via check valve (item 4.5) into the main tank.

The position sensor (item 12) will indicate the gate position during the raising and when the gate reaches the fully open position, the electrical control system will stop the motor [8].

2.2.3.2. Normal Lowering of the intake gate

Lowering time is 11 minutes for all the units except unit 4 within 1 minute.

From the opened position of the gate (top end position)

On pressing the lower pushbutton on the control panel or pressing 0 then OK from the remote central control room, the solenoid of the 3/2 way directional valve DCV1 (item 6.1) for normal lowering, will be energized and reverse its position. The hydraulic oil in the rod chamber, which is now under the pressure of the gate weight, will open the pilot operated check valve (item 5.1).

Now the oil flows from the rod chamber to the piston chamber via the pilot operated check valve. The flow control valve (item 8.1) will regulate the closing speed by restricting the oil flow from the rod chamber to the piston chamber and hence control the normal lowering time. The addition oil to fill the piston chamber will be drawn from the overhead tank [8].

2.2.3.3. Emergency lowering of the intake gate

The emergency lowering is used for the protection of the turbine in case of over speed or when oil pressure in the governor system drops below the minimum.

From the opened position of the intake gate (top end position), on pressing the Emergency Lower pushbutton on the control panel or from the central control room, the solenoid of the 3/2 way directional valve DCV (item 6.2), for emergency lowering, will be energized and reverse its position. The hydraulic oil in the rod chamber, which is now under the pressure of the gate weight, will open the pilot operated check valve (item 5.2).

Now the oil flows from the rod chamber to the piston chamber via the pilot operated check valve. The flow control valve (item 8.2) will regulate the closing speed by restricting the oil flow from the rod chamber to the piston chamber [8].

CHAPTER THREE

SIMULATION MODEL AND GUI

A physical model of the hydraulic control system of power intake gates has been created with Matlab/Simscape/SimHydraulics toolbox. The simulation results from the model will be compared against measurements from the real system to see if they will give the same results.

SimHydraulics is a more powerful tool in modeling hydraulic systems .Instead of deriving, programming and solving dynamics equations of hydraulic system components, SimHydraulics provides more detailed component blocks for modeling hydraulic components in the Simulink environment. In SimHydraulics library, there are over 50 different blocks which include linear and rotary actuators, pumps, valves, pipelines. One of the most important advantages of this toolbox is that a SimHydraulics model can be connected to a mechanical system for a multidomain simulation. Moreover, a SimHydraulics system model closely resembles the hydraulic schematic, which lets the user to understand and analyze the model much more efficiently.

The SimHydraulics model is shown in figure 3.1. The model is constructed with the same components as the actual system, with some exceptions like the flow measurement which is not included in the real system. All the parameters of the simHydraulics blocks were extracted from components datasheets, some of them can be found directly and others must be calculated.

It is assumed that the friction in the piping, hoisting cylinder, between the gate rollers and the guide in the dam bodyis neglected to simplify the model.

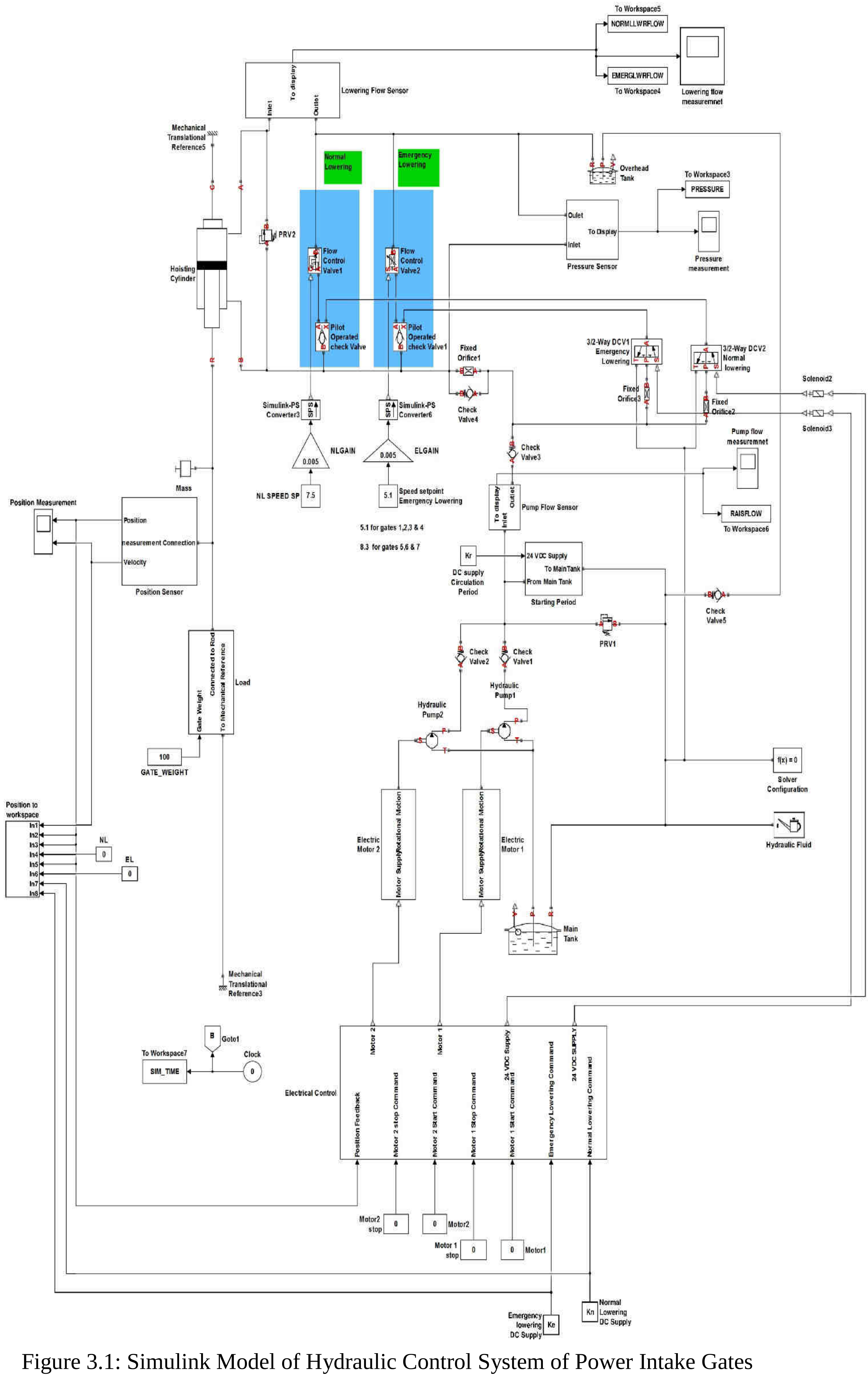


Figure 3.1: Simulink Model of Hydraulic Control System of Power Intake Gates

3.1. Electrical Control Model

The electrical Control system is modeled using the electric control subsystem as illustrated in figure 3.2.

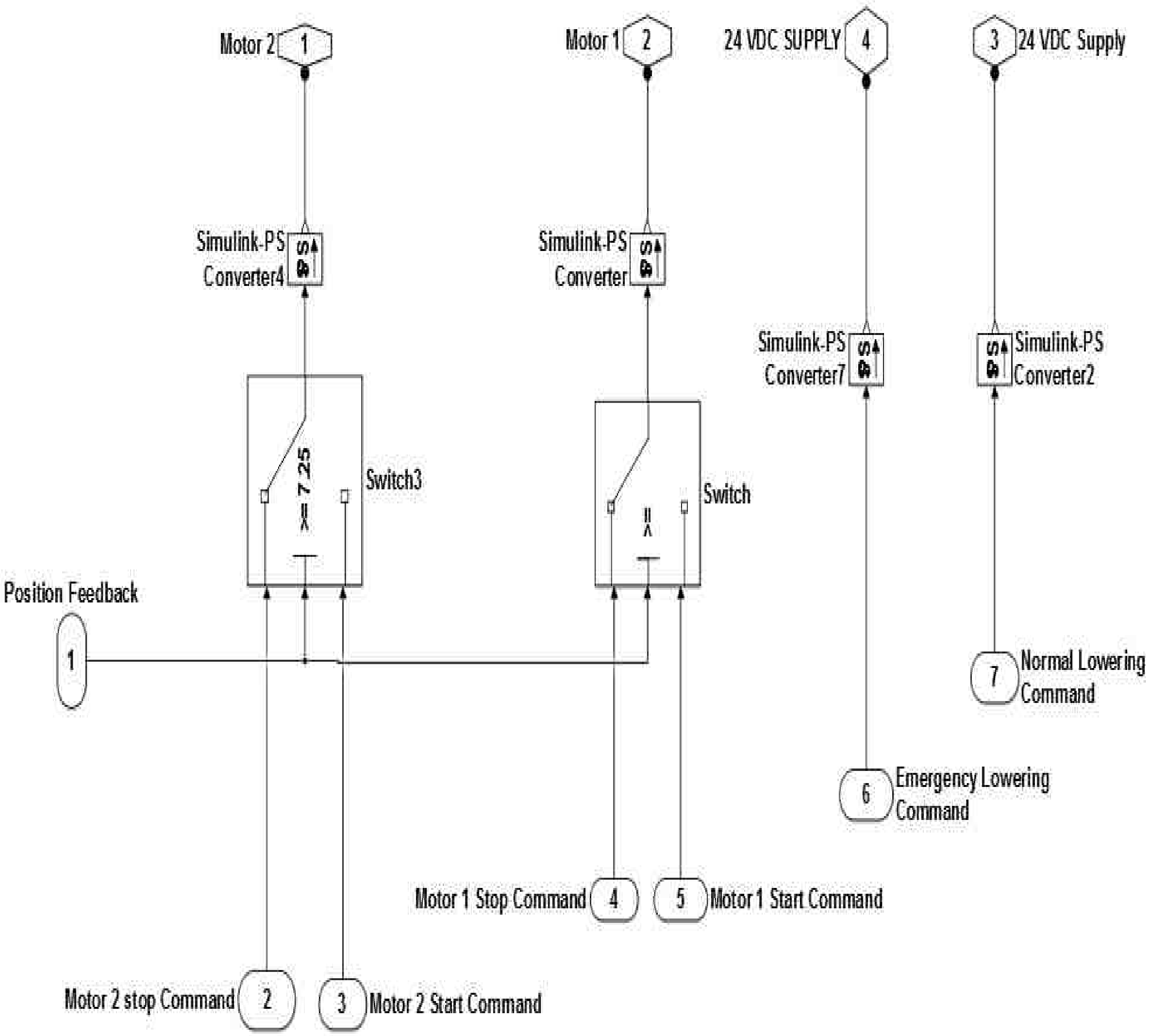


Figure 3.2: Electrical Control Subsystem

The subsystem consists of two switch blocks, one for each motor, that accept commands of start, stop and position feedback to stop the motor when the gate is fully opened at 7.25m and outputs rotational speed value for the electric motor subsystems.

It also accepts commands of emergency and normal lowering. The electric control subsystem supplies the solenoid valves for normal and emergency lowering, with voltage of 24 DC, through output ports 3 and 4 respectively.

The Simulink-PS Converter block is a very important block. It will be commonly used in the rest of the model and whenever there is a need for a simulink to physical signal conversion.

3.2. Oil tank model

The main oil tank and the overhead tank were modeled using Reservoir block (see figure 3.3). This block represents a pressurized hydraulic reservoir, in which fluid is stored under a specified pressure. The pressurization remains constant regardless of volume change. Connections P and R are hydraulic conserving ports associated with the pump and return lines, respectively. Connection V is a physical signal port.

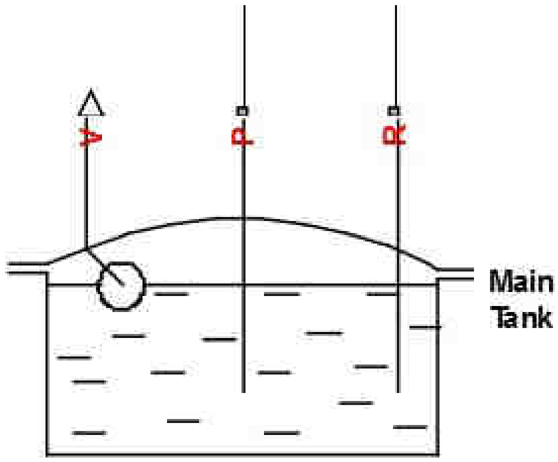


Figure 3.3: Reservoir block

The block computes volume of fluid in a tank and exports it outside through the physical signal port V. The fluid volume value does not affect the results of simulation. It is introduced merely for information purposes. It is possible for the fluid volume to become negative during simulation, which signals that the fluid volume is not enough for the proper operation of the system. By viewing the results of the simulation, you can determine the extent of the fluid shortage [11].

3.3. Electrical motors model

The two electrical motors are modeled by the subsystems Electrical Motor 1 and 2. Each subsystem is constructed with the Ideal Angular Velocity Source and the mechanical reference blocks. The Angular velocity source is connected to the mechanical reference and to the Electric Control Subsystem via port 2 (see figure 3.4).

This block receives a physical input signal, which is required rotational speed of the motor, from port 2 and output it to port 1 which is input to the hydraulic pump.

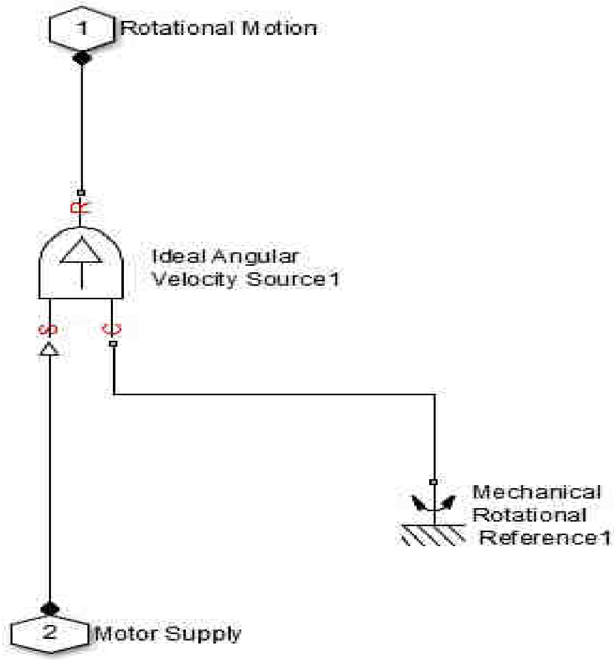


Figure 3.4: Electrical Motor subsystem

3.4. Hydraulic pumps model

The two hydraulic pumps were modeled using the Fixed-Displacement Pump block. The block represents a positive, fixed-displacement pump of any type as a data-sheet-based model. Port S is connected to the electrical motor, Port T to the main oil tank, and Port P is connected to the pressure line (see figure 3.5).

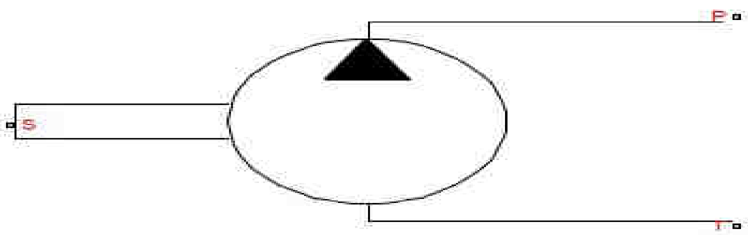


Figure 3.5: Hydraulic Pump block

The key parameters required for this block are pump displacement, volumetric and total efficiencies, nominal pressure, and angular velocity (see figure 3.6). All these parameters are generally provided in the data sheets or catalogs.

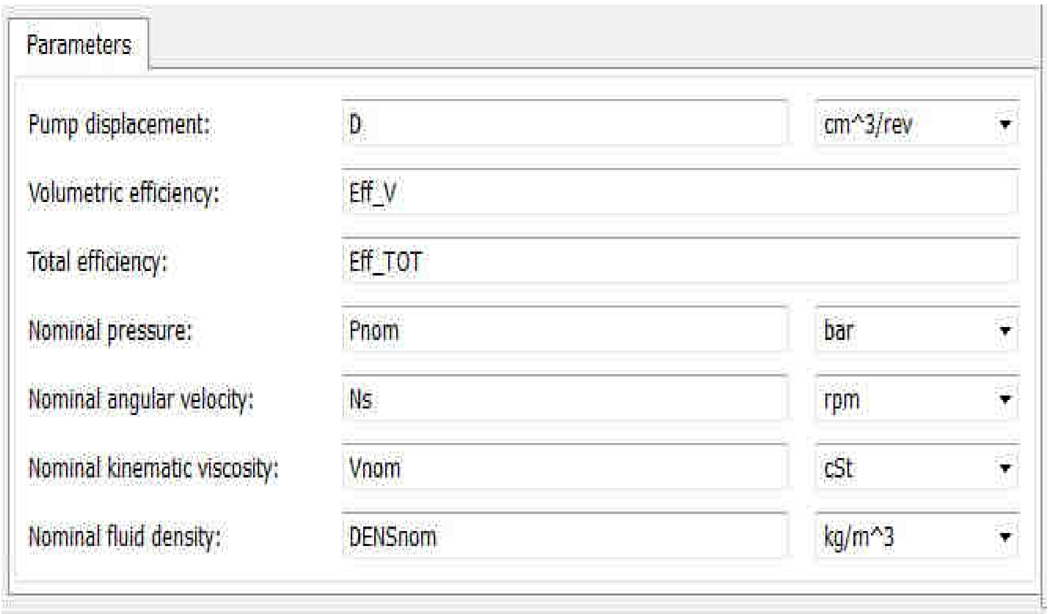


Figure 3.6: parameters used in pump block

Then these parameters are used in the calculations of the following equations:

$$q = D_{pump} \cdot \omega - k_{leak} \cdot P \qquad (3.1)$$

$$T = D \cdot P/\eta_{mech} \qquad (3.2)$$

$$k_{leak} = \frac{k_{HP}}{\nu \cdot \rho} \qquad (3.3)$$

$$k_{HP} = \frac{q_{leak} \cdot \nu_{nom} \cdot \rho_{nom}}{P_{nom}} \qquad (3.4)$$

$$P = P_P - P_T \qquad (3.5)$$

The leakage flow is determined based on the assumption that it is linearly proportional to the pressure differential across the pump and can be computed by using the Hagen-Poiseuille formula

$$P = \frac{128\mu l}{\pi d^4} q_{leak} = \frac{\mu}{k_{HP}} q_{leak} \quad (3.6)$$

The leakage flow at $p = p_{\mathbf{nom}}$ and $\nu = \nu_{\mathbf{nom}}$ can be determined from the catalog data

$$q_{leak} = D \cdot \omega_{nom}(1 - \eta_v) \qquad (3.7)$$

This provides the formula to determine the Hagen-Poiseuille coefficient in equation (3.4).

3.5. Check valve model

The five check valves are modeled by check valve block. The Check Valve block represents a hydraulic check valve as a data-sheet-based model (see figure 3.7).

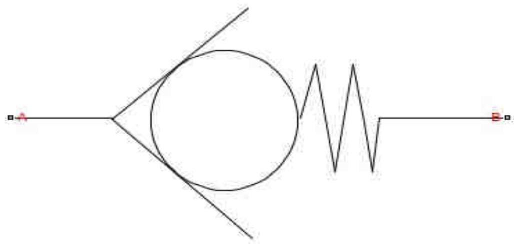


Figure 3.7: check valve block

The purpose of the check valve is to permit flow in one direction and block it in the opposite direction. The following figure 3.8 shows the typical dependency between the valve passage area A and the pressure differential across the valve $P=P_A-P_B$.

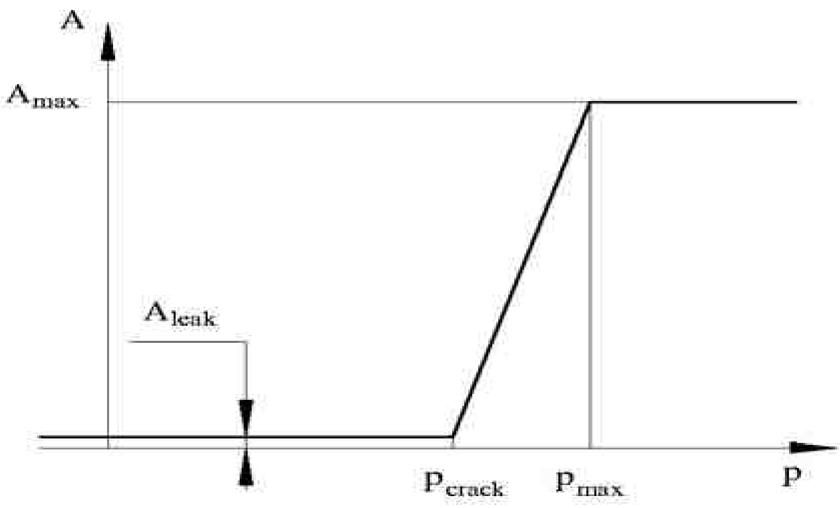


Figure 3.8: relation between passage area and pressure differential across the check valve

The valve remains closed while pressure differential across the valve is lower than the valve cracking pressure. When cracking pressure is reached, the value control member (spool, ball, poppet, etc.) is forced off its seat, thus creating a passage between the inlet and outlet. If the flow rate is high enough and pressure continues to rise, the area is further increased until the control member reaches its maximum. At this moment, the valve passage area is at its maximum. The valve maximum area and the cracking and maximum pressures are generally provided in the catalogs and are the three key parameters of the block [11].

In addition to the maximum area, the leakage area is also required to characterize the valve. The main purpose of the parameter is not to account for possible leakage, even though this is also important, but to maintain numerical integrity of the circuit by preventing a portion of the system from getting isolated after the valve is completely closed. An isolated or "hanging" part of the system could affect computational efficiency and even cause failure of computation. Theoretically, the parameter can be set to zero, but it is not recommended [11].

The flow rate is determined according to the following equations:

$$q = C_D \cdot A(p)\sqrt{(2/\rho)} \cdot \frac{P}{(P^2+P_{cr}^2)^{1/4}} \quad (3.8)$$

$$A(p) = \begin{cases} A_{leak} for P \leq P_{cr} \\ A_{leak} + k \cdot (P - P_{cr}) for P_{cr} \leq P < P_{max} \\ A_{max} for P \geq P_{max} \end{cases} \quad (3.9)$$

$$k = \frac{A_{max}}{(P^2+P_{cr}^2)^{1/4}} \quad (3.10)$$

$$P = P_A - P_B (3.11)$$

$$P_{cr} = \frac{\rho}{2\left(\frac{Re_{cr} \cdot \nu}{C_D \cdot D_H}\right)^2} \quad (3.12)$$

3.6. DCV model

The DCVs were modeled using a 3-way Directional Valve block. The fluid flow is pumped in the valve through the inlet line and is distributed between an outside pressure line and the return line. The block has three hydraulic connections, corresponding to inlet port (P), actuator port (A), and return port (T), and one physical signal port connections (S), which controls the spool position (see figure 3.9). The block is built of two Variable Orifice blocks, connected as shown in the following diagram figure 3.9.

