AN ADAPTIVE NEURAL NETWORKS CONTROLLER FOR THE INLINE SURFACE OIL/WATER SEPARATOR

BY

WESAM ABDULLAH AL-SHAMMASI

A Thesis Presented to the

DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

ELECTRICAL ENGINEERING

JUNE 2005

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis written by

WESAM A. AL-SHAMMASI

under the direction of his thesis advisor and approved by his thesis committee has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING.

Thesis Committee

Prof. Youssef L. Abdel-Magid (Chairman)

Dr. Ibrahim Babelli (Co Chairman)

Jamil. M. Bakhashwain
Dr. Jamil M. Bakhashwain
Department Chairman

M. Mohandes
Dr. Mohamed Mohandes (Member)

Dr. Mohammad A. Al-Ohali
Dean of Graduate Studies

Date 2-7-2005

Dedicated to

My Beloved Parents

and

My sincere Wife

ACKNOWLEDGEMENTS

In the name of Allah, the most Merciful, the most Gracious. All praise and glory go to Almighty Allah (Subhanahu Wa Ta'ala) who gave me the courage and patience to carry out this work.

I would like to acknowledge all those who helped me to complete this thesis. Acknowledgement is due to King Fahd University of Petroleum & Minerals and Schlumberger for providing the full support and facilities in carrying out this work.

My special thanks go to Dr. Youssef L. Abdel-Magid for his extreme patience. There were many times that I doubted that I would get this thesis completed, but he never let me think that was an option. His enthusiasm and guidelines is one of the main boosting elements for me to finish my MS thesis.

My deep appreciation and profound gratitude goes to Dr. Ibrahim Babli for his constant endeavor, guidance and invaluable instructions and suggestions to accomplish this work. His Stimulating Suggestions made this work interesting and valuable to me.

Many thanks go to my thesis committee members, especially, Dr. Mohammed Mohandes who give me the Artificial Neural Network Course where I had a very wonderful experience. Thanks to him for his continuous encouragement and support during the course of this work.

Special thanks go to my friend Salah Al-Jishi and to my brother in law Nashat Al-Marhoon for their encouragement with great deal of moral to finish this work.

During the course of this thesis, my brothers, sisters, and family members were very close to me by encouraging me towards success. Bahera, Beshra, Hussain, Bassam

deserve my warm thanks. With special, thank to my uncle Zaki Al-Khunaizi who kept following with me to finish this work.

I wish to record my very special sincere, appreciation and thanks to my father and mother who give me the warm encouragement with close follow up to continue my higher studies. Their prayers and moral support will be always in my heart.

My most sincere thanks to my beloved wife Lamia who gave me here endless supports, patience, emotional and encouragement to continue my study. My son Rayan, his happiness, and his marvelous smile gave me the greater encouragement to finish this work. The best part of the life is sharing it with the one you love. Thank you Lamia and Rayan.

Table of Contents

List of Tables

LIST OF FIGURES

THESIS ABSTRACT

Name Wesam Abdullah Al-Shammasi

Title AN ADAPTIVE NEURAL NETWORK CONTROLLER FOR THE IN-LINE SURFACE OIL/WATER SEPARATOR

Degree MASTER OF SCIENCE

Major Field ELECTRICAL ENGINEERING

Date of Degree

The objective of this thesis is to develop an adaptive Artificial Neural Networks (ANN) controller to control water extraction of the Inline Surface Oil/Water Separator (ISOWS). The thesis contains both a theoretical part and a practical part of the implementation of the ANN controller. The definition of the process problem is briefly discussed to explain the practical application of the ANN controller. An overview of the ANN types and the calculations commonly used for training Feed Forward ANNs are given. More emphasis is shown on the RBF Networks with online training. The practical part shows the detailed implementation blocks of this project consisting of the system used to control water extraction from the ISOWS, which includes water-in-oil detection probes, input processing, ANN controller, anti-windup routine, output processing, and the motor-operated-valves (MOVs), which are controlled using the output parameters. The results of the practical application of the ANN controller to the ISOWS are illustrated using screen captures of the graphical interface of the developed software and a short video for the actual operation and control of the ISOWS.