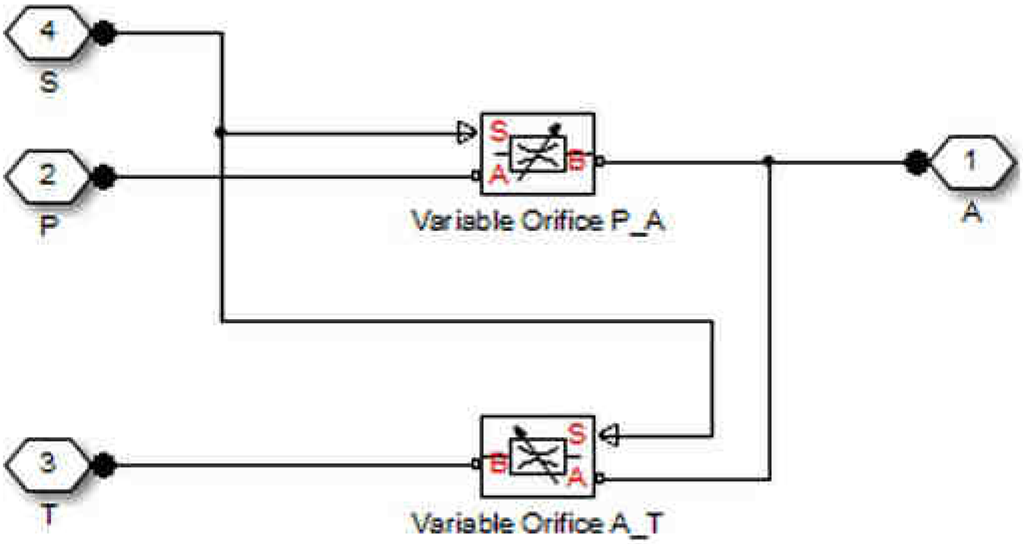


Figure 3.9: 3/2 Way Directional Valve block

One Variable Orifice block, called orifice P_A, is installed in the P – A path. The second Variable Orifice block, called orifice A – T, is installed in the A – T path. Both blocks are controlled by the same position signal, provided through the physical signal port S, but the orifice orientation parameter in the block instances is set in such a way that positive signal at port S opens orifice P – A and closes orifice A – T. As a result, the openings of the orifices are computed as follows:

$$h_{PA} = h_{PA0} + x (3.13)$$

$$h_{AT} = h_{AT0} - x (3.14)$$

Figure 3.10 shows the parameters used in the direction control valve block.

The valve simulated by the 3-Way Directional Valve block is assumed to be symmetrical. This means that both orifices are of the same shape and size and are parameterized with the same method.

The passage area is assumed to be linearly dependent on the control member displacement, that is, the orifice is assumed to be closed at the initial position of the control member (zero displacement), and the maximum opening takes place at the maximum displacement [11].

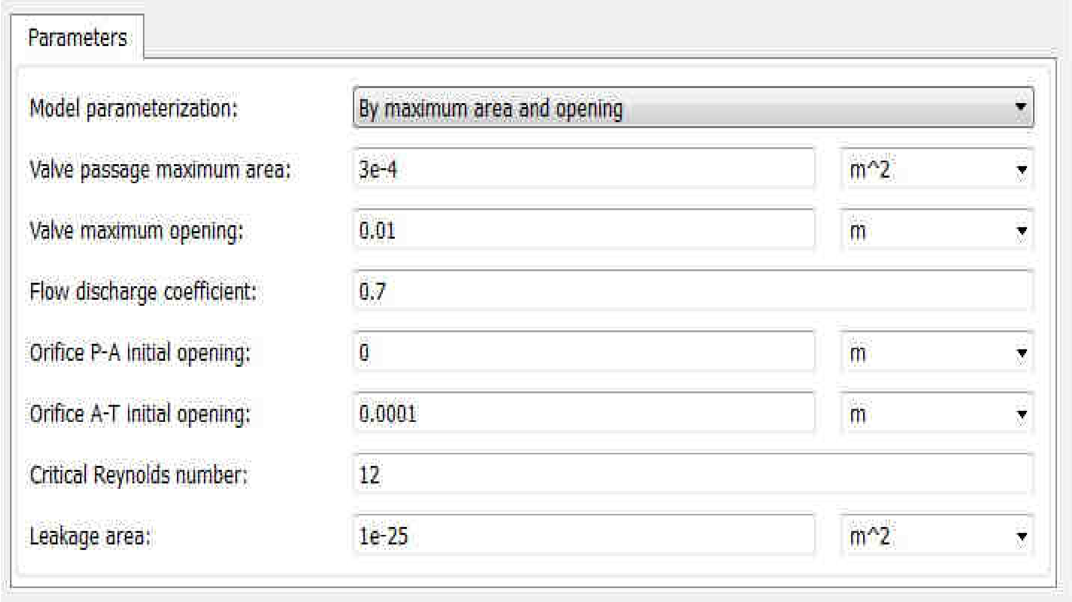


Figure 3.10: Parameters used in Directional Valve block

3.7. Starting Period of the Motor Model

The starting period of the motor is modeled using the starting period subsystem as illustrated in the figure 3.11.

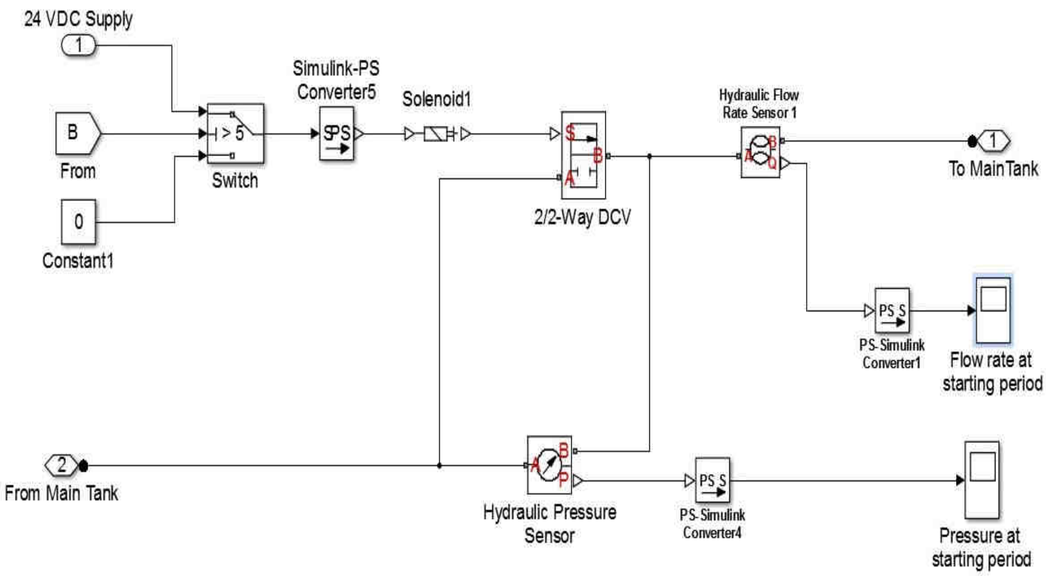


Figure 3.11: Starting Period Subsystem

The subsystem consists of 2/2 directional control valve block that it solenoid accepts a delayed 24 VDC signal through input port 1. The delay time is 5 s which is a condition in the switch block. During this time the pump circulates the oil from the tank to the tank (from port A to port B). After the expiration of the starting period the directional valve will reverse its position and port A blocked allowing the pressure to build up and flow into the hydraulic circuit below the cylinder piston. There are also a flow and pressure sensor to visualize the flow and pressure of oil at the starting period.

3.8. FCV Model

The two FCVs for normal lowering was modeled using the FCV block. The block represents a pressure-compensated FCV as a data-sheet-based model. The valve is based on a Pressure Compensator block installed upstream from a Variable Orifice block, as shown in figure 3.12.

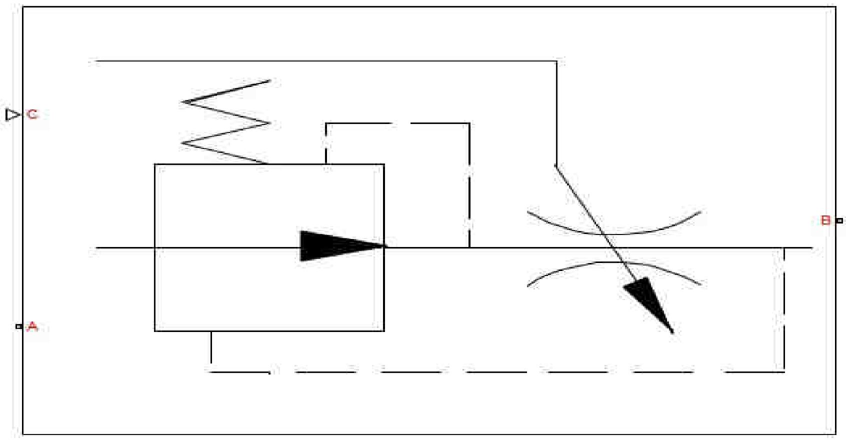


Figure 3.12: Flow control valve Block

The passage area of the variable orifice is assumed to be linearly dependent on the control member displacement, that is, the orifice is assumed to be closed at the initial position of the control member (zero displacement), and the maximum opening takes place at the maximum displacement.

After the area has been determined, the flow rate is computed according to the following equations:

$$q = C_D \cdot A(h)\sqrt{(2/\rho)} \cdot \frac{P}{(P^2+P_{cr}^2)^{1/4}} \quad (3.15)$$

$$P = P_A - P_B \quad (3.16)$$

$$P_{cr} = \frac{\rho}{2\left(\frac{Re_{cr} \cdot \nu}{C_D \cdot D_H}\right)^2} \quad (3.17)$$

$$h = x_0 + x \cdot or \quad (3.18)$$

$$A(h) = \begin{cases} \frac{h \cdot A_{max}}{h_{max}} + A_{leak} for h > 0 \\ A_{leak} for h \leq 0 \end{cases} \quad (3.19)$$

$$D_H = \sqrt{\frac{4A(h)}{\pi}} \quad (3.20)$$

The Pressure Compensator block represents a hydraulic pressure compensating valve which are used to maintain preset pressure differential across a hydraulic component to minimize the influence of pressure variation on a flow rate passing through the component. The block is implemented as a data-sheet-based model, based on parameters usually provided in the manufacturer's catalogs or data sheets.

Pressure compensator is a normally open valve. Its opening is proportional to pressure difference between ports X and Y and the spring force. Figure 3.13 shows typical relationship between the valve passage area A and the pressure difference p_{xy} [11].

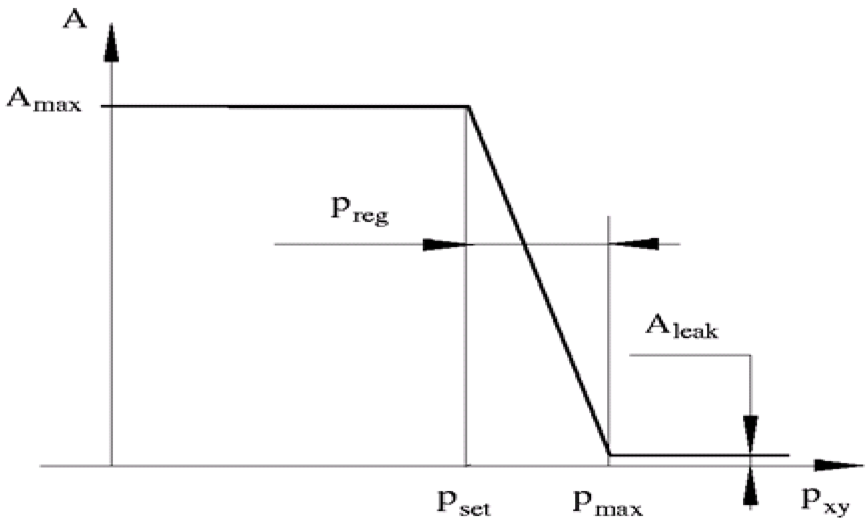


Figure 3.13: relation between passage area and pressure differential across the compensator

The main parameters of the block are the valve maximum area and regulation range. In addition, you need to specify the leakage area of the valve. Physically, it represents a possible clearance in the closed valve, but the main purpose of the parameter is to maintain numerical integrity of the circuit by preventing a portion of the system from getting isolated after the valve is completely closed. An isolated or "hanging" part of the system could affect computational efficiency and even cause failure of computation [11].

The flow rate is computed according to the following equations:

$$q = C_D \cdot A(h)\sqrt{(2/\rho)} \cdot \frac{P}{(P^2 + P_{cr}^2)^{1/4}} \quad (3.21)$$

$$P = P_A - P_B \quad (3.22)$$

$$P_{cr} = \frac{\rho}{2\left(\frac{Re_{cr} \cdot \nu}{C_D \cdot D_H}\right)^2} \quad (3.23)$$

$$D_H = \sqrt{\frac{4A(P)}{\pi}} \quad (3.24)$$

$$A(p) = \begin{cases} A_{max} \, for \, P_{xy} \leq P_{set} \\ A_{max} + k \cdot (P_{xy} - P_{set}) \, for \, P_{set} \leq P_{xy} < P_{max} \\ A_{max} \, for \, P_{xy} \geq P_{max} \end{cases} \quad (3.25)$$

$$k = \frac{A_{max} - A_{leak}}{P_{reg}} \quad (3.26)$$

Then the parameters of the variable orifice block and the pressure compensator block can be combined together to get the parameters of the flow control valve as shown in figure 3.14.

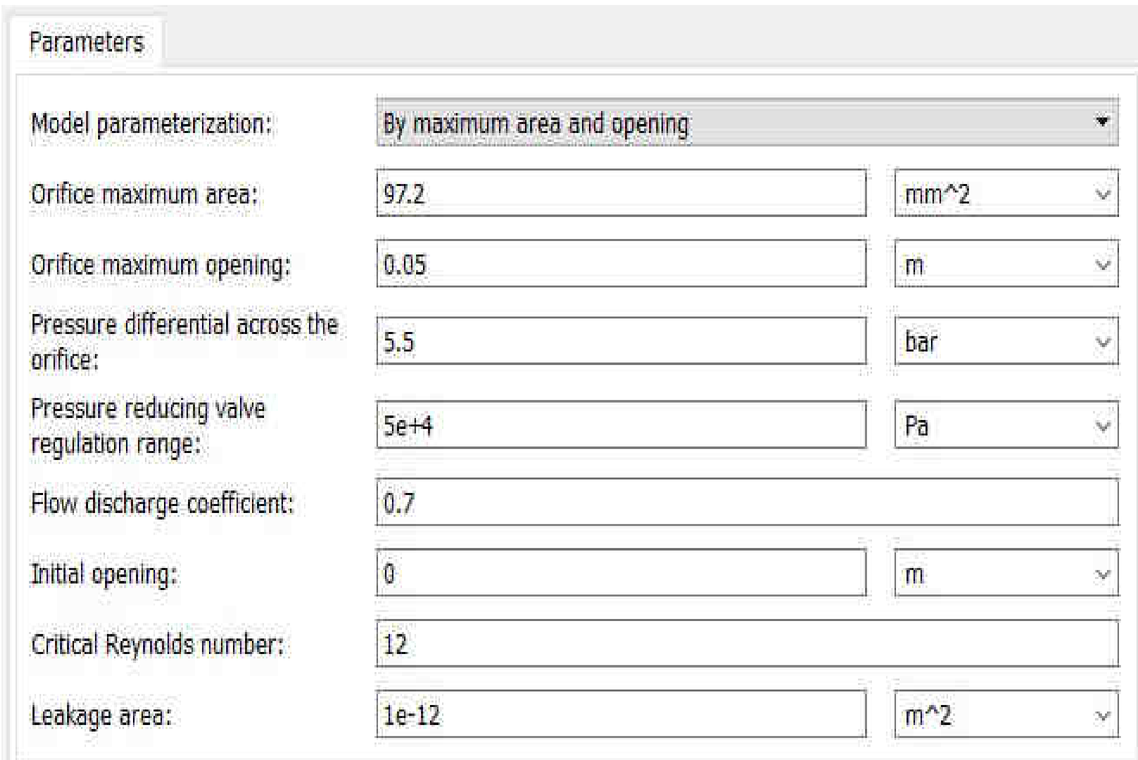


Figure 3.14: Parameters used in Flow Control Valve block for normal lowering

While the flow control valve for emergency lowering was modeled using a variable orifice block only without a pressure compensator. The equations used to compute the flow rate are the same as those used in the previous variable orifice for normal lowering. Trial and error method was followed to obtain the parameters of this block, since there is no available data sheet for it. Figure 3.15 shows the parameters used in the Flow Control Valve block for emergency lowering.

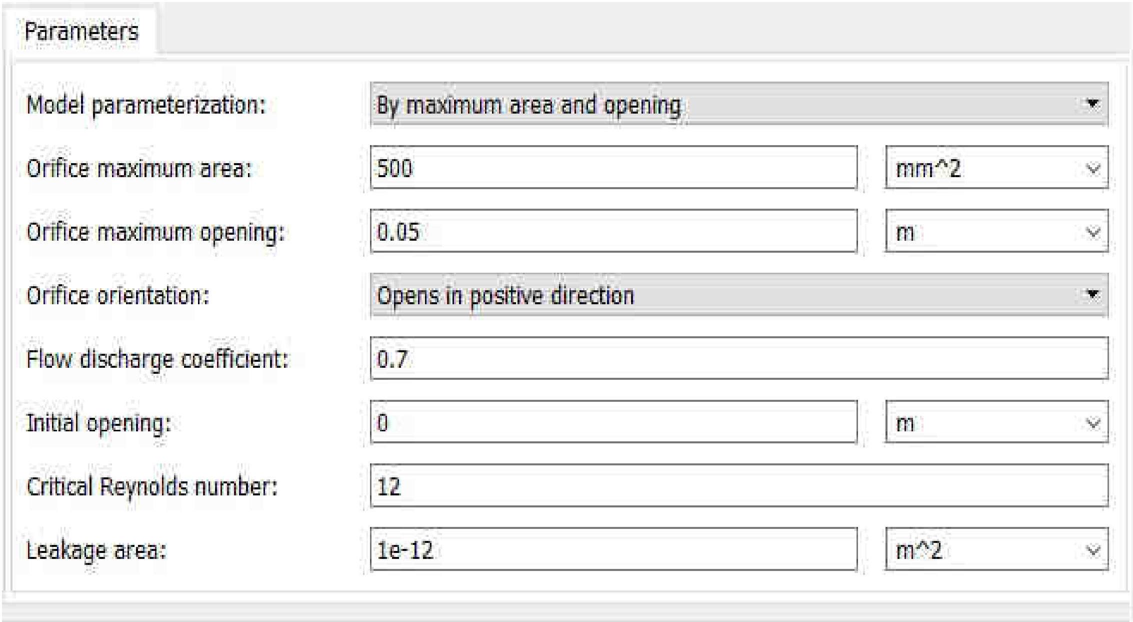


Figure 3.15: Parameters used in Flow Control Valve block for Emergency lowering

3.9. PRV Model

The two PRVs in the system were modeled using the pressure relief valve block which represents a hydraulic pressure relief valve as a data-sheet-based model (see figure 3.16).

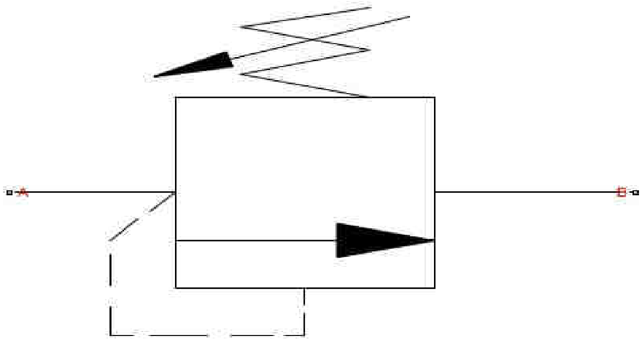


Figure 3.16: Pressure Relief Valve Block

Typical dependency between the valve passage area A and the pressure differential p across the valve is shown in figure 3.17.

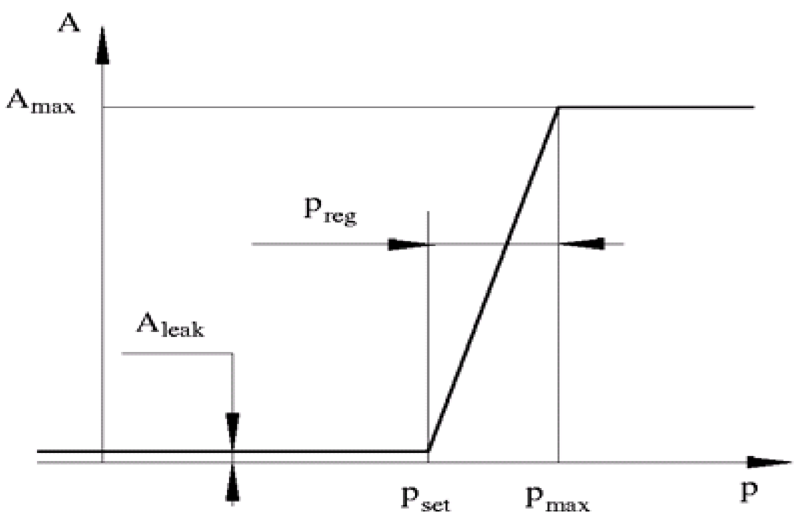


Figure 3.17: Relation between passage area and pressure differential across the pressure relief valve

The valve remains closed while pressure at the valve inlet is lower than the valve preset pressure. When the preset pressure is reached, the value control member (spool, ball, poppet, etc.) is forced off its seat, thus creating a passage between the inlet and outlet. Some fluid is diverted to a tank through this orifice, thus reducing the pressure at the inlet. If this flow rate is not enough and pressure continues to rise, the area is further increased until the control member reaches its maximum. At this moment, the maximum flow rate is passing through the valve. The value of a maximum flow rate and the pressure increase over the preset level to pass this flow rate are generally provided in the catalogs. The pressure increase over the preset level is frequently referred to as valve steady state error, or regulation range. The valve maximum area and regulation range are the key parameters of the block [11].

In addition to the maximum area, the leakage area is also required to characterize the valve. The main purpose of the parameter is not to account for possible leakage, even though this is also important, but to maintain numerical integrity of the circuit by preventing a portion of the system from getting isolated after the valve is completely closed. An isolated or "hanging" part of the system could affect computational efficiency and even cause failure of computation. Theoretically, the parameter can be set to zero, but it is not recommended. Figure 3.18 shows parameters used in pressure relief valve block [11].

The flow rate is determined according to the following equations:

$$q = C_D \cdot A(h)\sqrt{(2/\rho)} \cdot \frac{P}{(P^2+P_{cr}^2)^{1/4}} \quad (3.27)$$

$$P = P_A - P_B \quad (3.28)$$

$$P_{cr} = \frac{\rho}{2\left(\frac{Re_{cr}\cdot\nu}{C_D\cdot D_H}\right)^2} \quad (3.29)$$

$$A(p) = \begin{cases} A_{leak} for P \leq P_{set} \\ A_{leak} + k\cdot(P - P_{set}) for P_{set} \leq P < P_{max} \\ A_{max} for P \geq P_{max} \end{cases} \quad (3.30)$$

$$k = \frac{A_{max} - A_{leak}}{P_{reg}} \quad (3.31)$$

$$D_H = \sqrt{\frac{4A(P)}{\pi}} \quad (3.32)$$

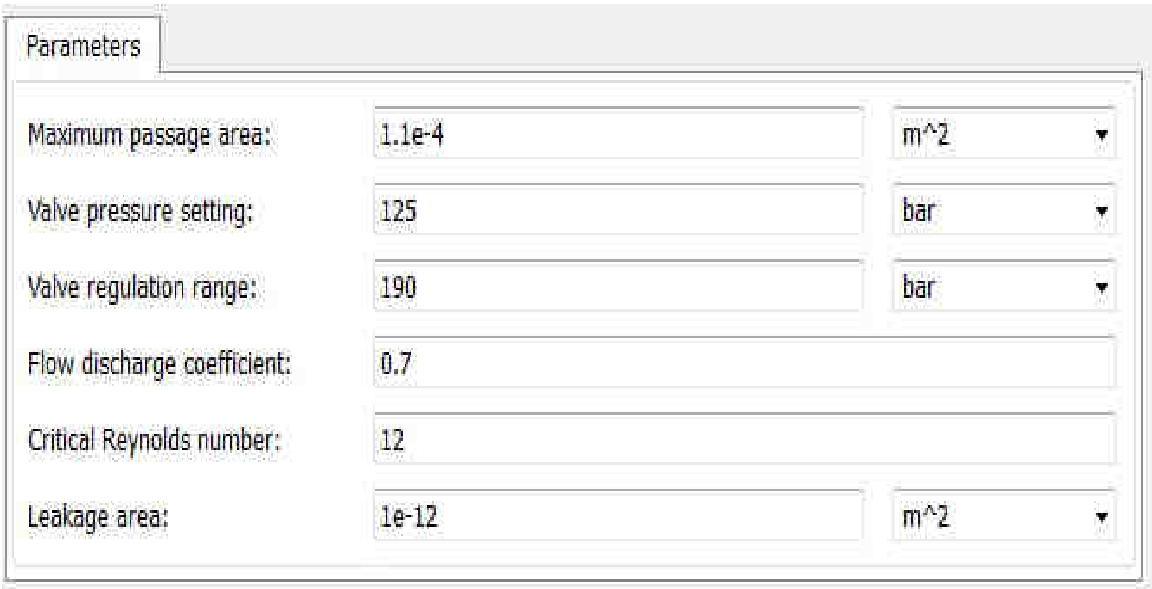


Figure 3.18: Parameters used in Pressure Relief Valve block

3.10. The Pilot-Operated Check Valve Model

The two Pilot-Operated Valves for normal lowering and emergency lowering were modeled using the Pilot-Operated Valve block which represents a hydraulic pilot-operated

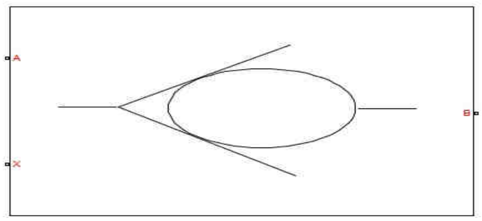
check valve as a data-sheet-based model (see figure 3.18a). The purpose of the check valve is to permit flow in one direction and block it in the opposite direction but unlike a conventional check valve; the pilot-operated check valve can be opened in the opposite direction by pilot pressure P_X (see figure 3.19b).
The force acting on the poppet is determined as

$$F = P_A \cdot A_A + P_X \cdot A_X - P_B \cdot A_B \quad (3.33)$$

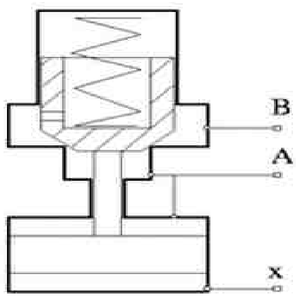
This equation is commonly used in a slightly modified form

$$P_e = P_A + P_X \cdot K_P - P_B \quad (3.34)$$

Where kp = AX / AA is usually referred to as pilot ratio and pe is the equivalent pressure differential across the poppet. The valve remains closed while this pressure differential across the valve is lower than the valve cracking pressure. When cracking pressure is reached, the value control member (spool, ball, poppet, etc.) is forced off its seat, thus creating a passage between the inlet and outlet. If the flow rate is high enough and pressure continues to rise, the area is further increased until the control member reaches its maximum. At this moment, the valve passage area is at its maximum. The valve maximum area and the cracking and maximum pressures are generally provided in the catalogs and are the three key parameters of the block [10].



(a) Block



(B) Cross section

Figure 3.19: Pilot-Operated Check valve

In addition to the maximum area, the leakage area is also required to characterize the valve. The main purpose of the parameter is not to account for possible leakage, even though this is also important, but to maintain numerical integrity of the circuit by preventing a portion of the system from getting isolated after the valve is completely closed. An isolated or "hanging" part of the system could affect computational efficiency and even cause failure of computation. Theoretically, the parameter can be set to zero, but it is not recommended. Figure 3.20 shows parameters used in pilot operated check valve block [11].

The flow rate is determined according to the following equations:

$$q = C_D \cdot A(p)\sqrt{(2/\rho)} \cdot \frac{P}{(P^2+P_{cr}^2)^{1/4}} \quad (3.35)$$

$$P_e = P_A + P_X \cdot K_P - P_B (3.36)$$

$$A(p) = \begin{cases} A_{leak} for P \leq P_{cr} \\ A_{leak} + k \cdot (P - P_{cr}) for P_{cr} \leq P < P_{max} \\ A_{max} for P \geq P_{max} \end{cases} \quad (3.37)$$

$$k = \frac{A_{max}}{(P^2+P_{cr}^2)^{1/4}} \quad (3.38)$$

$$P = P_A - P_B \quad (3.39)$$

$$P_{cr} = \frac{\rho}{2\left(\frac{Re_{cr} \cdot v}{C_D \cdot D_H}\right)^2} \quad (3.40)$$

$$D_H = \sqrt{\frac{4A(P)}{\pi}} (3.41)$$

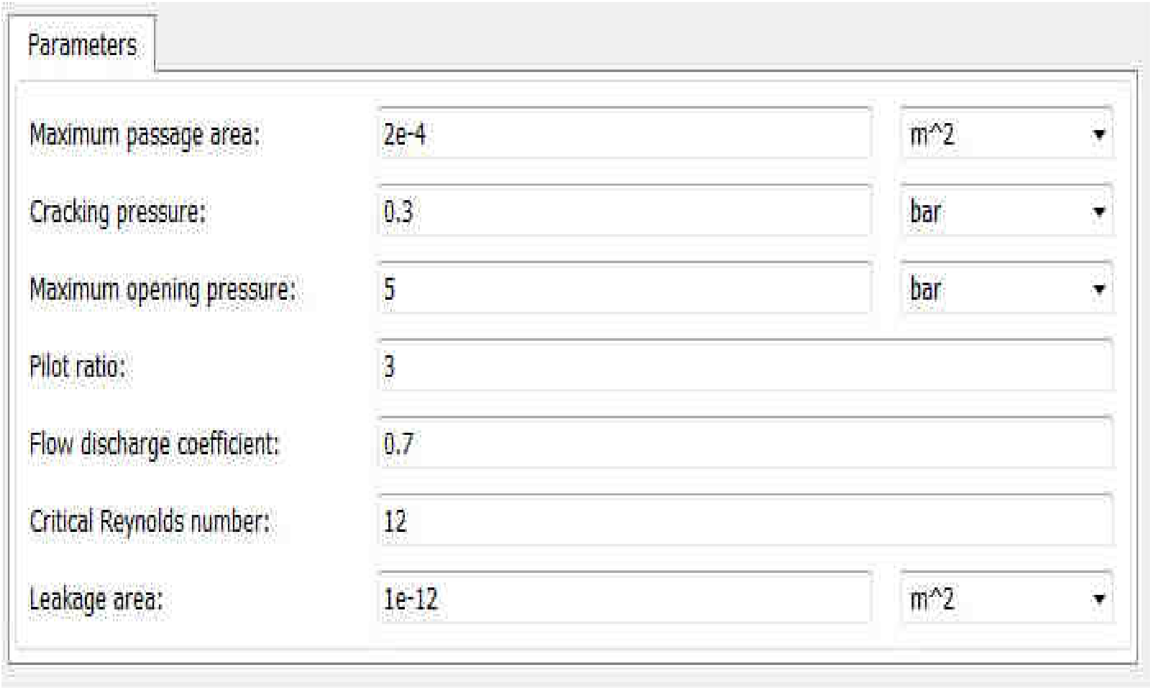


Figure 3.20: Parameters used in Pilot-Operated Check Valve block

3.11. The Hydraulic Cylinder Model

The hydraulic cylinder was modeled using the Double-Acting Hydraulic Cylinder (Simple) block which represents a simplified version of a double-acting hydraulic cylinder (see figure 3.21).

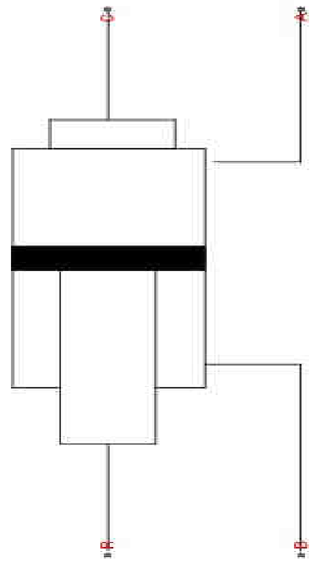


Figure 3.21: Double-Acting Hydraulic Cylinder (Simple) block

The block developed for applications where only the basic cylinder functionality must be reproduced, in exchange for better numerical efficiency. For these reasons, such factors as

fluid compressibility, friction, and leakages are assumed to be negligible. The hard stops are assumed to be fully inelastic, to eliminate any possible oscillations at the end of the stroke. The model is especially suitable for real-time and HIL simulation, if such simplifications are acceptable. Parameters used in the Hydraulic cylinder block are shown in figure 3.22 [11].
The model is described with the following equations:

$$F = P_A \cdot A_A - P_B \cdot A_B - F_C \qquad (3.42)$$

$$q_A = A_A \cdot V \quad (3.43)$$

$$q_B = A_B \cdot V \qquad (3.44)$$

$$\frac{dx}{dt} = V \qquad (3.45)$$

$$V = V_R - V_C \qquad (3.46)$$

$$F_C = \begin{cases} (x - x_E) \cdot K_P \cdot V \quad if\, x > x_E\,, V > 0 \\ (x - x_R) \cdot K_P \cdot V \quad if\, x < x_R\,, V < 0 \\ 0 \qquad otherwise \end{cases} \qquad (3.47)$$

$$x_E = S - x_0 \qquad (3.48)$$

$$x_R = -x_0 \qquad (3.49)$$

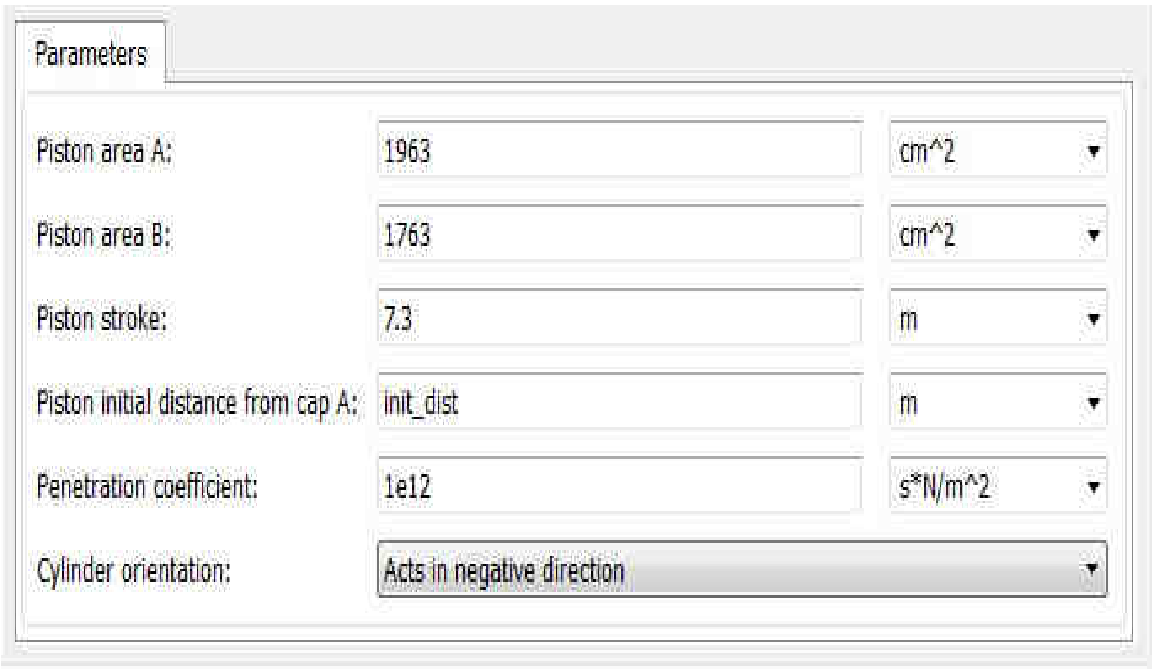


Figure 3.22: Parameters used in Double-Acting Hydraulic Cylinder (Simple) block

3.12. The Gate Weight Model

The Gate Weight is modeled using load subsystem which contains an ideal source of mechanical energy that generates force proportional to the input physical signal and also it contains a gain block that takes into account the gravity acceleration and conversion of gate mass from tons to Kgs. The source is ideal in a sense that it is assumed to be powerful enough to maintain specified force at its output regardless of the velocity at source terminals (see figure 3.23).

Connections R and C, ideal source of mechanical energy, are mechanical translational conserving ports. Port S is a physical signal port that accepts the gate mass from Input port 1. Output Port 1 is connected to the hydraulic cylinder rod. You can use the entire variety of Simulink® signal sources to generate the desired force variation profile. Positive signal at port S generates force acting from C to R. The force generated by the source is directly proportional to the signal at the control port S [11].

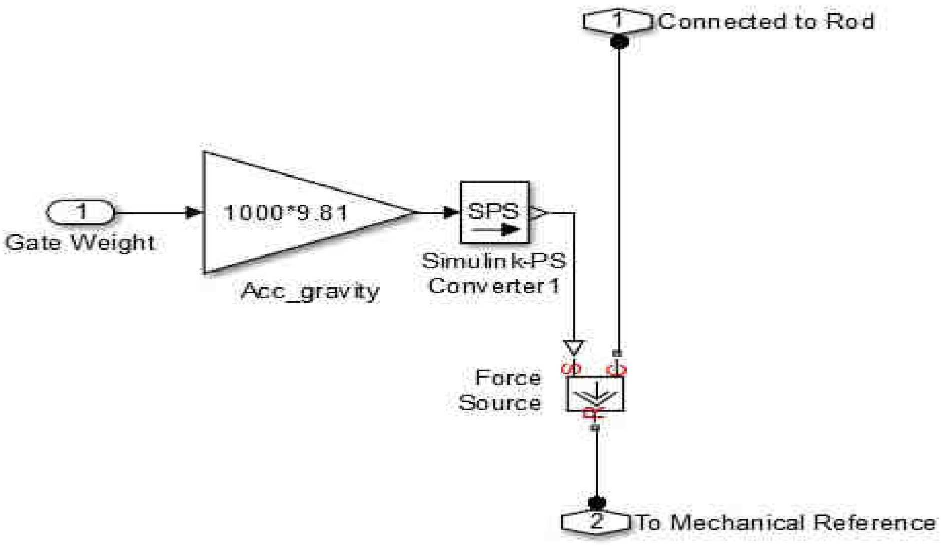


Figure 3.23: Load Subsystem

3.13. Sensors Model

There are three types of sensors used in the model.

- Position Sensor.
- Pressure Sensor.
- Flow Rate Sensor.

The position and pressure sensors are existing in the real system but the flow rate sensor is only found in the model.

3.13.1. The position sensor is modeled using the position sensor subsystem as illustrated in figure 3.24.

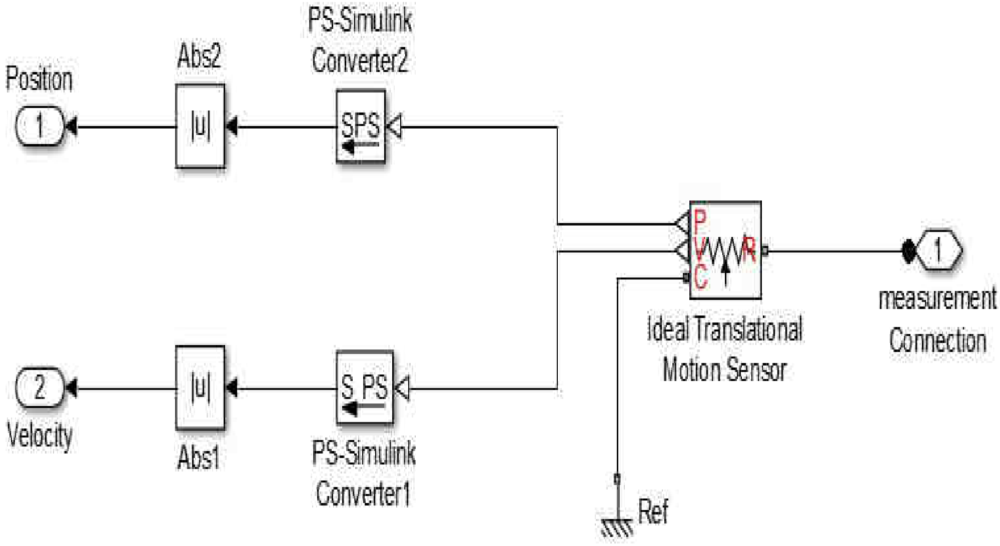


Figure 3.24: Position Sensor Subsystem

The Ideal Translational Motion Sensor block represents a device that converts an across variable measured between two mechanical translational nodes into a control signal proportional to velocity or position. You can specify the initial position (offset) as a block parameter.

The sensor is ideal since it does not account for inertia, friction, delays, energy consumption, and so on.

Connections R and C are mechanical translational conserving ports that connect the block to the nodes whose motion is being monitored. Connections V and P are physical signal ports that output the velocity and position values, respectively, for display [11].

3.13.2. The pressure sensor is modeled using the pressure sensor block subsystem as illustrated in figure 3.25.

The Hydraulic Pressure Sensor block represents an ideal hydraulic pressure sensor, that is, a device that converts hydraulic pressure differential measured between two points into a control signal proportional to this pressure. The sensor is ideal because it does not account for inertia, friction, delays, pressure loss, and so on [11].

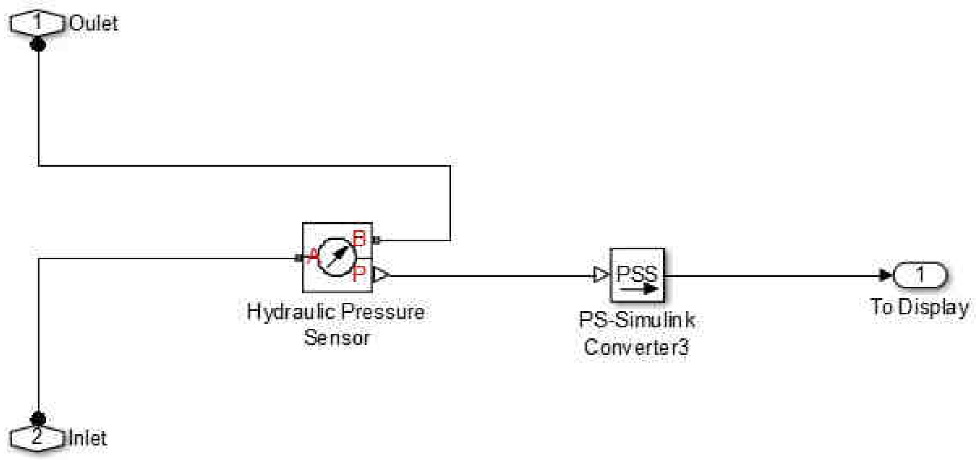


Figure 3.25: Pressure Sensor Subsystem

Connections A and B are conserving hydraulic ports connecting the sensor to the hydraulic line. Connection P is a physical signal port that outputs the pressure value for display. The sensor positive direction is from A to B.

3.13.3. The two flow rate sensors were modeled using the flow sensor subsystem as illustrated in figure 3.26.

The Hydraulic Flow Rate Sensor block represents an ideal flow meter, that is, a device that converts volumetric flow rate through a hydraulic line into a control signal proportional to this flow rate. The sensor is ideal because it does not account for inertia, friction, delays, pressure loss, and so on [11].

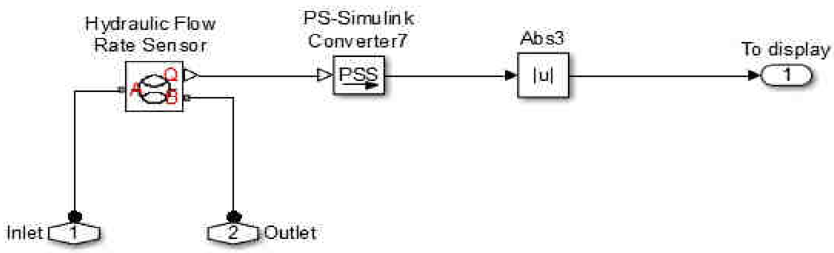


Figure 3.26: Flow Sensor subsystem

Connections A and B are conserving hydraulic ports connecting the sensor to the hydraulic line. Connection Q is a physical signal port that outputs the flow rate value for display. The sensor positive direction is from A to B. This means that the flow rate is positive if it flows from A to B [10].

3.14. Graphical User Interface

GUI is a graphical display in one or more windows containing controls, called components that enable a user to perform interactive tasks.

GUI components can include menus, toolbars, push buttons, radio buttons, list boxes, and sliders—just to name a few. GUIs created using MATLAB® tools can also perform any type of computation, read and write data files, communicate with other GUIs, and display data as tables or as plots.