Keywords: *Control System, Process Control, Instrumentation, Petroleum, Neural Networks, Radial Basis Function Networks*

MASTER OF SCIENCE DEGREE

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS, DHAHRAN

JUN 2005

خلاصة الرسالة

الاسم وسام عبدالله الشماسي

عنوان الرسالة التحكم بجهاز فصل الماء عن الزيت في الأنابيب بواسطة الشبكات العصبية المتكيفة

الدرجة الماجستير في العلوم

التخصص الهندسة الكهربائية

تاريخ التخرج يونيو 2005 م

الهدف من هذه الرسالة هو تطوير نظام للتحكم مبني على الشبكات العصبية الصناعية المتكيفة لجهاز فصل الماء عن الزيت في الأنابيب، حيث ينقسم هذا العمل إلى جانب نظري و جانب عملي، وستتضمن ايضا التعريف بجهاز فصل الماء عن الزيت في الأنابيب وما يساندها من نظريات تصاحب هذه العملية، وبعدها تم التطرق إلى الجانب النظري لأنواع الشبكات العصبية الصناعية، مع التركيز على الشبكات العصبية للدوال النصف قطرية ذات التدريب الفوري، بينما تم طرح الشرح التفصيلي للجانب العملي عن طريق تقسيم المسألة إلى عدة أقسام: قسم كشف الماء من الزيت بواسطة المسابير، ثم قسم معالجة الدخل، ثم قسم التحكم بواسطة الشبكات العصبية مع الكشف عن الصمامات المتشبعة، ثم قسم معالجة الخرج، ثم قسم الصمامات المتحكم بها بواسطة محرك حيث يتم التحكم بكمية الماء المستخلص من الزيت، وقد تم دعم هذا العمل بالنتائج العملية بواسطة لقطات رسومية من شاشة الإدخال و لقطات تلفزيونية من عمل نظام التحكم في جهاز فصل الماء عن الزيت في الأنابيب.

درجة الماجستير في العلوم

جامعة الملك فهد للبترول و المعادن

يونيو 2005 م

Chapter 1

INTRODUCTION

1.1 Research Problem

Oil/Water separation is a common process in the oil industry. Normally, crude oil is driven out from the well by the natural underground reservoir pressure. This pressure, however, is not always enough to bring out all the crude oil from the well, and even when it is enough at the start of production, it declines with reservoir lifetime. There are several effective solutions to restore production pressure, of which pumping water into the production zone is the most common. Due to the density of water being greater than that of oil, water will drive out the oil from well. Although this has a big advantage, it causes additional water to flow with the production stream, which needs thereafter to be separated from oil. The basic objective of the oil/water separator is to separate oil from water and reuse this water or dispose of it as per the local needs and regulations.

Conventional oil/water separators are generally located far away from oil wells, and to which a multitude of wells are connected. The problems of this type of separator are the high cost of construction, operating and maintenance. It requires vessels, tanks, long distance pipes, pumps, etc. In addition to this, increased water production is limiting the production capacity of the existing pipelines. Any water separation at or near the wellhead will significantly relieve the existing facilities.

In order to overcome these difficulties, Schlumberger introduced a new separation concept. The design is based on the fact that oil and water tend to separate and stratify when flowing in a horizontal or near horizontal pipelines under the action of gravity, as long as the local velocity of each phase within the pipe is below a certain threshold, which depends on pressure, physical properties, and inclination, among other parameters. Oil and water local velocities in production pipelines are safely below this threshold.

If the mixture is pumped in a pipe inclined uphill, the local interface between water and oil will shift upward thus increasing the water holdup (Figure 2.1). This is caused by water slowing down under gravity with oil accelerating, while there is no change in the individual flowrate of each phase. The location of the interface is governed mainly by the oil flowrate and pipe inclination. This is explained in details in the next Chapter.