The GUI is composed of two files:

- A FIGURE-file with extension .fig that contains a complete description of the GUI layout and each UI component, such as push buttons, axes, panels, menus, and so on. The FIG-file is a binary file and you cannot modify it except by changing the layout in GUIDE.
- A code file, with extension .m, that initially contains initialization code and templates for some callbacks that control GUI behavior. You generally add callbacks you write for your UI components to this file. As the callbacks are functions, the GUI code file can never be a MATLAB® script.

When GUI is saved for the first time, GUIDE automatically opens the code file in your default editor.

The FIG-file and the code file must have the same name. These two files usually reside in the same folder, and correspond to the tasks of laying out and programming the GUI. When you lay out the GUI in the Layout Editor, your components and layout is stored in the FIG-file. When you program the GUI, your code is stored in the corresponding code file.

It can be noticed that the GUI is analog to HMI concept in the automation field. A GUI is built for power intake gate so the user can run simulation, for raise and lower, and visualize the simulation results in one place. It consists of pushbuttons for RAISE, normal LOWER and EMERGENCY LOWER and a text fields showing displaying the simulation results for time in seconds and minutes, flow, pressure, position and the intake gate status whether it is opened or closed. So when you press on one of the previous pushbuttons, the GUI will run the model and after completion, it will import the simulation results from (To workspace) blocks found in the model and display it on the corresponding fields. The status field will flash to take user attention as in real control panel of the system.

Also another GUI is developed as a login form to protect the GUI of power intake gate by user name and password.

The FIGURE file of the developed GUIs is shown in figures 3.27& 3.28. And the code filesare found in Appendices.

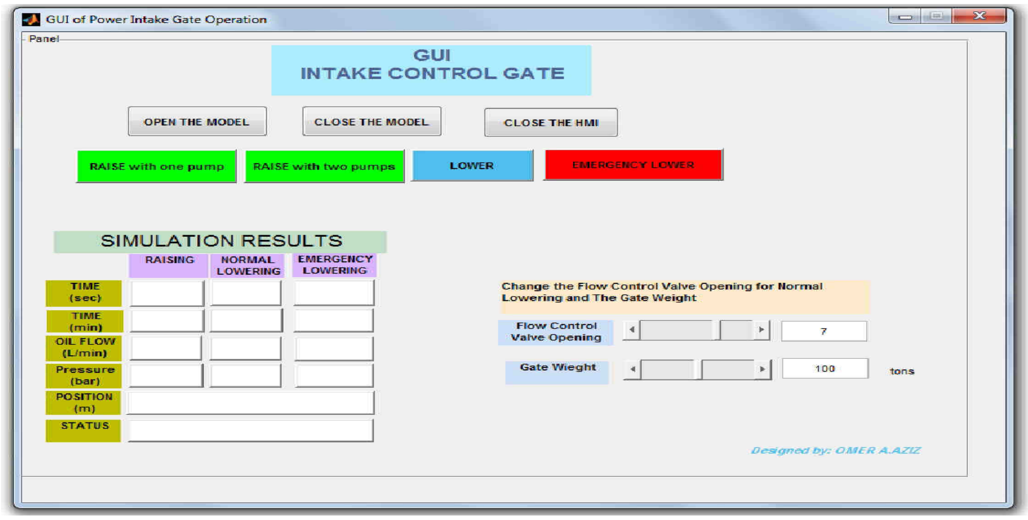


Figure 3.27: Intake Control Gate GUI

Figure 3.28: LOGIN FORM GUI

CHAPTER FOUR

VERIFICATION OF THE MODEL

4.1. Introduction

In Chapter 3, hydraulic control system model is developed. The objective of this chapter is to verify the model experimentally. For verification, measured variables on the real system will be compared with simulation results.

In modeling and simulation environment, all system variables can be measured. However the situation is different for real world. On the real system, it is nearly impossible to measure some of the variables, which are available in the model, due to installation problems, space allocation, improper design, lack of proper measurement instruments and data acquisition system and so on [12].

On the real system, it is nearly impossible to measure the oil flow rate, which is available in the model, due to installation problems and space allocation.

In this study, measurements are done directly on the real system and measured variables are:

- Raising and lowering time in seconds and minutes.
- Oil pressure of raising and lowering in bar.
- Gate position in meters.

Oil flow cannot be measured as mentioned before but it is stated in the hydraulic circuit diagram.

Measurement instruments and the data acquisition system used in the test will be clarified. Afterwards, real system and simulation model results are compared.

Also the simulation and experimental results will be compared against the required values which are available in the quality control data book.

4.2. Measuring Instruments and Data Acquisition

As mentioned in Chapter 2 the hydraulic control system is equipped a position sensor and pressure sensor which are the variables to be measured.

The type of position sensor used is Magnetostrictive Position Sensor form MTS sensors group and has a high repeatability which is a common required characteristic in selection of sensors.

Using the unique magnetorestrictive principle, which MTS pioneered, the sensor precisely senses the position of an external magnet, which is moving with the cylinder piston, through the housing wall to measure displacements with a high degree of resolution (see figure 4.1). The non-contact sensing eliminates the wear, noise and erroneous signal problems and guarantees the best durability without any recalibration. The output of the sensor is an analog signal of range 4 – 20 mA [13].

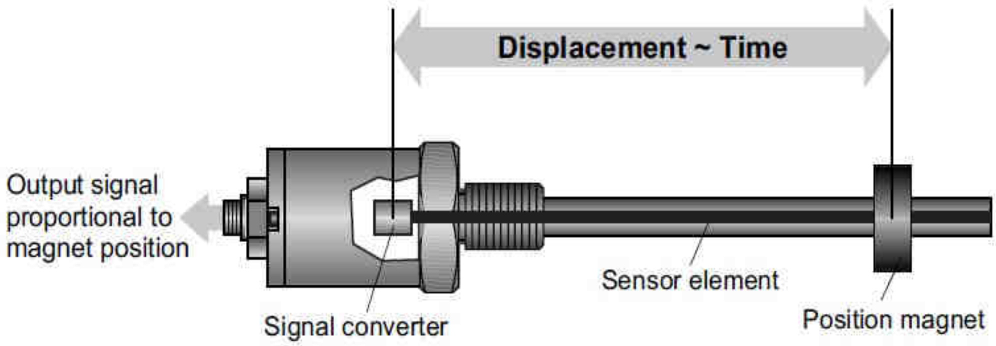


Figure 4.1: Position Sensor [13]

This signal is then fed back to the PLC controller and can be monitored on the local control panel HMI (see figure 4.2)

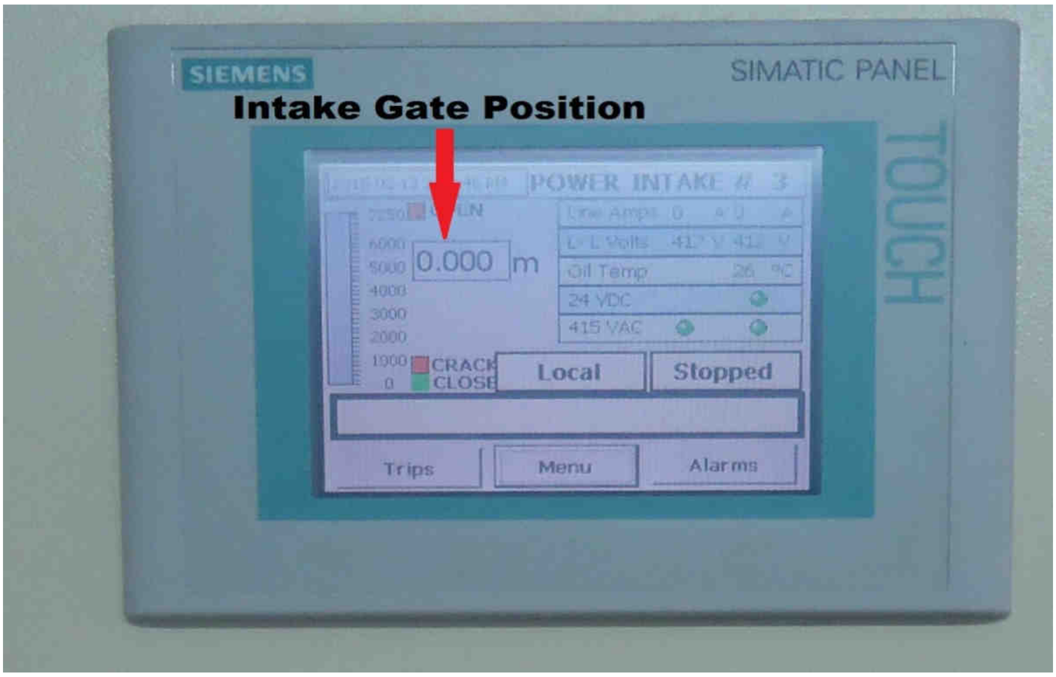


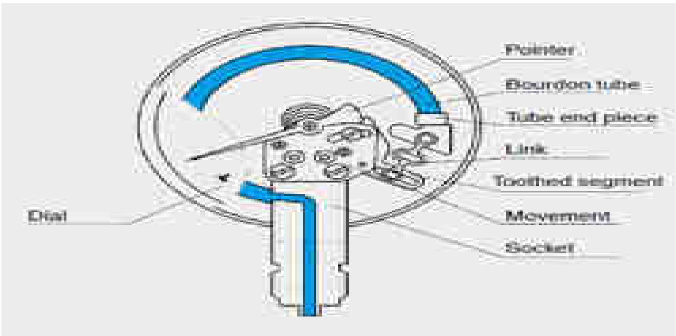
Figure 4.2: Intake Gate Position Signal on HMI

Also the time of the gate raising and lowering can be monitored at sequence of events in the CCR workstation. The sequence of events displays all the events occurring in the power plant in real time. It can also store the events and display it whenever needed.

The type of pressure sensor used is Bourdon tube pressure gauge. Bourdon tube pressure gauges are used for the measurement of relative pressures 0.6- 7,000 bar. They are classified as mechanical pressure measuring instruments, and thus operate without any electrical power and so it cannot be displayed, as in position signal case, neither at Local control panel HMI nor at the CCR workstation.

Bourdon tubes are radially formed tubes with an oval cross-section. The pressure of the measuring medium acts on the inside of the tube and produces a motion in the non-clamped end of the tube. This motion is the measure of the pressure and is indicated via the movement (see figure 4.3a).

The type Bourdon gauge used in this system with a measuring range of 0 to 25 Mpa or 0 to 250 bar (see figure 4.3b) [13].



(a) Sketch [11]

(b) In the Real system

Figure 4.3: Bourdon Tube Pressure Gauge

4.3. Model Verification

There was no need to shut down the units and do the experiments on the intake gates, which will cost a lot and cause a loss of generated power and since there are experimental results which recorded during the commissioning of the hydraulic system of power intake gates.In addition to that the simulation results will compared against the required values for all the results which are available together with the commissioning results in the quality control data book. Also there is an archiving server in the CCR that records all the events occurring in the power plant, including raising and lowering of the intake gates, which can be referred to at any time.

As in chapter 2, the intake gates are divided into two different groups, based on the weight of the gates, raising and lowering time.

Group 1 consists of the gates for units from 1 to 3.

Group 2 consists of the gates for units from 4 to 5.

So Measurements of two cases from Unit 1 and unit 6 will be taken as samples for group 1 and group 2, respectively. Furthermore the relation between the gate weight, raising and lowering time, pressure and oil flow will be analyzed.

4.3.1. Case 1

In case of unit 1, the gate weight is set to100 tons, flow control valve opening scale for normal lowering is set to 7.5 and flow control valve opening scale for emergency lowering is set to 5.1 in the GUI.

On pressing the 'RAISE with one pump' pushbutton on the GUI of power intake gate, the simulation model will run automatically according to the parameters of the raising, which is found in the M-file ICG_Raising. All simulation results will appear automatically on the GUI in the corresponding fields after simulation finished (see figure 4.7). Figure 4.4 shows all the plotted simulation results for 'RAISE with one pump' operation.

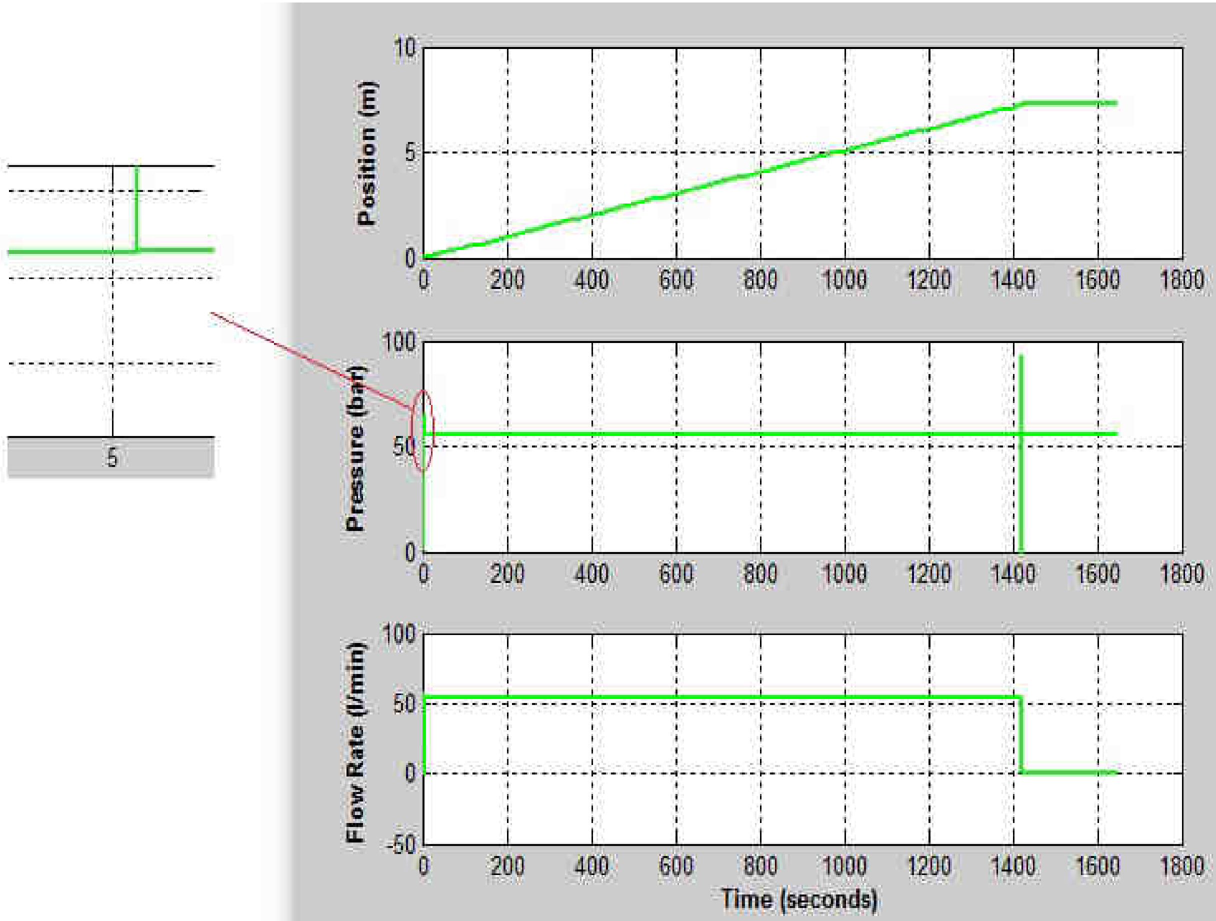


Figure 4.4: Simulation results Plots for RAISING unit 1 gate

On pressing the ‘NORMAL LOWER’ pushbutton on the GUI of power intake gate, the simulation model will run automatically according to the parameters of the normal lowering, which is found in the M-file ICG_Lowering.m. All simulation results will appear automatically on the GUI in the corresponding fields after simulation finished (see figure 4.7). Figure 4.5 shows all the plotted simulation results for ‘NORMAL LOWER’ operation.

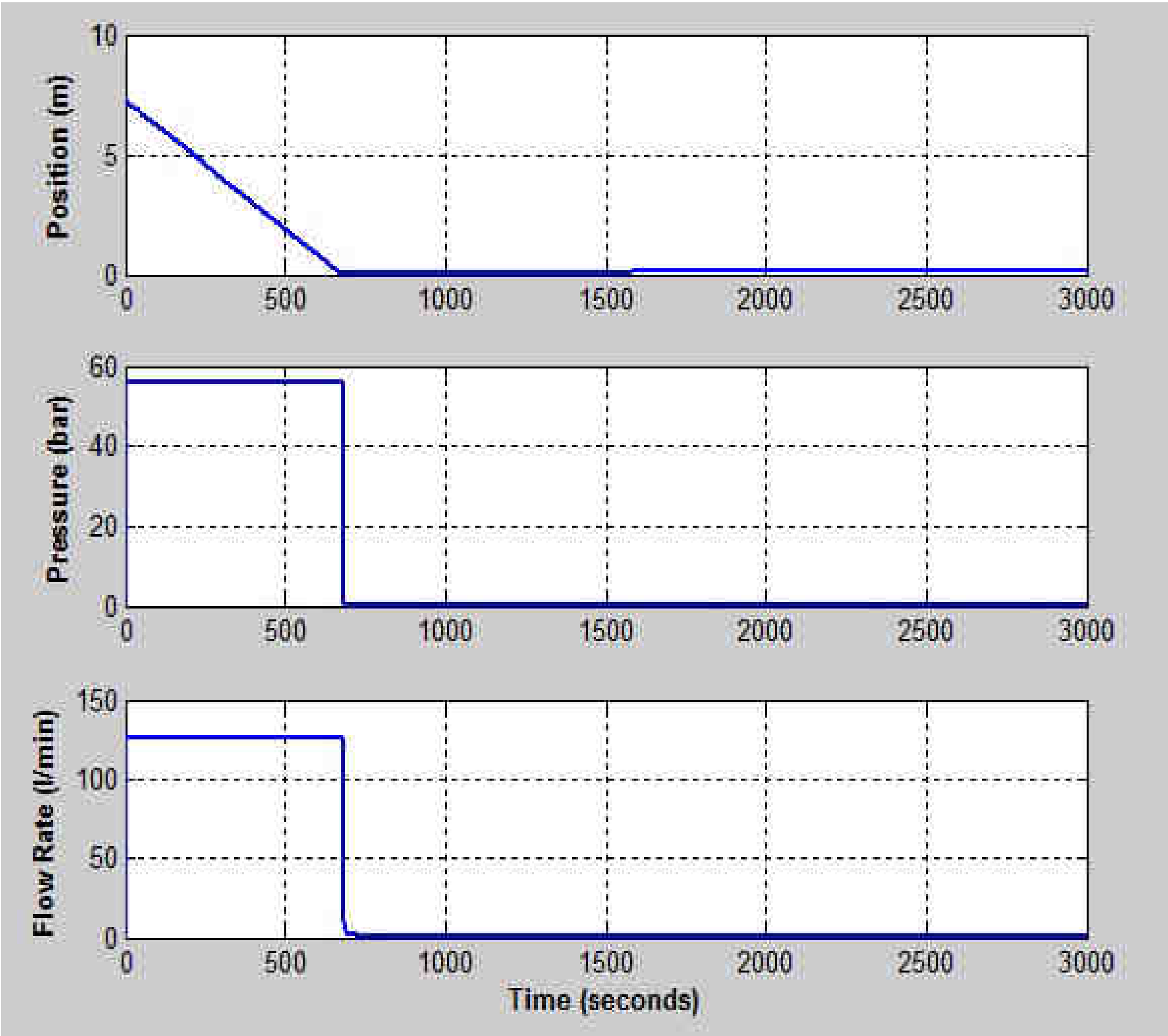


Figure 4.5: Simulation results Plots for NORMAL LOWERING of unit 1 gate

On pressing the ‘EMERGENCY LOWER’ pushbutton on the GUI of power intake gate, the simulation model will run automatically according to the parameters of the emergency lowering, which is found in the M-file ICG_Emerg_Lowering. All simulation results will appear automatically on the GUI in the corresponding fields after simulation finished (see figure 4.7). Figure 4.6 shows all the plotted simulation results for ‘EMERGENCY LOWER’ operation.

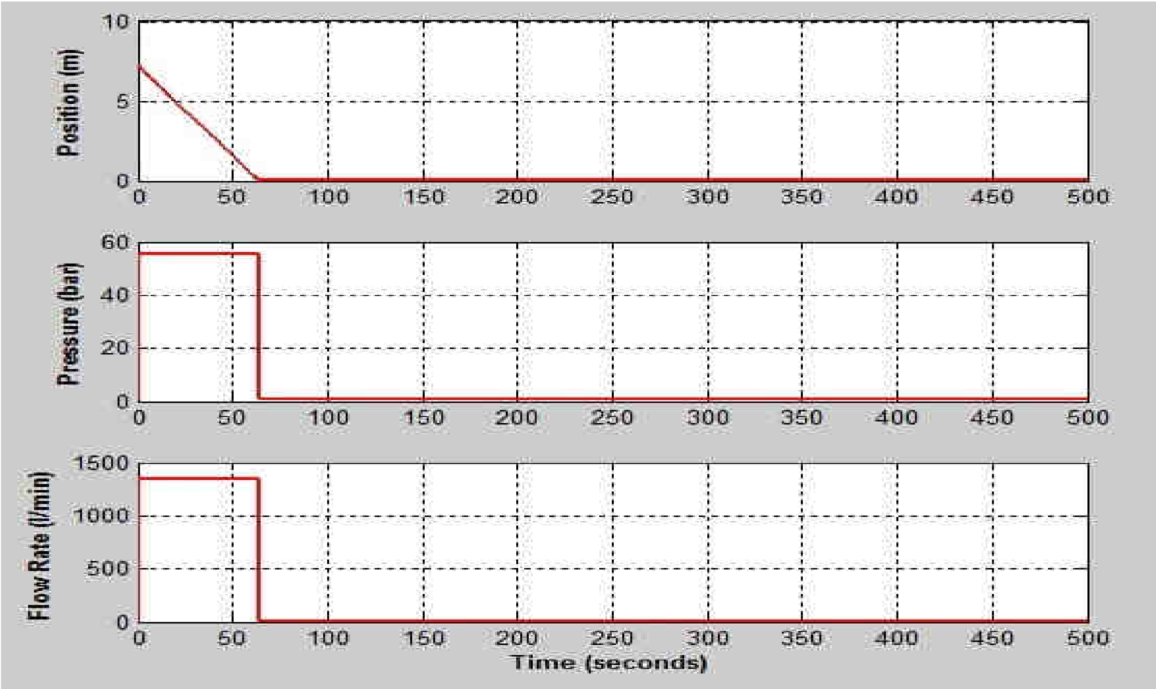


Figure 4.6: Simulation results Plots for EMERGENCY LOWERING of unit 1 gate

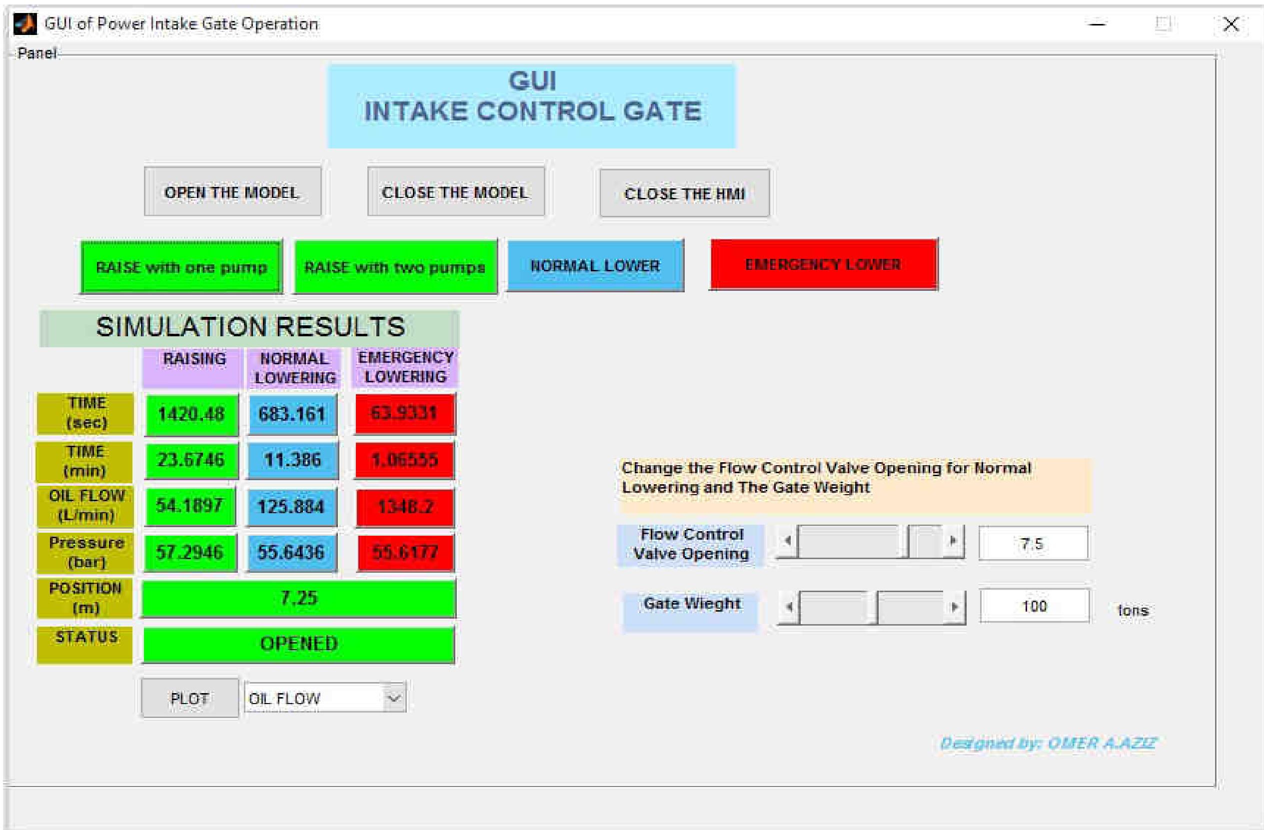


Figure 4.7: Simulation results on GUI for unit 1

The experimental results and required values are recordedin the quality control data book except the oil flow which is stated in the original hydraulic circuit diagram. Table 4.1 shows a comparison between the experimental and simulation results together against the required values.

Table 4.1: Comparison between experimental and real results for unit1 together with the required values

Results	RAISING				NORMAL LOWERING				EMERGENCY LOWERING			
	Exp	Sim	Required values	Error %	Exp	Sim	Required values	Error %	Exp	Sim	Required values	Error %
Time (min:sec)	21:25	23:40	≤ 24 ±1 sec	10.5	10:48	11:23	≤ 11 ± 30sec	5.5	48	01:04	≤ 1 ± 5sec	33.8
Oil flow (l/min)	-	54.2	56	3.2	-	125.9	109	15.5	-	1348.2	1304	3.4
Pressure (bar)	55	57.3	≤ 80	4.2	55	55.64	≤ 80	1.2	55	55.62	≤ 80	1.1
Position (m)	7.25	7.25	7.25	0	7.25	7.25	7.25	0	7.25	7.25	7.25	0

Simulation and experimental results were acceptable compared to the required values. The position was linearly progressing with the time, the pressure and oil flow were steady during all simulation processes. It can be noticed that there are spikes overshooting and undershooting in the oil pressure in the raising simulation plot at t=5and t=1420 s. Theses spikes overshooting and undershooting could not be noticed on the real system, may be due to the low response time of the pressure gauge. The cause of overshooting at t=5 s, is due to the fact that after the oil circulation period expired, the 2/2 way DCV will switch its position suddenly and the full load will be applied on the pump. The cause of overshooting at t=1420 s is due to the hydraulic pump is still working when the piston reaches the top end of the cylinder and hence oil pressure will build up. After pump stop, the check valves will take a time to close and during this time ports A and B of the pressure gauge will be opened to the main oil tank at atmospheric pressure which will cause pressure undershooting. After closing of check valves the pressure will build up again.

The error in the result of emergency lowering time is relatively large but it does meet with the criteria of the required value.

4.3.2. Case 2

In the case of unit 6, the gate weight is set to 36 tons, flow control valve opening scale for normal lowering is set to 7.5and flow control valve opening scale for emergency lowering is set to 8.3in the GUI.

When pressing the 'RAISE with two pump' pushbutton on the GUI of power intake gate, the simulation model will run automatically according to the parameters of the normal lowering, which is found in the M-file ICG_Raising. All simulation results will appear automatically on the GUI in the corresponding fields after simulation finished (see figure 4.11). Figure 4.8 shows all the plotted simulation results for 'RAISE with two pump' operation.

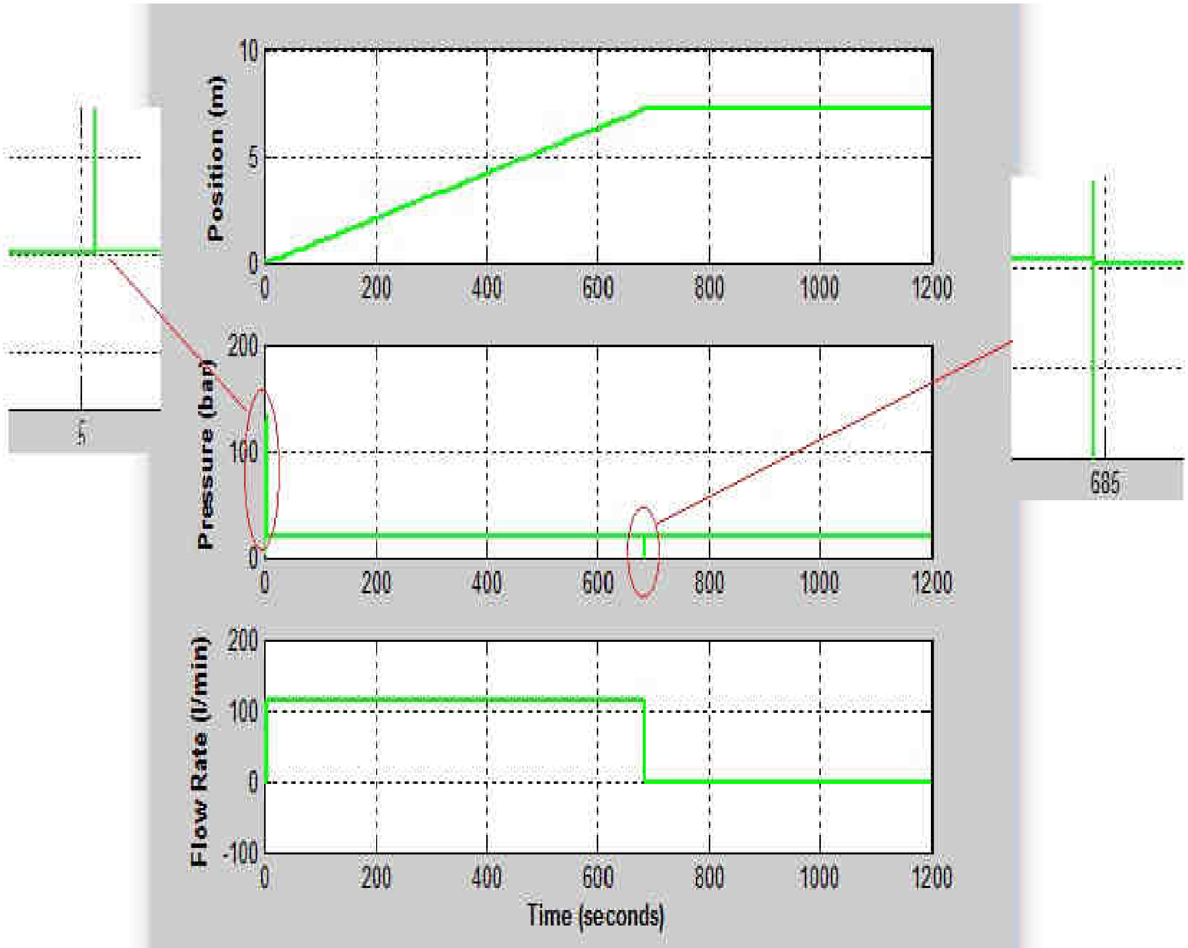


Figure 4.8: Simulation results Plots for RAISING unit 6 gate

When pressing the ‘NORMAL LOWER’ pushbutton on the GUI of power intake gate, the simulation model will run automatically according to the parameters of the normal lowering, which is found in the M-file ICG_Lowering.m. All simulation results will appear automatically on the GUI in the corresponding fields after simulation finished (see figure 4.11). Figure 4.9 shows all the plotted simulation results for ‘NORMAL LOWER’ operation.

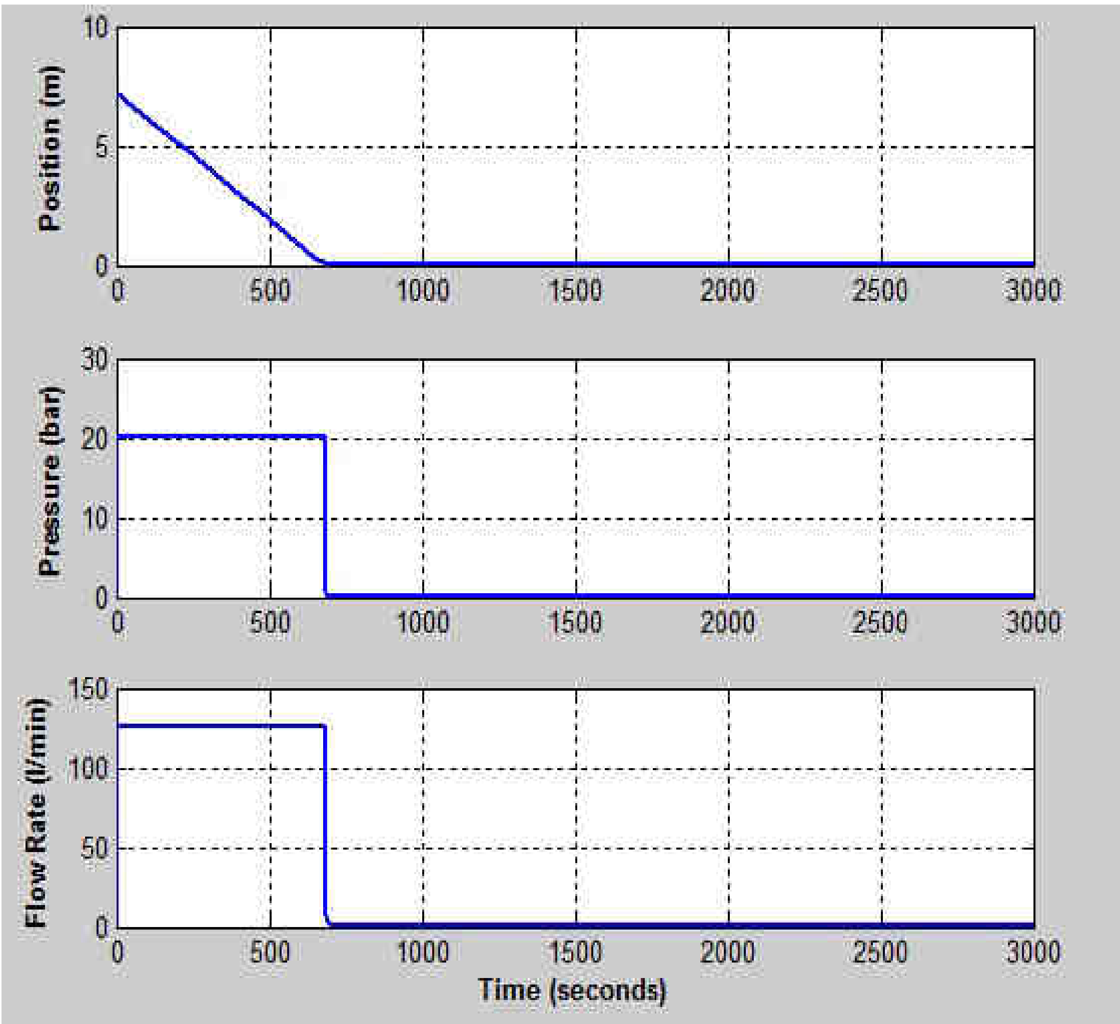


Figure 4.9: Simulation results Plots for NORMAL LOWERING of unit 6 gate

When pressing the ‘EMERGENCY LOWER’ pushbutton on the GUI of power intake gate, the simulation model will run automatically according to the parameters of the emergency lowering, which is found in the M-file ICG_Emerg_Lowering. All simulation results will appear automatically on the GUI in the corresponding fields after simulation finished (see figure 4.11). Figure 4.10 shows all the plotted simulation results for ‘EMERGENCY LOWER’ operation.

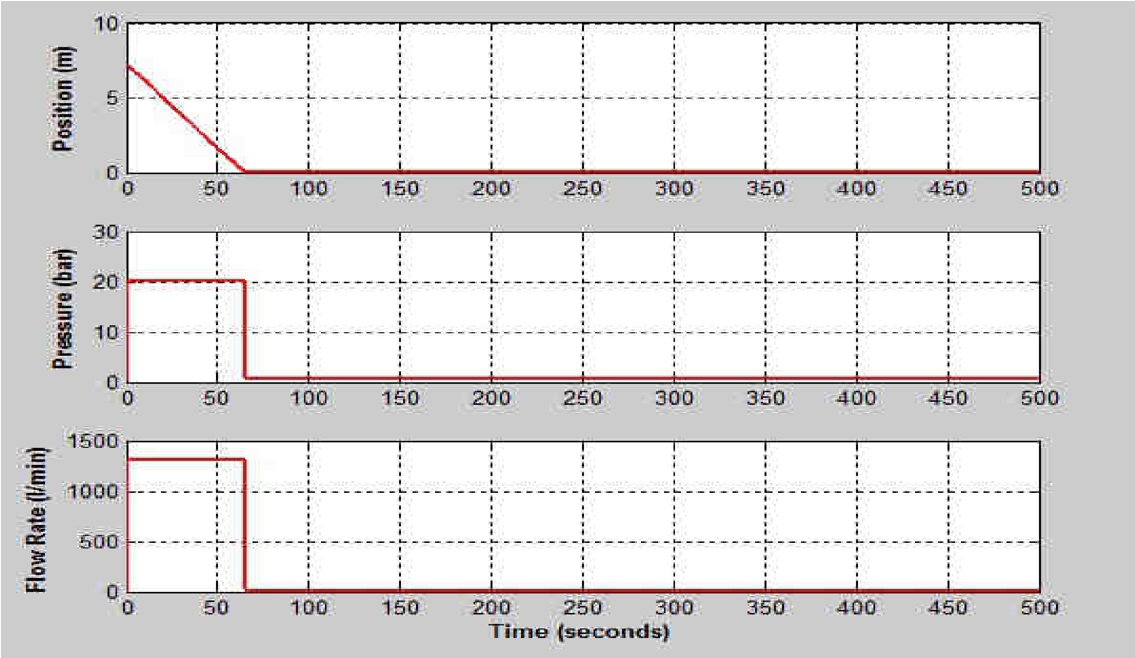


Figure 4.10: Simulation results Plots for EMERGENCY LOWERING of unit 1 gate

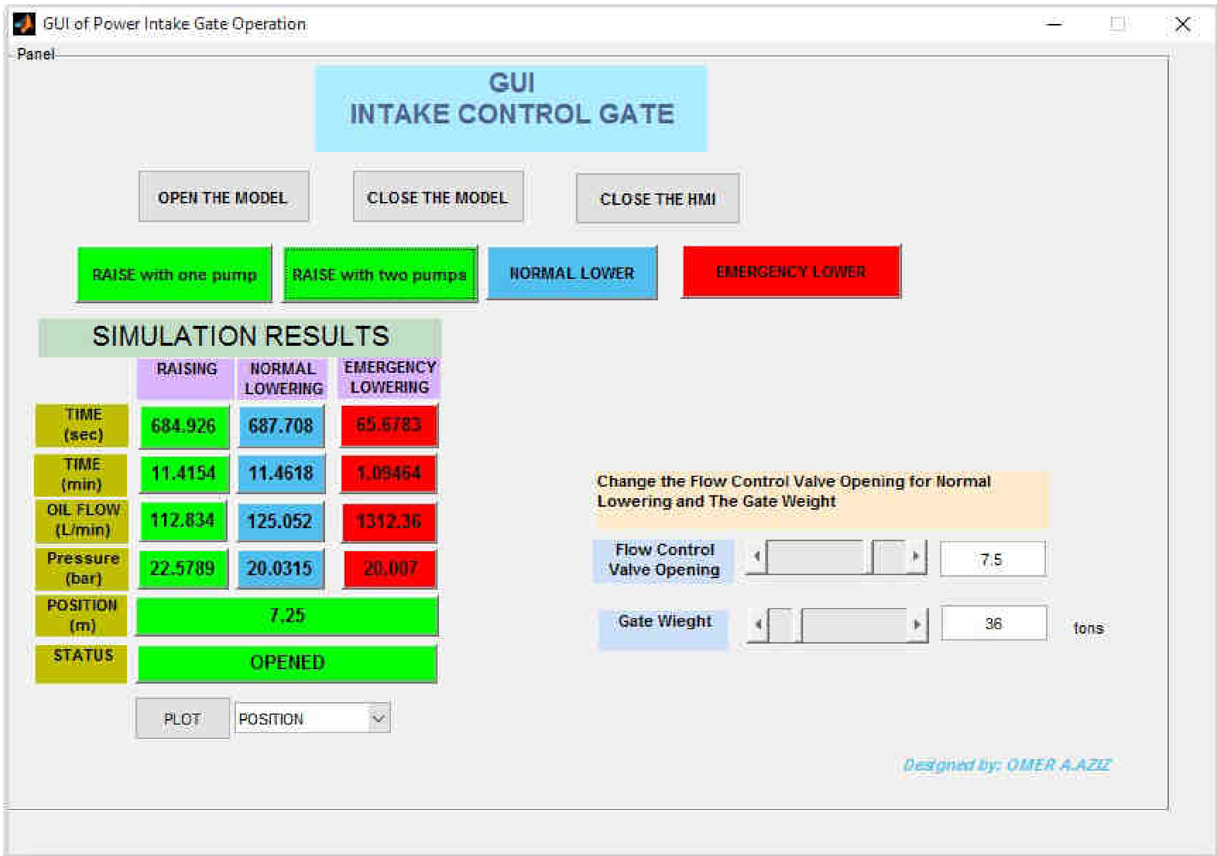


Figure 4.11: Simulation results on GUI for unit 6

The experimental results and required values are recorded in the quality control data book except the oil flow which is stated in the original hydraulic circuit diagram. Table 4.2 shows a comparison between the experimental and simulation results together with the required values.

In this case also Simulation and experimental results were also acceptable compared to the required values. The position was linearly progressing with the time, the pressure and oil flow were steady during all simulation processes. It can be also noticed here, same as in case 1, that there is spike overshooting and undershooting also in the oil pressure in the raising simulation at t=5 and t=684 s and also it occurs for the same reason as in case 1.

Also the oil flow has doubled in this case since the two pumps are working together.

The error in result of emergency lowering time is relatively large but it does meet with the criteria of the required value.

Table 4.2: Comparison between experimental and real results for unit6 together with the required values

Results	RAISING				NORMAL LOWERING				EMERGENCY LOWERING			
	Exp	Sim	Required values	Error %	Exp	Sim	Required values	Error %	Exp	Sim	Required values	Error %
Time (min:sec)	10:30	11:24	≤ 12 ±1 sec	8.7	10:28	11:27	≤ 11 ± 30 sec	9.5	49	01:05	≤ 1 ± 5sec	32.9
Oil flow (l/min)	-	112.8	112	0.7	-	125	109	14.7	-	1312	1304	0.6
Pressure (bar)	25	22.58	≤ 80	9.7	25	20.03	≤ 80	19.9	25	20.03	≤ 80	19.9
Position (m)	7.25	7.25	7.25	0.0	7.25	7.25	7.25	0	7.25	7.25	7.25	0

4.3.3. Relation between gate weight , system pressure and oil flow for raising with one pump

The gate weight was varied from 10 to 200 tons using the GUI and then simulation was done for raising with one pump. The pressure, oil flow and raising time were recorded and plotted against the gate weight.

It can be noticed that the raising time and pump oil flow is nearly independent of the gate weight. This is reasonable if it is known that the raising time depends on the oil flow which depends on oil displacement of the pump and the last one is fixed. This can be verified by raising the intake gate of unit 6 in case 2, which is 36 tons, by one pump and same simulation results as in case 1, except for the raising pressure, will be obtained. The simulation results can be also found in the plot at gate weight=36 tons (see figure 4.12).

Also the results conforms with the characteristic curve of the pump used in the hydraulic system where there also the oil flow drops a little as the pressure increased (see figure 4.13).