As shown in the figure, water flows in the lower section of the pipe and oil flows in the upper section. To extract water from the mixture, an extraction port (or ports) is located at or just downstream of the lowest, either absolute or local, point in the pipelines where water accumulates and forms a sump. The water sump will extend when oil is flowing on top of it thus forming a tail of water that ends somewhere downstream in the pipe, beyond which there will only be crude oil, which will continue to flow in the pipe.

This method has many advantages: It will reduce the cost of pumping water to the conventional separator units; hence the reliability of these units will increase (load

reducing), de-bottlenecking of the oil production facilities due to the fact that the amount of water sent to conventional separator will be reduced and will be replaced by crude oil.

The objective of this work is to develop a controller to control the axial tail of water in the uphill section by extracting the water from the up steam of the uphill section of the Inline Surface Oil/Water Separator (ISOWS).

1.2 Motivation of the Work

In the next chapter the discussion will focus on the physical phenomena and the experimental installation of the new water/oil separator. It will be shown that the system is highly nonlinear, since there are many disturbances, which are difficult to estimate using conventional calculations. This is due to the effect of sudden variations in oil production from the field, which may be manifested by frequent changes in oil flowrates, water flowrates, and water-liquid-ratios (WLRs) form different wells in the field, or from different fields.

Since the fundamental parameters, i.e., flow rates and flow ratios, of the system keep on changing, it is required to use a controller that has the ability to change its behavior, i.e., adapt, in order to conform to the new circumstances. The basic function of an adaptive controller is its ability to change its response based on the changes of dynamics of the process and reduce the effect of the disturbances. The new controller shall be an adaptive type that has "adjustable parameters and a mechanism for adjusting

the parameters". In this case the controller will be nonlinear since it has an adjustable mechanism for the parameters. [1]

According to the above description, this thesis proposes a new technique for water extraction control that uses intelligent control scheme. The proposed technique is to use Artificial Neural Network (ANN) as a controller.

A question may naturally arises: Why neural networks? And how would ANN be used to control the system in question? ANN applications are widely used in engineering applications. For example, they are used in function approximations, pattern recognition, system classification, communications, signal and image processing, power systems, and adaptive controllers. Neural Network's potential in learning linear and nonlinear systems is due to its non-linear and multi-stage structure [2] [3].

Because of the powerful ability of the ANN to identify almost any nonlinear function, ANN will be used in this thesis to identify the inverse of the process plant. By doing this, the process plant will try to follow the desired input.

1.3 Thesis Objective

In this thesis, the inverse of the plant will be identified using intelligent technique (ANN) to control the level of water within the inline water/oil separator system. In order to fulfill this objective, the following activities are to be carried out:

1. Develop analog reading software. The purpose of this software is to convert the readings of the localized water detection probes, within the instrumented section of the inline separator, into water level measurements inside a pipe.
2. Develop Radial Basis Function Network as an ANN controller.
3. Develop Anti-windup software for the ANN controller. The reason for adopting this technique is because of using saturated control valves.

1.4 Thesis Organization

This thesis is organized as follows. Chapter I is an introduction. Chapter II will review the physical concepts and design of the inline water/oil separator. In Chapter III, an overview of the history and background of Artificial Neural Networks ANN, famous techniques like the Back-Propagation (BP) and Radial Basis Function Network (RBFN) will be given. Chapter IV will provide the technique used to control water extraction in the inline water/oil separator based on online inverse plant identification and using ANN with RBFN technique. It will demonstrate the way used to develop the Analog Input reading, ANN Controller, Control Valve and Controller Anti-windup. Finally, Chapter V will show the result the test and the requirement for the controller.