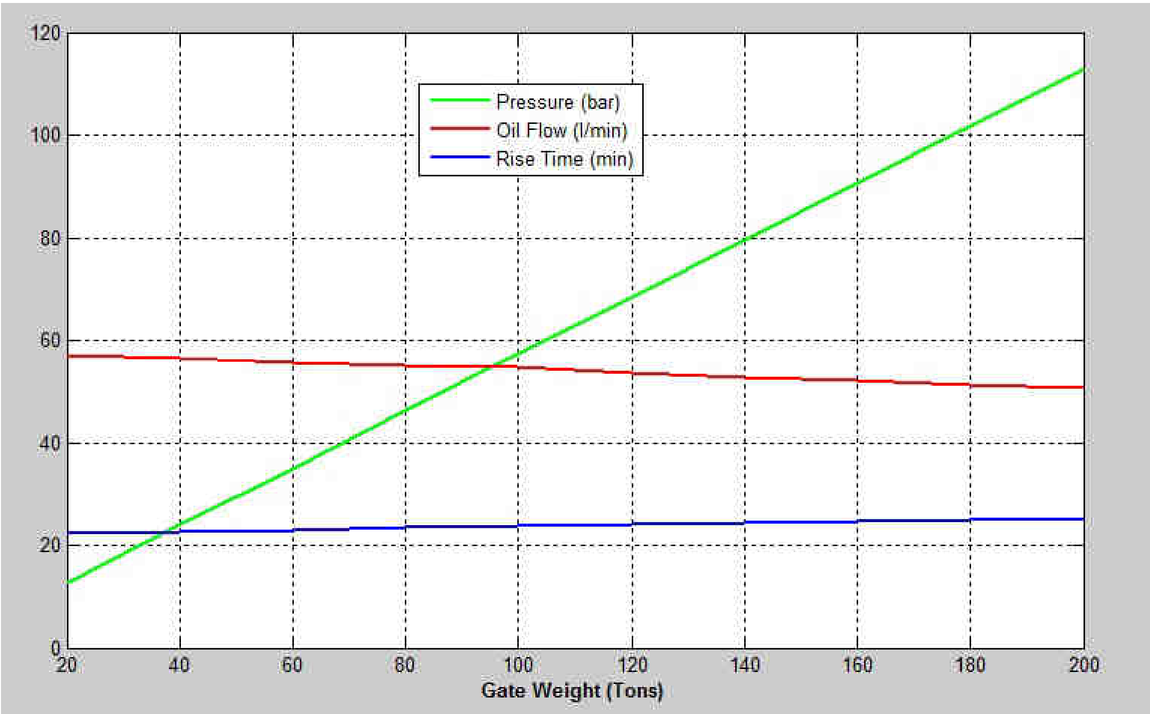


Figure 4.12: Relation between gate weight, pressure, and oil flow and rise time for raising with one pump

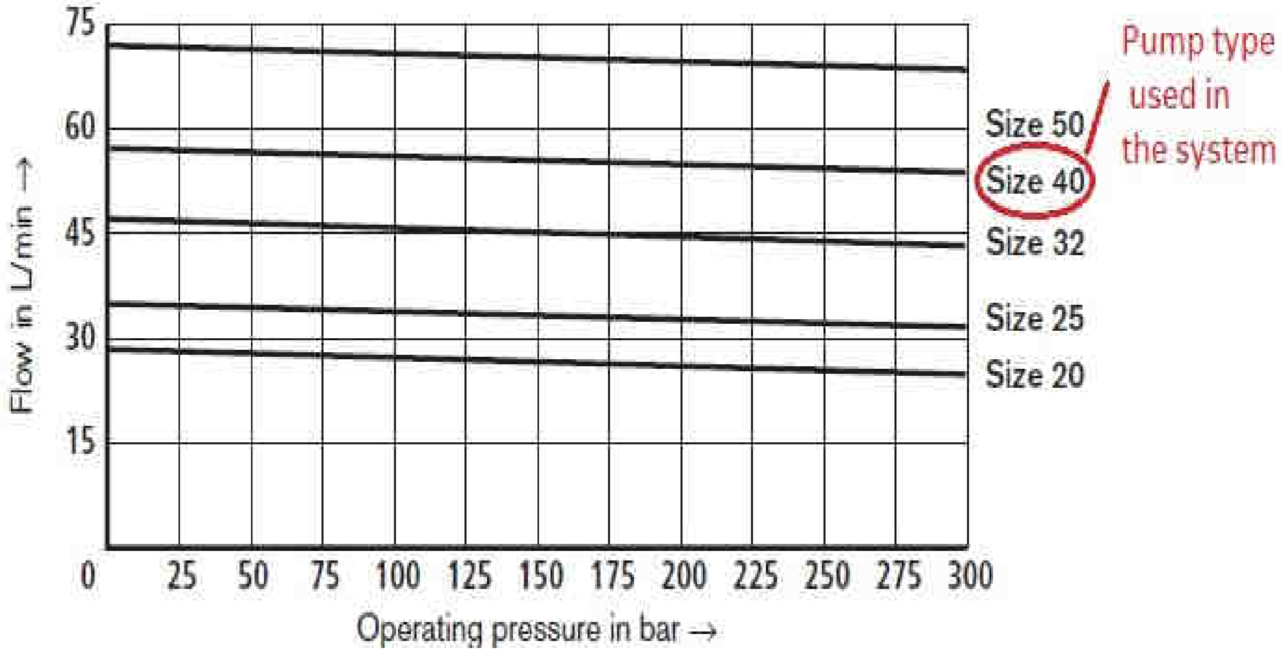


Figure 4.13: Characteristic curve of the used pump

4.3.4. Relation between gate weight , system pressure and oil flow for raising with two pump

The gate weight was varied from 10 to 200 tons using the GUI and then simulation was done for raising with two pumps. The pressure, oil flow and raising time were recorded and plotted against the gate weight.

Also It can be noticed that the raising time and pump oil flow is nearly independent of the gate weight. This is reasonable also for the same reason in the previous paragraph. This can be verified by raising the intake gate of unit 1 in case 1, which is 100 tons, by one pump. Same simulation results as in case 1, except for the raising pressure, will be obtained. The simulation results can be also found in the plot at gate weight=100 tons (see figure 4.14).

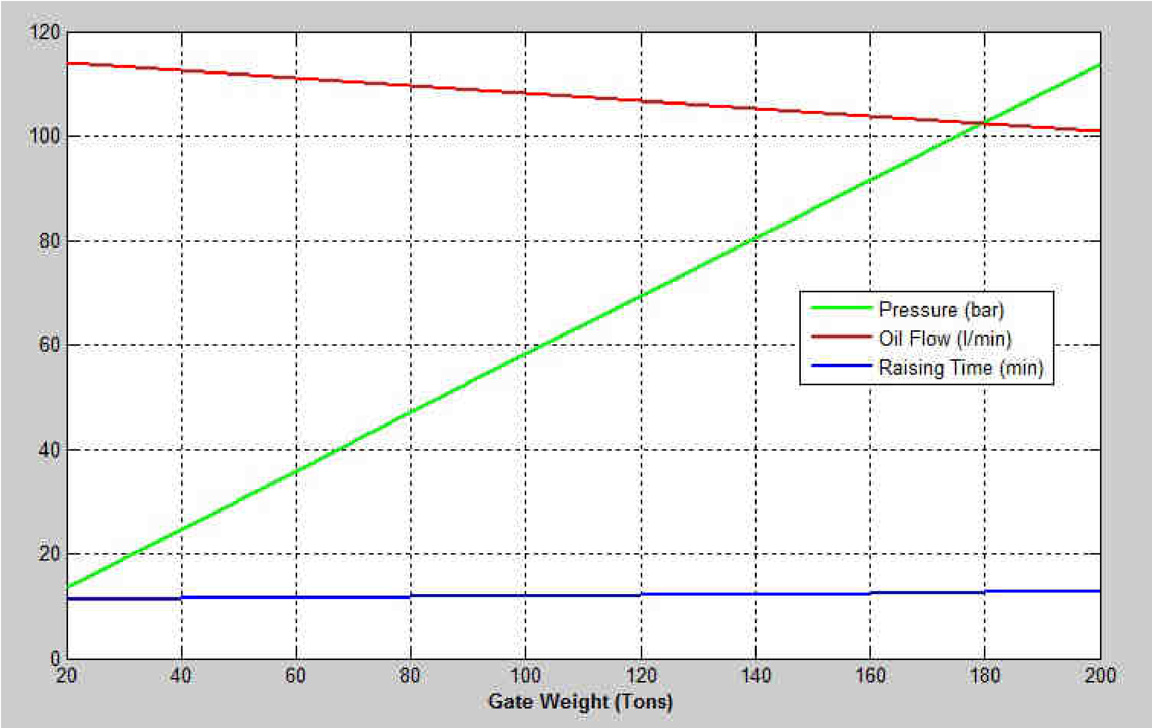


Figure 4.14: Relation between gate weight, pressure, and oil flow and rise time for raising with two pump

4.3.5. Relation between gate weight , system pressure and oil flow for Normal Lowering

The gate weight was varied from 10 to 200 tons, the flow control valve opening for normal lowering was fixed at 7.5 and then simulation was done for Normal Lowering. The pressure, oil flow and normal lowering time were recorded and plotted against the gate weight (see figure 4.15).

It can be noticed that the normal lowering time is nearly independent of the gate weight at a fixed opening of normal lowering flow control valve. This is also reasonable since the type of flow control valve is pressure compensated FCV which was described in chapter two.

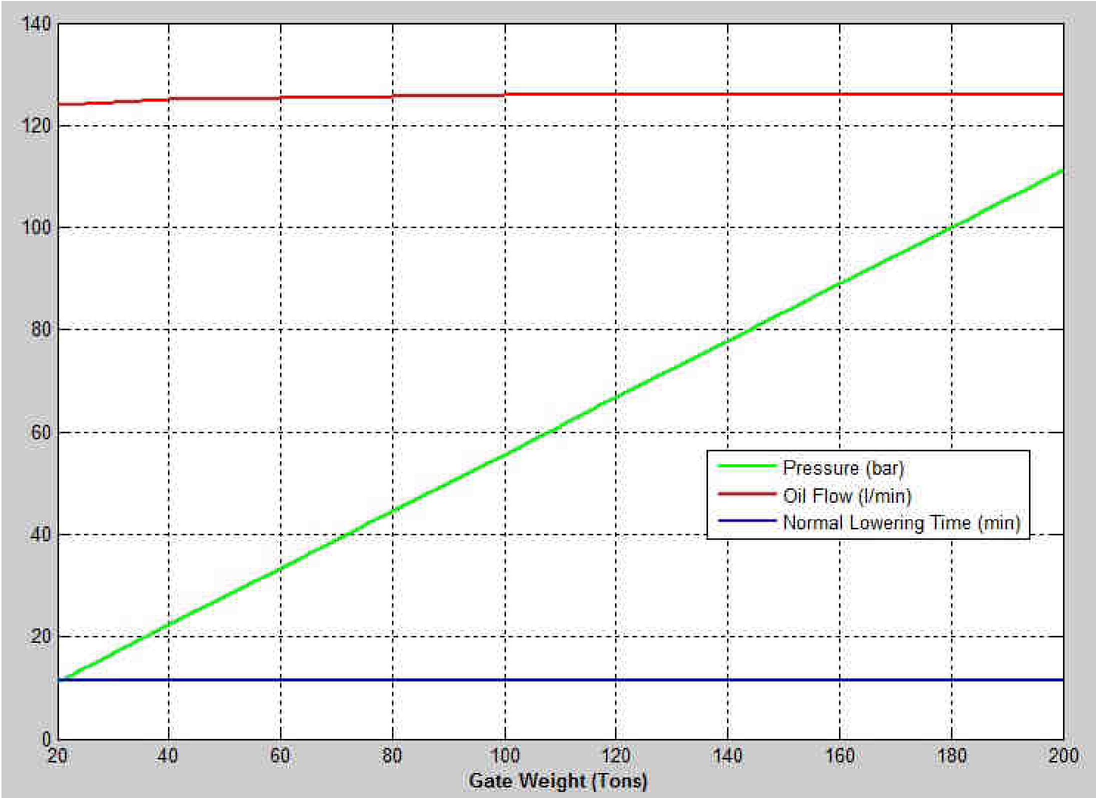


Figure 4.15: Relation between gate weight, pressure, and oil flow and rise time for normal lowering

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1. Conclusion

This thesis focuses on modeling of a hydraulic control system of power intake gates in Rosiers hydro powerplant. MATLAB/Simulink® software is used to develop the model.
Instead of deriving the equations of hydraulic and mechanical components, physical modeling toolboxes of MATLAB (SimHydraulics® and SimMechanics™) are used to model the system components. Standard blocks in SimHydraulics® library are used for modeling hydraulic pumps, hydraulic fluid, check, relief, pressure and directional valves.
Measurements were gathered from previouslyrecorded values during the commissioning in the quality control data book of ICG which is available in Rosiers Hydro power plant documentation. Comparing the simulation results with the experimental data reveals that the model is reasonably accurate.
In addition, the model is utilized in order to analyze the relation between the gate weight and lowering and raising time.

Cushioning Block was not included in the model, since there is no available data for it and trial and error method followed many times to find its parameters but resulted in suspending MATLAB software and caused failure of computation many times. Knowing that the cushioning device acts only on the last 100 mm of the cylinder piston stroke and by comparing it to the full 7.25 m stroke, it is of no significant effect on the simulation results. But its existence is very important inthe real system to avoid hitting of the cylinder top and bottomends by the piston.

The assumption of neglecting friction was also reasonably correct since the simulation results were acceptable compared to the experimental results and required values.

MATLAB's GUI capabilities were very useful tool as well. GUI allowed the user to vary the gate weight and the opening of the flow control valve for normal lowering, which facilitate investigating of their effect on raising and lowering time of intake gate.

In conclusion, a MATLAB/Simulink® model for hydraulic control system is developed in this work. Prepared model is verified by comparison of simulation and experimental results. The Developed model can be used as base model and mechanical model for the gate can be integrated to it. Finally, points that are open to improvement can be determined over the model and changes in the system parameters can be simulated before implementation on the real system.

5.2. Recommendations for Future Work

In the present study, electrical motor is modeled as an ideal angular velocity source. In future investigations, a more comprehensive model including cushioning block and induction motor can be developed to see dynamic effect of the motor startup on the system. Also Mechanical model for the intake gate can be included and integrated to the model using SimMechanics toolbox which is available also in MATLAB software.

Proportional control valve can be used instead of the ON/OFF 2/2 DCV of oil circulation to reduce spike overshooting in the pump oil pressure when the valve switch its position suddenly.

In addition, the parameters of simHydraulic blocks used in the model can be improved by using parameter estimation techniques for more tuning, specifically for emergency flow control valve and cushioning blocks, instead of using trial and error method as described in chapter 3.

Finally it is recommended to develop a model using different simulation software such as automation studio, since there are many of them, and compare it with the developed Matlab/SimHydraulics model.

REFERENCES

1. P and B, Per-Willy Lauvli and Bjørn Victor Lund, 2010. *Modeling, Simulation and Experimentation of a. Hydrostatic Transmission*. M.Sc. in Mechatronics Engineering. Norway: University of Agder.
2. Weinan Cao et al., (2015). Research on Model and Simulation of Hydraulic Lifting System of the Wave Power Generating Platform based on AMESim. *In International Industrial Informatics and Computer Engineering Conference (IIICEC)*. Jakarta.
3. Shrikrishna N. Joshi, “Mechatronics and Manufacturing Automation”; Course lecture notes, Indian Institute of Technology Guwahati.
4. Allen Van Gent, 2011. BASIC HYDRAULICS TRAINING. HYTEC GROUP.
5. Jagadeesha T, “Fluid Power Control”; Course lecture Notes, Indian Institute of Technology Madras.
6. H. Exner and others, 1991. Basic Principles and Components of Fluid Technology. Mannesman Rexroth,
7. HYTEC Operation and Maintenance Manual for Rosiers Power Intake Gates 2013.
8. http://WWW.WIKIPEDIA.COM, last visited on July 2015.
9. Rohner, P, 1995. Industrial Hydraulic Control. Fourth edition. Brisbane: John Wiley & Sons.
10. The Mathworks Inc, 2014, SimHydraulics Reference R2014a.
11. Ersen Torman, 2013. Modeling, Simulation and Experimental Verification of a Hydraulic Fan Drive System. M.Sc. in Mechanical Engineering. Turkey: Middle East Technical University.
12. HYTEC Electrical Components Manual for Rosiers Power Intake Gates 2013.
13. HYTEC Quality Control Data Book for Rosiers Power Intake Gates 2013.

Appendix A

Commissioning sheets and Sequence of Events

HYTEC AFRIPOWER (Pty) Ltd t/a Hytec	CCMD Joint Venture POWER INTAKE GATE SYSTEM	Document No: TP54001038/07 Page 4 of 6 Author: K. Marggraff

4.11	Crack open position set and confirmed	YES / NO	50mm
4.12	Filter blocked indication	YES / NO	✓
4.13	High pressure trip function PS1	12.5 MPa	15,5 MPa
4.14	Low pressure trip function PS 2	1 MPa	1 MPa.
4.15	Set pressure on Pressure relief valve PRV 1 on power pack	14 MPa	16,5 Mpa.
4.16	Set pressure on pressure relief valve on cylinder item 102	YES / NO	16 MPa.
4.17	Set flow control valve for normal lowering	Scale 6-8	7,5 ✓
4.18	TT 1 function Low filter block indication High temperature alarm High High Temperature Trip Temperature Override 80 Deg C	YES / NO	✓ ✓ ✓ ✓ ✓
4.19	LS 1 Confirm oil level switch set points Low Level alarm Low level trip	YES / NO	✓ ✓ ✓
4.20	Drift alarm and Drift leakage indication	YES / NO	✓
4.21	Wiring to terminal box	Pull wire test	✓
4.22	Confirm all pipe clamps are in place	Checked	✓
4.23	All painting and touch up painting done	Checked	✓
4.24	All solenoids and switches wired to junction box	Checked	✓
4.25	Record raising pressure	≤ 8 MPa	5,5 MPa
4.26	Record rising speed	≤ 24 minutes ± 1	21:25
4.27	Record the slow lowering speed	≤ 11 minutes ± 30sec	10:48
4.28	Record the emergency lowering speed up to start of cushioning	≤ 1 minutes ± 5sec	48 sec
4.29	Emergency Stop (Electric motors)	YES / NO	✓
4.30	Lamp Test Function in (Local)	Checked	✓
4.31	Remote Emergency Lower Signal	Checked	✓
4.32	Local Emergency Lower Signal	Checked	✓
4.33	Remote Open Signal	Checked	✓
4.34	Remote Close Signal	Checked	✓
4.35	Remote Penstock Full (Pressure Balanced)	Checked	✓
4.36	Low Oil Alarm Signal to NEC	Checked	✓
4.37	High System Pressure Signal to NEC	Checked	✓

Pre commissioning Procedure Hydraulic System for power Intake Gate KM13-231

Figure A-1: Commission sheet of unit 1 [13]

HYTEC AFRIPOWER (Pty) Ltd t/a Hytec	CCMD Joint Venture POWER INTAKE GATE SYSTEM	Document No: TP54001038/07 Page 4 of 6 Author: K. Marggraff

4.11	Crack open position set and confirmed	~~YES~~ / NO	140mm ✓
4.12	Filter blocked indication	~~YES~~ / NO	✓
4.13	High pressure trip function PS1	8.5 MPa	8,5 MPa
4.14	Low pressure trip function PS 2	1 MPa	1 MPa
4.15	Set pressure on Pressure relief valve PRV 1 on power pack	11 MPa	11 MPa
4.16	Set pressure on pressure relief valve on cylinder item 102	YES / ~~NO~~	9 MPa
4.17	Set flow control valve for normal lowering	Scale 6-8	7,2
4.18	TT 1 function Low filter block indication High temperature alarm High High Temperature Trip Temperature Override 80 Deg C	YES / ~~NO~~	✓ ✓ ✓
4.19	LS 1 Confirm oil level switch set points Low Level alarm Low level trip	YES / ~~NO~~	✓
4.20	Drift alarm and Drift leakage indication	YES / ~~NO~~	✓
4.21	Wiring to terminal box	Pull wire test	✓
4.22	Confirm all pipe clamps are in place	Checked	✓
4.23	All painting and touch up painting done	Checked	✓
4.24	All solenoids and switches wired to junction box	Checked	✓
4.25	Record raising pressure	≤ 8 MPa	2-3 MPa
4.26	Record rising speed	≤ 12 minutes ± 1	10,3 min
4.27	Record the slow lowering speed	≤ 11 minutes ± 30sec	10,28 min
4.28	Record the emergency lowering speed up to start of cushioning	≤ 1 minutes ± 5sec	49 sec
4.29	Emergency Stop (Electric motors)	YES / ~~NO~~	✓
4.30	Lamp Test Function in (Local)	Checked	✓
4.31	Remote Emergency Lower Signal	Checked	✓
4.32	Local Emergency Lower Signal	Checked	✓
4.33	Remote Open Signal	Checked	✓
4.34	Remote Close Signal	Checked	✓
4.35	Remote Penstock Full (Pressure Balanced)	Checked	✓
4.36	Low Oil Alarm Signal to NEC	Checked	✓
4.37	High System Pressure Signal to NEC	Checked	✓

Pre commissioning Procedure Hydraulic System for power Intake Gate KM13-231

Figure A-2: Commission sheet of unit 6 [13]

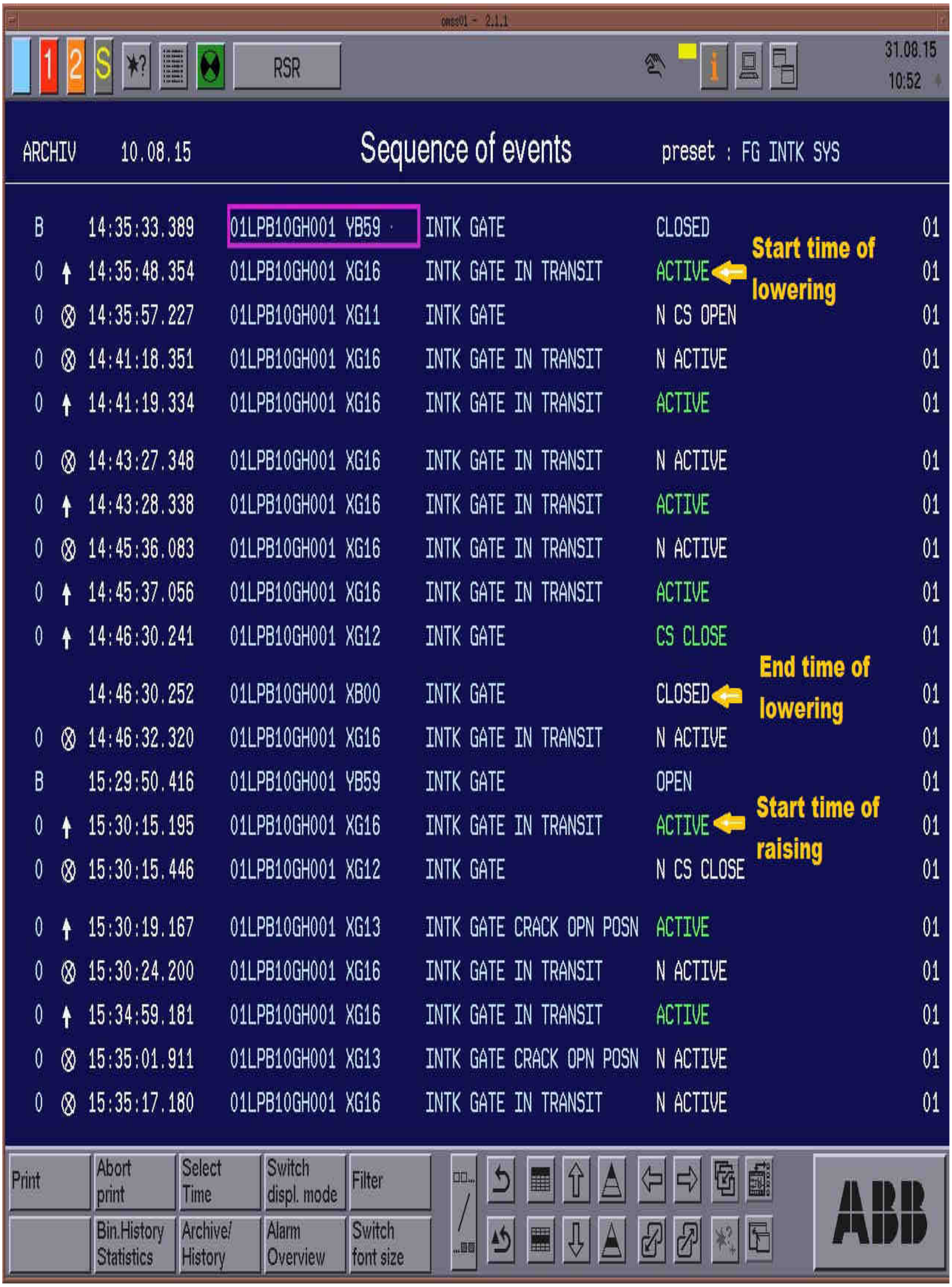


Figure A-3: SOE of start, end of normal lowering and start of raising for unit 1

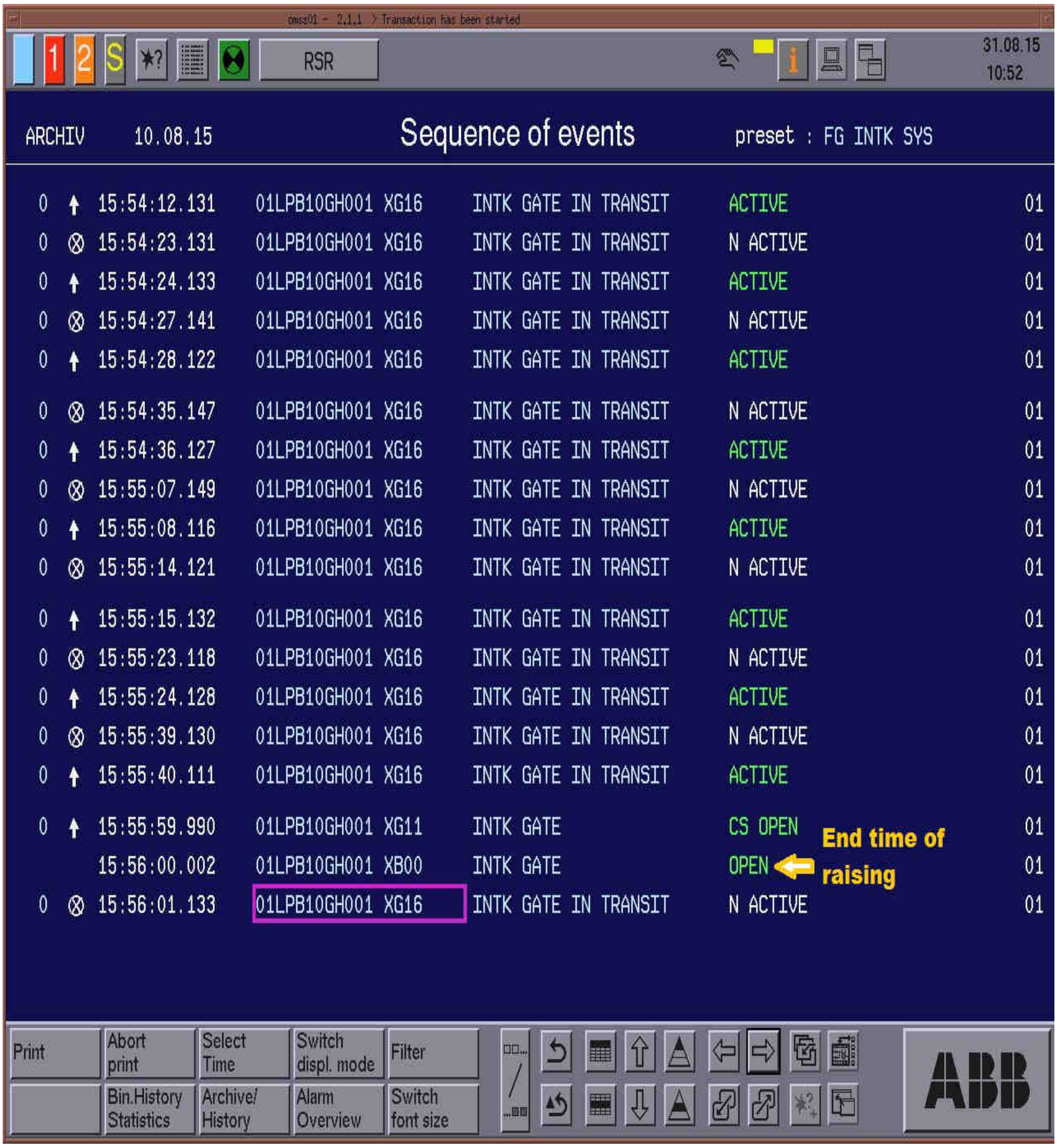


Figure A-4: SOE of end of raising for unit 1

Raising and normal lowering time for unit 1can be calculated from figures A-3 and A-4 as following:

Raising time = End of raising – Start of raising = 15:56:01 – 15:30:15= 25 min + 46 sec
Lowering time = End of lowering – Start of lowering = 14:46:30 – 14:35:48 = 10 min + 42 sec

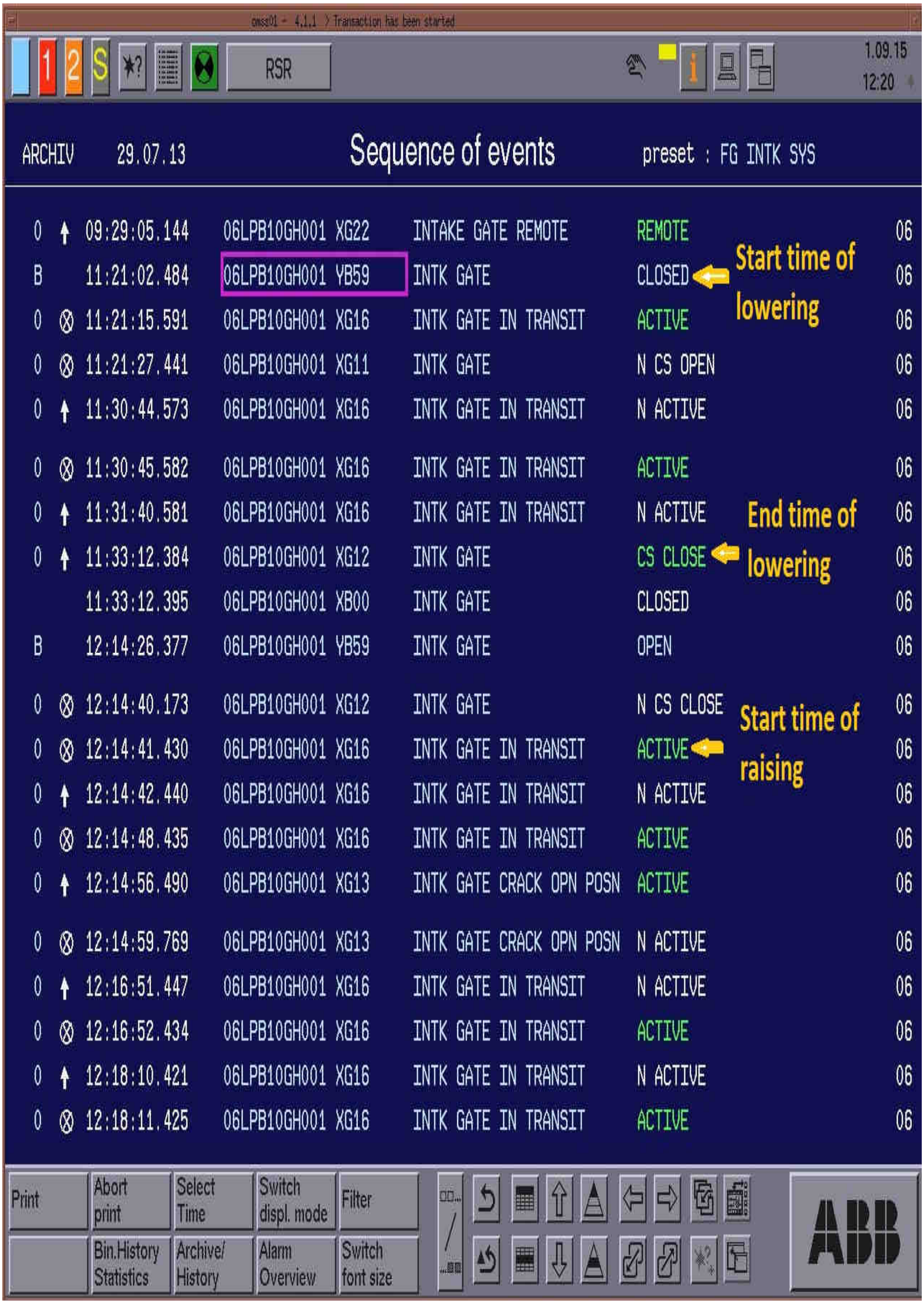


Figure A-5: SOE of start, end of normal lowering and start of raising for unit 6

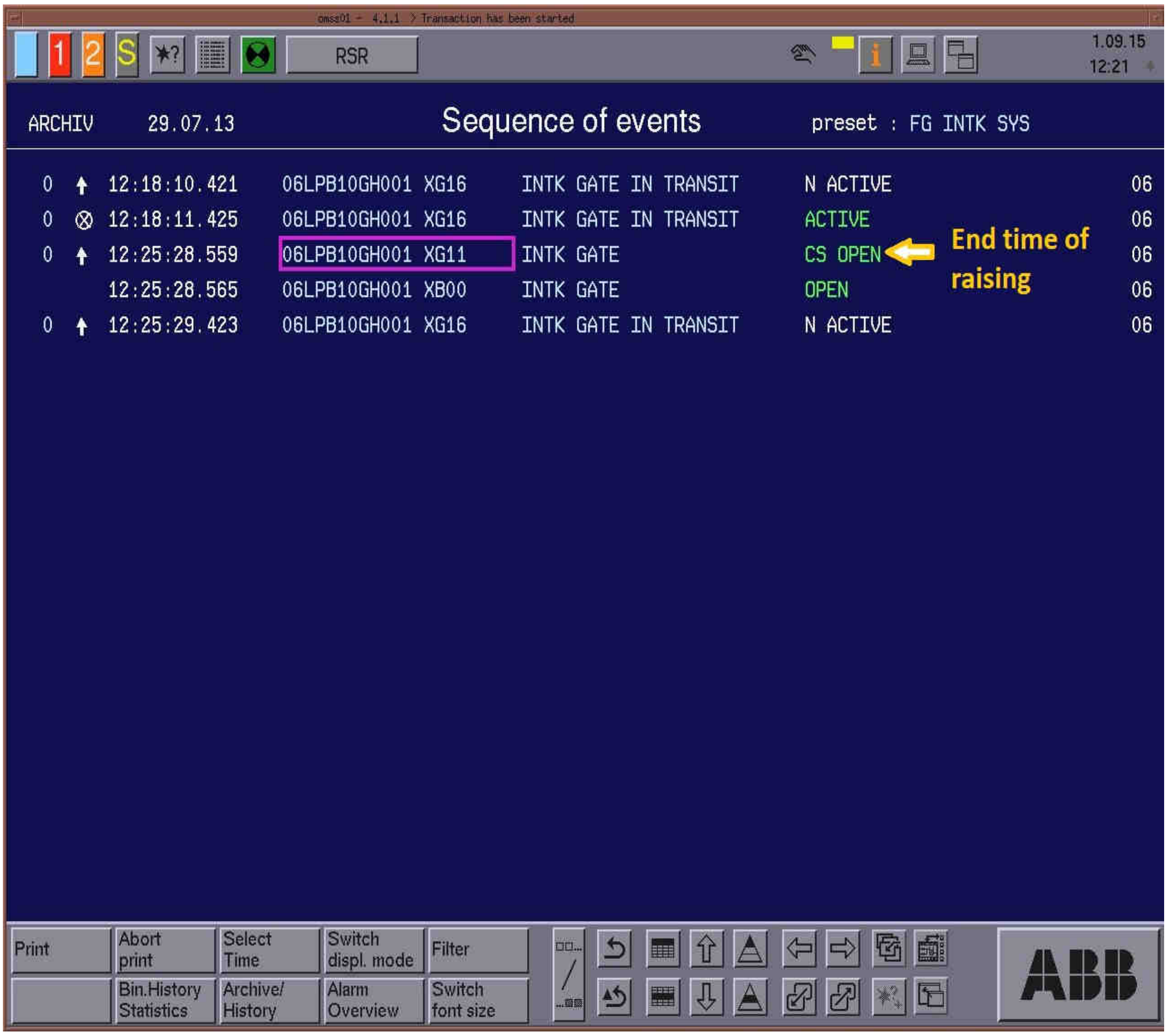


Figure A-6: SOE of end of raising for unit 6

Also raising and normal lowering time for unit 6 can be calculated from figures A-5 and A-6 as following:

Raising time = End of raising – Start of raising = 12:25:28 – 12:14:41 = 10 min + 47 sec

Lowering time = End of lowering – Startof lowering = 11:33:12 – 11:21:15 = 11 min + 57 sec

APPENDIX B

MATLAB CODE FILES

M-File containing main hydraulic system parameters

```
% motor speed or Nominal angular velocity
Ns=1450         %rpm=1450 on raise, =0 on lowering
% hydraulic Fluid Parameters, Fluid type used is TOTAL AZOLLA ZS 46
DENSnom=850       % Nominal Fluid density Kg/m3
Vnom=46          % Nominal Kinematic Viscosity, cSt

% hydraulic Pump Parameters
D=40.1              % Displacement, cm3 per revolution
Eff_V=0.92          % Volumetric efficiency
Eff_TOT=0.8         % Total efficiency
Pnom=70             %70 Nominal pressure, bar
```

M-File containing parameters for RAISING

```
% this file is executed on pressing Button of RAISE with one pump or RAISE with two
pumps in GUI
% Initial distance from cylinder top (cap A)
init_dist=7.25       % 7.25 on raise, 0 on lower
% initial position of position sensor
init_pos=0        % 0 on raise, 7.25 on lower
%Normal Lowering relay DC voltage
Kn=0              %0 on raise, 24 on lower
%Emergency Lowering relay DC voltage
Ke=0              %0 on raise, 24 on lower
% DC voltage of relay for Filtration period
Kr=24             % 24 on raise, 0 on lower
```

M-File containing parameters for Normal Lowering

```
% this file is executed on pressing Button of NORMAL LOWER in GUI
%Initial distance from top (cap A)
init_dist=0       % 7.25 on raise, 0 on lower
% initial position of position sensor
init_pos=7.25       % 0 on raise, 7.25 on lower
%Normal Lowering relay DC voltage
Kn=24             %0 on raise, 24 on lower
%Emergency Lowering relay DC voltage
Ke=0              %0 on raise, 24 on lower
% DC voltage of relay for Filtration period
Kr=0              % 24 on raise, 0 on lower
```

M-File containing parameters for Emergency Lowering simulation

```
 % this file is executed on pressing Button of EMERGENCY LOWER in GUI
%Initial distance from top (cap A)
```

```
init_dist=0       % 7.25 on raise, 0 on lower
% initial position of position sensor
init_pos=7.25       % 0 on raise, 7.25 on lower
%Normal Lowering relay DC voltage
Kn=0           %0 on raise, 24 on lower
%Emergency Lowering relay DC voltage
Ke=24           %0 on raise, 24 on lower
% DC voltage of relay for Filtration period
Kr=0           % 24 on raise, 0 on lower
```

Code file of GUI

```
function varargout = ICG_Panel(varargin)
% ICG_PANEL MATLAB code for ICG_Panel.fig
%      ICG_PANEL, by itself, creates a new ICG_PANEL or raises
the existing
%      singleton*.
%
%      H = ICG_PANEL returns the handle to a new ICG_PANEL or
the handle to
%      the existing singleton*.
%
%      ICG_PANEL('CALLBACK',hObject,eventData,handles,...)
calls the local
%      function named CALLBACK in ICG_PANEL.M with the given
input arguments.
%
%      ICG_PANEL('Property','Value',...) creates a new
ICG_PANEL or raises the
%      existing singleton*.  Starting from the left, property
value pairs are
%      applied to the GUI before ICG_Panel_OpeningFcn gets
called.  An
%      unrecognized property name or invalid value makes
property application
%      stop.  All inputs are passed to ICG_Panel_OpeningFcn
via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI
allows only one
```

```
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help ICG_Panel

% Last Modified by GUIDE v2.5 21-Apr-2015 19:51:34

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
'gui_Singleton',  gui_Singleton, ...
'gui_OpeningFcn', @ICG_Panel_OpeningFcn, ...
'gui_OutputFcn',  @ICG_Panel_OutputFcn, ...
'gui_LayoutFcn',  [] , ...
'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State,
varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before ICG_Panel is made visible.
function ICG_Panel_OpeningFcn(hObject, eventdata, handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
```

```
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
% varargin   command line arguments to ICG_Panel (see
VARARGIN)


% Choose default command line output for ICG_Panel
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes ICG_Panel wait for user response (see UIRESUME)
% uiwait(handles.ICG_HMI);


% --- Outputs from this function are returned to the command
line.
function varargout = ICG_Panel_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see
VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% Ensure that the Simulink model is open
```

```
% --------------------------------------------------------------
function model_open(handles)
% Make sure the diagram is still open
if  isempty(find_system('Name','ICG_SIM_MODEL')),
        open_system('ICG_SIM_MODEL');

        fig = handles.ICG_HMI;
        figure(fig);
end

%endfunction model_open

function RESET_VALS(handles)
    NewStrVal = '';
    NewVal = str2num(NewStrVal);



set(handles.ICG_POS,'String',NewVal,'Backgroundcolor','white')

set(handles.ICG_STATUS,'String',NewVal,'Backgroundcolor','whit
e')

set(handles.RAISING_TIME_SEC,'String',NewVal,'Backgroundcolor'
,'white')

set(handles.TIME_ELAPSED_R,'String',NewVal,'Backgroundcolor','
white')

set(handles.NLOWER_TIME_SEC,'String',NewVal,'Backgroundcolor',
'white')

set(handles.TIME_ELAPSED_NL,'String',NewVal,'Backgroundcolor',
'white')
```

```
set(handles.ELOWER_TIME_SEC,'String',NewVal,'Backgroundcolor',
'white')

set(handles.TIME_ELAPSED_EL,'String',NewVal,'Backgroundcolor',
'white')

set(handles.RAIS_FLOW,'String',NewVal,'Backgroundcolor','white
')

set(handles.NLOWER_FLOW,'String',NewVal,'Backgroundcolor','whi
te')

set(handles.EMLOWER_FLOW,'String',NewVal,'Backgroundcolor','wh
ite')

set(handles.RAIS_PRESS,'String',NewVal,'Backgroundcolor','whit
e')

set(handles.NLOWER_PRESS,'String',NewVal,'Backgroundcolor','wh
ite')

set(handles.EMLOWER_PRESS,'String',NewVal,'Backgroundcolor','w
hite')
%
%  function SolverConfg_msgbox(handles)
%      h=msgbox(['Solver configuration Parameters:       '
%          '                                        '
%          'On Raising:                             '
%          'relative tolerance = 1e-3               '
%          'absolute tolerance = auto               '
%          'On lowering:                            '
%          'relative tolerance = 1e-5               '
%          'absolute tolerance = 1e-6               '
```

```
%          '                                      '],'Solver
Configuration Parameters');
%
%      pause(5)
%      delete(h)

% --- Executes on button press in Open_model.
function Open_model_Callback(hObject, eventdata, handles)
% hObject    handle to Open_model (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
model_open(handles)

% --- Executes on button press in Close_model.
function Close_model_Callback(hObject, eventdata, handles)
% hObject    handle to Close_model (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
    set_param('ICG_SIM_MODEL/NL SPEED SP','Value','7.5')   %
set to defualt value
%set_param('ICG_SIM_MODEL/ELGAIN','Gain','0.001')     % set to
defualt value
    set_param('ICG_SIM_MODEL/GATE_WEIGHT','Value','100') % set
to defualt value
    save_system('ICG_SIM_MODEL')
    close_system('ICG_SIM_MODEL')
    evalin('base','clear all')
    run('Clean_CMD.m')

% function for flashing the intake gate status when opened
function Open_Status_Flashing(handles)
```

```
        Z='green';

set(handles.ICG_STATUS,'String','OPENED','Backgroundcolor',Z)
        pause(1)

set(handles.ICG_STATUS,'String','OPENED','Backgroundcolor','wh
ite')
        pause(1)

set(handles.ICG_STATUS,'String','OPENED','Backgroundcolor',Z)
        pause(1)

set(handles.ICG_STATUS,'String','OPENED','Backgroundcolor','wh
ite')
        pause(1)

set(handles.ICG_STATUS,'String','OPENED','Backgroundcolor',Z)
        pause(1)

set(handles.ICG_STATUS,'String','OPENED','Backgroundcolor','wh
ite')
        pause(1)

set(handles.ICG_STATUS,'String','OPENED','Backgroundcolor',Z)


% function for flashing the intake gate status when closed
function Close_Status_Flashing(handles)
        Z=[0.302 0.745 0.933];

set(handles.ICG_STATUS,'String','CLOSED','Backgroundcolor',Z)
        pause(1)

set(handles.ICG_STATUS,'String','CLOSED','Backgroundcolor','wh
ite')
```

```
        pause(1)

set(handles.ICG_STATUS,'String','CLOSED','Backgroundcolor',Z)
        pause(1)

set(handles.ICG_STATUS,'String','CLOSED','Backgroundcolor','wh
ite')
        pause(1)

set(handles.ICG_STATUS,'String','CLOSED','Backgroundcolor',Z)
        pause(1)

set(handles.ICG_STATUS,'String','CLOSED','Backgroundcolor','wh
ite')
        pause(1)

set(handles.ICG_STATUS,'String','CLOSED','Backgroundcolor',Z)


% --- Executes on button press in CLOSE_HMI.
function CLOSE_HMI_Callback(hObject, eventdata, handles)
% hObject    handle to CLOSE_HMI (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
    close(handles.ICG_HMI);

% --- Executes on button press in ICG_Raise.
function ICG_Raise_Callback(hObject, eventdata, handles)
% hObject    handle to ICG_Raise (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
```

```
%     RESET_VALS(handles)
    evalin('base','clear all')
    run('Clean_CMD.m')
% RESET_VALS(handles)
%     SolverConfg_msgbox(handles)
    model_open(handles)

    evalin('base','ICG_Raising')
    set_param('ICG_SIM_MODEL/Motor1','Value','1450')  %start
motor 1
    set_param('ICG_SIM_MODEL/Motor2','Value','0')     %do not
start motor 2
%set_param('ICG_SIM_MODEL/NL SPEED SP','Value','7')
%set_param('ICG_SIM_MODEL/ELGAIN','Gain','0.001')

set_param('ICG_SIM_MODEL','Solver','ode15s','StopTime','1650')
    set_param('ICG_SIM_MODEL', 'SimulationCommand', 'start')
    set(handles.OPER_CODE,'Value',1)

    pause(10)

% get the raising time in terms of seconds and minutes of the
intake gate
    RAIS_POS=evalin('base','RAIS_POS');
    SIM_TIME=evalin('base','SIM_TIME');
    L = length(RAIS_POS);
    V=RAIS_POS(L);
if RAIS_POS(L)>7
        r=1;
while RAIS_POS(r) < 7.25;
            r=r+1;
end
        N=(r-1);
        M=SIM_TIME(N);
        G=M/60;
```

```
        Z='green';
else
        M=0;
        G=0;
        Z='white';
end

set(handles.RAISING_TIME_SEC,'String',M,'Backgroundcolor',Z)
    set(handles.TIME_ELAPSED_R,'String',G,'Backgroundcolor',Z)

% get the flow measurment on gate raising
    RAISFLOW=evalin('base','RAISFLOW');
    SIM_TIME=evalin('base','SIM_TIME');
    L = length(RAISFLOW);
if RAIS_POS(L)>7
        r=1;
while RAIS_POS(r) < 7.25;
            r=r+1;
end
        N=(r-1);
        M=RAISFLOW(N);
        Z='green';
else
        M=0;
        Z='white';
end
    set(handles.RAIS_FLOW,'String',M,'Backgroundcolor',Z)

% get the pressure measurement on gate raising
    PRESSURE=evalin('base','PRESSURE');
    SIM_TIME=evalin('base','SIM_TIME');
    L = length(PRESSURE);
if RAIS_POS(L)>7
        r=1;
while RAIS_POS(r) < 7.25;
```

```
            r=r+1;
end
        N=(r-1);
        M=PRESSURE(N);
        Z='green';
else
        M=0;
        Z='white';
end
    set(handles.RAIS_PRESS,'String',M,'Backgroundcolor',Z)

% get the position and status of the intake gate
    RAIS_POS=evalin('base','RAIS_POS');
    SIM_TIME=evalin('base','SIM_TIME');
    L = length(RAIS_POS);
    V=RAIS_POS(L);
if V>1
        R=V;
        Z='green';
        Open_Status_Flashing(handles)
else
        R=round(V);
        Z='white';

set(handles.ICG_STATUS,'String','CLOSED','Backgroundcolor',Z)
end
    set(handles.ICG_POS,'String',R,'Backgroundcolor',Z)



% --- Executes on button press in ICG_RAIS_2PPs.
function ICG_RAIS_2PPs_Callback(hObject, eventdata, handles)
% hObject    handle to ICG_RAIS_2PPs (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
```

```
% handles    structure with handles and user data (see
GUIDATA)
%     RESET_VALS(handles)
    evalin('base','clear all')
    run('Clean_CMD.m')
% RESET_VALS(handles)
%     SolverConfg_msgbox(handles)
    model_open(handles)
    evalin('base','ICG_Raising')
    set_param('ICG_SIM_MODEL/Motor1','Value','1450')  %start
motor 1
    set_param('ICG_SIM_MODEL/Motor2','Value','1450')  %start
motor 2
%set_param('ICG_SIM_MODEL/NL SPEED SP','Value','7')
%set_param('ICG_SIM_MODEL/ELGAIN','Gain','0.001')

set_param('ICG_SIM_MODEL','Solver','ode15s','StopTime','1200')
    set_param('ICG_SIM_MODEL', 'SimulationCommand', 'start')
    set(handles.OPER_CODE,'Value',2)

    pause(10)

% get the raising time in terms of seconds and minutes of the
intake gate
    RAIS_POS=evalin('base','RAIS_POS');
    SIM_TIME=evalin('base','SIM_TIME');
    L = length(RAIS_POS);
    V=RAIS_POS(L);
if RAIS_POS(L)>7
        r=1;
while RAIS_POS(r) < 7.25;
            r=r+1;
end
        N=(r-1);
        M=SIM_TIME(N);
```

```
        G=M/60;
        Z='green';
else
        M=0;
        G=0;
        Z='white';
end

set(handles.RAISING_TIME_SEC,'String',M,'Backgroundcolor',Z)
    set(handles.TIME_ELAPSED_R,'String',G,'Backgroundcolor',Z)

% get the flow measurment on gate raising
    RAISFLOW=evalin('base','RAISFLOW');
    SIM_TIME=evalin('base','SIM_TIME');
    L = length(RAISFLOW);
if RAIS_POS(L)>7
        r=1;
while RAIS_POS(r) < 7.25;
            r=r+1;
end
        N=(r-1);
        M=RAISFLOW(N);
        Z='green';
else
        M=0;
        Z='white';
end
    set(handles.RAIS_FLOW,'String',M,'Backgroundcolor',Z)

% get the pressure measurement on gate raising
    PRESSURE=evalin('base','PRESSURE');
    SIM_TIME=evalin('base','SIM_TIME');
    L = length(PRESSURE);
if RAIS_POS(L)>7
        r=1;
```

```
while RAIS_POS(r) < 7.25;
            r=r+1;
end
        N=(r-1);
        M=PRESSURE(N);
        Z='green';
else
        M=0;
        Z='white';
end
    set(handles.RAIS_PRESS,'String',M,'Backgroundcolor',Z)

% get the position and status of the intake gate
    RAIS_POS=evalin('base','RAIS_POS');
    SIM_TIME=evalin('base','SIM_TIME');
    L = length(RAIS_POS);
    V=RAIS_POS(L);
if V>1
        R=V;
        Z='green';
        Open_Status_Flashing(handles)
else
        R=round(V);
        Z='white';

set(handles.ICG_STATUS,'String','CLOSED','Backgroundcolor',Z)
end
    set(handles.ICG_POS,'String',R,'Backgroundcolor',Z)

% --- Executes on button press in ICG_Lower.
function ICG_Lower_Callback(hObject, eventdata, handles)
% hObject    handle to ICG_Lower (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
```

```
% handles    structure with handles and user data (see
GUIDATA)
%     RESET_VALS(handles)
    evalin('base','clear all')
    run('Clean_CMD.m')
% RESET_VALS(handles)
%     SolverConfg_msgbox(handles)
    model_open(handles)
    evalin('base','ICG_Lowering')
%set_param('ICG_SIM_MODEL/ELGAIN','Gain','0.001')
    set_param('ICG_SIM_MODEL/Motor1','Value','0')     %do not
start motor 1
    set_param('ICG_SIM_MODEL/Motor2','Value','0')     %do not
start motor 2

set_param('ICG_SIM_MODEL','Solver','ode15s','StopTime','3000')
    set_param('ICG_SIM_MODEL', 'SimulationCommand', 'start')
    set(handles.OPER_CODE,'Value',3)

    pause(5)

% get the normal lowering time in terms of seconds and minutes
of the intake gate
    SIM_TIME=evalin('base','SIM_TIME');
    NLOWER_POS=evalin('base','NLOWER_POS');
    L = length(NLOWER_POS);
if NLOWER_POS(L)<=0
        G=0;
        M=0;
        Z='white';
else
       r=1;
while NLOWER_POS(r) > 0.05;
            r=r+1;
end
```

```
        r=r+3
while NLOWER_POS(r) <= 0.05;
            r=r+1;
end
        N=(r-1);
        M=SIM_TIME(N);
        G=M/60;
        Z=[0.302 0.745 0.933];
end
  set(handles.NLOWER_TIME_SEC,'String',M,'Backgroundcolor',Z)
  set(handles.TIME_ELAPSED_NL,'String',G,'Backgroundcolor',Z)
% get the flow measurement on gate normal lowering
    NORMLLWRFLOW=evalin('base','NORMLLWRFLOW');
    L = length(NLOWER_POS);
    SIM_TIME=evalin('base','SIM_TIME');

if NLOWER_POS(L)<=0
         M=0;
         Z='white';
else
        r=1;
while NLOWER_POS(r) > 0.001;
            r=r+1;
end
        N=(r-1);
        M=NORMLLWRFLOW(N);
        Z=[0.302 0.745 0.933];
end
  set(handles.NLOWER_FLOW,'String',M,'Backgroundcolor',Z)

% get the presssure measurement on gate normal lowering
    PRESSURE=evalin('base','PRESSURE');
    L = length(NLOWER_POS);
    SIM_TIME=evalin('base','SIM_TIME');
```

```
if NLOWER_POS(L)<=0
        M=0;
        Z='white';
else
       r=1;
while NLOWER_POS(r) > 0.001;
           r=r+1;
end
       N=(r-1);
       M=PRESSURE(N);
       Z=[0.302 0.745 0.933];
end
  set(handles.NLOWER_PRESS,'String',M,'Backgroundcolor',Z)

% get the position and status of the intake gate
    RAIS_POS=evalin('base','RAIS_POS');
    SIM_TIME=evalin('base','SIM_TIME');
    L = length(RAIS_POS);
    V=RAIS_POS(L);
if V>1
        R=V;
        Z='white';

set(handles.ICG_STATUS,'String','OPENED','Backgroundcolor',Z)
else
        R=round(V);
        Z=[0.302 0.745 0.933];
        Close_Status_Flashing(handles)
end
    set(handles.ICG_POS,'String',R,'Backgroundcolor',Z)


% --- Executes on button press in ICG_EMERG_LOWER.
function ICG_EMERG_LOWER_Callback(hObject, eventdata, handles)
% hObject    handle to ICG_EMERG_LOWER (see GCBO)
```

```
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
%      RESET_VALS(handles)
    evalin('base','clear all')
    run('Clean_CMD.m')
% RESET_VALS(handles)
%      SolverConfg_msgbox(handles)
    model_open(handles)
    evalin('base','ICG_Emerg_Lowering')
%set_param('ICG_SIM_MODEL/NL SPEED SP','Value','10')
%set_param('ICG_SIM_MODEL/ELGAIN','Gain','0.05')
    set_param('ICG_SIM_MODEL/Motor1','Value','0')     %do not
start motor 1
    set_param('ICG_SIM_MODEL/Motor2','Value','0')     %do not
start motor 2

set_param('ICG_SIM_MODEL','Solver','ode15s','StopTime','500')
    set_param('ICG_SIM_MODEL', 'SimulationCommand', 'start')
    set(handles.OPER_CODE,'Value',4)

    pause(5)

% get the emergency lowering time in terms of seconds and
minutes of the intake gate
    ELOWER_POS=evalin('base','ELOWER_POS');
    L = length(ELOWER_POS);
    SIM_TIME=evalin('base','SIM_TIME');
if ELOWER_POS(L)<=0
        G=0;
        M=0;
        Z='white';
else
        r=1;
```

```
while ELOWER_POS(r) > 0.05;
            r=r+1;
end
       r=r+3
while ELOWER_POS(r) <= 0.05;
           r=r+1;
end
        N=(r-1);
        M=SIM_TIME(N);
        G=M/60;
        Z='red';
end

set(handles.ELOWER_TIME_SEC,'String',M,'Backgroundcolor',Z)

set(handles.TIME_ELAPSED_EL,'String',G,'Backgroundcolor',Z)

% get the flow measurement on gate emergency lowering
    EMERGLWRFLOW=evalin('base','EMERGLWRFLOW');
    L = length(ELOWER_POS);
    SIM_TIME=evalin('base','SIM_TIME');
if ELOWER_POS(L)<=0
        M=0;
        Z='white';
else
        r=1;
while ELOWER_POS(r) > 0.001;
            r=r+1;
end
        N=(r-1);
        M=EMERGLWRFLOW(N);
        Z='red';
end
    set(handles.EMLOWER_FLOW,'String',M,'Backgroundcolor',Z)
```

```
% get the pressure measurement on gate emergency lowering
    PRESSURE=evalin('base','PRESSURE');
    L = length(ELOWER_POS);
    SIM_TIME=evalin('base','SIM_TIME');
if ELOWER_POS(L)<=0
        M=0;
        Z='white';
else
        r=1;
while ELOWER_POS(r) > 0.001;
            r=r+1;
end
        N=(r-1);
        M=PRESSURE(N);
        Z='red';
end
    set(handles.EMLOWER_PRESS,'String',M,'Backgroundcolor',Z)



% get the position and status of the intake gate
    RAIS_POS=evalin('base','RAIS_POS');
    SIM_TIME=evalin('base','SIM_TIME');
    L = length(RAIS_POS);
    V=RAIS_POS(L);
if V>1
        R=V;
        Z='white';

set(handles.ICG_STATUS,'String','OPENED','Backgroundcolor',Z)
else
        R=round(V);
        Z='red';
        Close_Status_Flashing(handles)
end
```

```
    set(handles.ICG_POS,'String',R,'Backgroundcolor',Z)

% --- Executes on slider movement.
function LWRSPDSLIDER_Callback(hObject, eventdata, handles)
% hObject    handle to LWRSPDSLIDER (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to
determine range of slider
    model_open(handles) % Ensure model is open

% Get the new value for the NLGAIN Gain from the slider
    NewVal = get(hObject,'Value');

% Set the value of the NLGAINCURRVAL to the new value set by
slider
    set(handles.NLGAINCURRVAL,'String',NewVal)

% Set the gain parameter of the NLGAIN gain Block to the
% new value
    set_param('ICG_SIM_MODEL/NL SPEED
SP','Value',num2str(NewVal))
%endfunction LWRSPDSLIDER_Callback



function NLGAINCURRVAL_Callback(hObject, eventdata, handles)
% hObject    handle to NLGAINCURRVAL (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
```

```
% Hints: get(hObject,'String') returns contents of
NLGAINCURRVAL as text
%        str2double(get(hObject,'String')) returns contents of
NLGAINCURRVAL as a double
    model_open(handles) % Ensure model is open

% Get the new value for the Ki Gain
    NewStrVal = get(hObject,'String');
    NewVal = str2double(NewStrVal);

% Check that the entered value falls within the allowable
range
if  isempty(NewVal) || (NewVal< 10) || (NewVal>0)
% Revert to last value, as indicated by LWRSPDSLIDER
        OldVal = get(handles.LWRSPDSLIDER,'Value');
        set(hObject,'String',OldVal)

else% Use new NLGAIN value
% Set the value of the LWRSPDSLIDER to the new value
        set(handles.LWRSPDSLIDER,'Value',NewVal)

% Set the Numerator parameter of the NLGAIN Gain Block to the
% new value
        set_param('ICG_SIM_MODEL/NL SPEED
SP','Value',NewStrVal)
end
%endfunction NLGAINCURRVAL_Callback


% --- Executes on slider movement.
function LOADSLIDER_Callback(hObject, eventdata, handles)
% hObject    handle to LOADSLIDER (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
```

```
% handles    structure with handles and user data (see
GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to
determine range of slider

    model_open(handles) % Ensure model is open

% Get the new value for the GATE_WIEGHT  from the slider
    NewVal = get(hObject,'Value');

% Set the value of the LOADCURRVAL to the new value set by
slider
    set(handles.LOADCURRVAL,'String',NewVal)

% Set the Constant parameter of the GATE_WIEGHT constant Block
to the new value

set_param('ICG_SIM_MODEL/GATE_WEIGHT','Value',num2str(NewVal))



function LOADCURRVAL_Callback(hObject, eventdata, handles)
% hObject    handle to LOADCURRVAL (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% Hints: get(hObject,'String') returns contents of LOADCURRVAL
as text
%        str2double(get(hObject,'String')) returns contents of
LOADCURRVAL as a double
model_open(handles) % Ensure model is open
```

```
% Get the new value for the Ki Gain
    NewStrVal = get(hObject,'String');
    NewVal = str2double(NewStrVal);

% Check that the entered value falls within the allowable
range
if  isempty(NewVal) || (NewVal< 3) || (NewVal>0)
% Revert to last value, as indicated by LWRSPDSLIDER
        OldVal = get(handles.LOADSLIDER,'Value');
        set(hObject,'String',OldVal)

else% Use new NLGAIN value
% Set the value of the LWRSPDSLIDER to the new value
        set(handles.LOADSLIDER,'Value',NewVal)

% Set the Numerator parameter of the NLGAIN Gain Block to the
% new value
    set_param('ICG_SIM_MODEL/GATE_WEIGHT','Value',NewStrVal)
end

% --- Executes on selection change in POPUP_MENU.
function POPUP_MENU_Callback(hObject, eventdata, handles)
% hObject    handle to POPUP_MENU (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns
POPUP_MENU contents as cell array
%        contents{get(hObject,'Value')} returns selected item
from POPUP_MENU

function POSITION_PLOTPROPERTIES(handles)
```

```
%                 title('Position','FontSize',14,...
%                 'FontWeight','bold');
%                 xlabel('Time (seconds)','FontSize',12,...
%                 'FontWeight','bold');
                ylabel('Position (m)','FontSize',10,...
'FontWeight','bold');
                ylim([0 max(ylim)])
function PRESSURE_PLOTPROPERTIES(handles)
%                 title('Oil Pressure','FontSize',14,...
%                 'FontWeight','bold');
%                 xlabel('Time (seconds)','FontSize',12,...
%                 'FontWeight','bold');
                ylabel('Pressure (bar)','FontSize',10,...
'FontWeight','bold');
                ylim([0 max(ylim)])
function FLOW_PLOTPROPERTIES_RAISE(handles)
%                 title('Oil Flow Rate','FontSize',14,...
%                 'FontWeight','bold');
                xlabel('Time (seconds)','FontSize',10,...
'FontWeight','bold');
                ylabel('Flow Rate (l/min)','FontSize',10,...
'FontWeight','bold');
function FLOW_PLOTPROPERTIES_LOWER(handles)
%                 title('Oil Flow Rate','FontSize',14,...
%                 'FontWeight','bold');
                xlabel('Time (seconds)','FontSize',10,...
'FontWeight','bold');
                ylabel('Flow Rate (l/min)','FontSize',10,...
'FontWeight','bold');
                 ylim([0 max(ylim)])


% --- Executes on button press in PLOT_RESULT.
function PLOT_RESULT_Callback(hObject, eventdata, handles)
% hObject    handle to PLOT_RESULT (see GCBO)
```

```
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
        R = get(handles.OPER_CODE,'Value');
        V = get(handles.POPUP_MENU,'Value');

if R==1 %Operation code for Raise with one pump
             SIM_TIME=evalin('base','SIM_TIME');
             RAIS_POS=evalin('base','RAIS_POS');
             PRESSURE=evalin('base','PRESSURE');
             RAISFLOW=evalin('base','RAISFLOW');
             figure(1)
if V==1
                subplot(3,1,1)
                plot(SIM_TIME,RAIS_POS,'g','linewidth',2);grid
                POSITION_PLOTPROPERTIES(handles)
elseif V==2
                subplot(3,1,2)
                plot(SIM_TIME,PRESSURE,'g','linewidth',2);grid
                PRESSURE_PLOTPROPERTIES(handles)
elseif V==3
                subplot(3,1,3)
                plot(SIM_TIME,RAISFLOW,'g','linewidth',2);grid
                FLOW_PLOTPROPERTIES_RAISE(handles)
end
elseif R==2 %Operation code for Raise with two pump
             SIM_TIME=evalin('base','SIM_TIME');
             RAIS_POS=evalin('base','RAIS_POS');
             PRESSURE=evalin('base','PRESSURE');
             RAISFLOW=evalin('base','RAISFLOW');
             figure(2)
if V==1
                subplot(3,1,1)
                plot(SIM_TIME,RAIS_POS,'g','linewidth',2);grid
```

```
                POSITION_PLOTPROPERTIES(handles)
elseif V==2
                subplot(3,1,2)
                plot(SIM_TIME,PRESSURE,'g','linewidth',2);grid
                PRESSURE_PLOTPROPERTIES(handles)
elseif V==3
                subplot(3,1,3)
                plot(SIM_TIME,RAISFLOW,'g','linewidth',2);grid
                FLOW_PLOTPROPERTIES_RAISE(handles)
end
elseif R==3 %Operation code for Normal Lowering
             SIM_TIME=evalin('base','SIM_TIME');
             NLOWER_POS=evalin('base','NLOWER_POS');
             PRESSURE=evalin('base','PRESSURE');
             NORMLLWRFLOW=evalin('base','NORMLLWRFLOW');
             figure(3)
if V==1
                subplot(3,1,1)

plot(SIM_TIME,NLOWER_POS,'b','linewidth',2);grid
                POSITION_PLOTPROPERTIES(handles)
elseif V==2
                subplot(3,1,2)
                plot(SIM_TIME,PRESSURE,'b','linewidth',2);grid
                PRESSURE_PLOTPROPERTIES(handles)
elseif V==3
                subplot(3,1,3)

plot(SIM_TIME,NORMLLWRFLOW,'b','linewidth',2);grid
                FLOW_PLOTPROPERTIES_LOWER(handles)
end
elseif R==4 %Operation code for Emergency Lowering
             SIM_TIME=evalin('base','SIM_TIME');
             ELOWER_POS=evalin('base','ELOWER_POS');
             PRESSURE=evalin('base','PRESSURE');
```

```
EMERGLWRFLOW=evalin('base','EMERGLWRFLOW');
figure(4)
if V==1
    subplot(3,1,1)

plot(SIM_TIME,ELOWER_POS,'r','linewidth',2);grid
    POSITION_PLOTPROPERTIES(handles)
elseif V==2
    subplot(3,1,2)
    plot(SIM_TIME,PRESSURE,'r','linewidth',2);grid
    PRESSURE_PLOTPROPERTIES(handles)
elseif V==3
    subplot(3,1,3)

plot(SIM_TIME,EMERGLWRFLOW,'r','linewidth',2);grid
    FLOW_PLOTPROPERTIES_LOWER(handles)
end

end
```