Chapter 2

Water/ Oil Separation Overview

2.1 Conventional Water/ Oil Separation – an overview

The separation of water from oil in the oil production stream is a very common process in the oil industry. Water is naturally found in the reservoirs beneath oil and is also added to the reservoir, when the reservoir pressure drops, in order to compensate for this pressure drop, such that oil would reach the surface without pumping or lifting. The water/oil mixture is then transported via pipelines from various wells to a common manifold, and then to a central water/oil separation facility. Water separation is carried out using bulky separation tanks where water separates from oil under gravity and settles at the bottom of the tank whereas oil, having lower density remains at the top of the tank. Water is then extracted and is either pumped back to the reservoirs for re-injection or disposed of, while separated oil is pumped to the storage and/or refinery areas. This method of central separation has the following disadvantages:

1. The central separation facilities include bulky and heavy tanks, which are expensive, in addition to space and weight limitations for offshore platforms.

2. Having only central separation facilities dictates transporting water and oil in the production stream for hundreds of kilometers, and then the separated water needs to be pumped further hundreds of kilometers for re-injection and/or disposal. This reduces significantly the pipeline oil transportation capacities.
3. The increasing water content in the production stream will negatively affect the life cycle of the central water handling facilities.
4. With increasing water cut in the production stream, the cost for building central separation facilities is spiraling, both in terms of separation facilities and pipelines.

2.2 New concept of Water/ Oil Separator

The disadvantages of using central water separation facilities discussed above have necessitated a fresh look at water separation techniques and approaches. A new concept of inline surface water separation at/or near the wellhead was proposed and developed at Schlumberger Dhahran Carbonate Research (SDCR) center in Dhahran, Saudi Arabia. The fundamental operational principles of this new technique are similar, in principle, to the conventional method of separation in that the basic principle is based on gravitational separation of water and oil. In contrast to central separation, however, gravity is taken advantage of in the new technique while both phases are flowing concurrently in a near-horizontal pipeline that is long enough for the two phases, i.e., oil and water, to stratify (separate) under the action of gravity. The pipe size where separation takes place is of the

same size of the pipeline, which renders this approach quite attractive in terms of location and cost. [4]

The phenomena and working principles will be discussed briefly in the coming sections.

2.3 System operating phenomena

Certain physical phenomena govern any mixture flow in a horizontal or near-horizontal pipeline. Understanding these phenomena can explain the principle of operation of the new inline water/oil separator. [4]

The first phenomenon is shown in Figure 2.1 below. If there exists a pipeline that is inclined downhill and then uphill, it is observed that water accumulates at the lowest point forming a sump in the pipeline. The values of oil flowrates inside the pipes, common to oil pipelines, does not cause water entrainment, which means that the water sump would not be depleted, and the water layer that is formed downstream of the sump remains intact as long as the oil flow velocity in the pipeline does not exceed certain limits. [4]

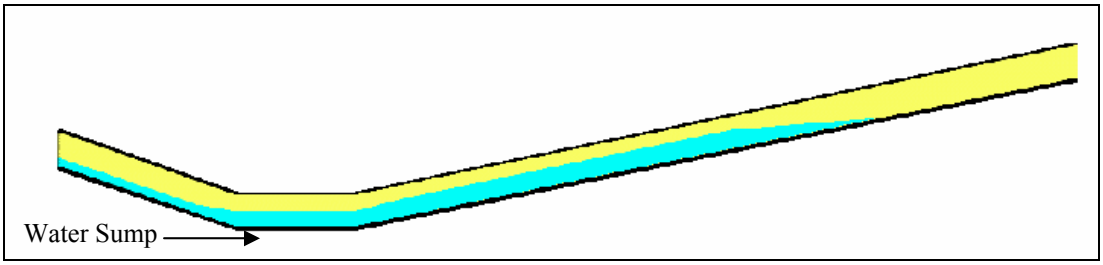


Figure 2.1: Water Sump formation at local low point

The second phenomenon is shown in Figure 2.2. This phenomenon can be explained as follows: If two liquid phases are flowing inside a pipeline that is inclined uphill, the heavier liquid, i.e., water, will decelerate under gravity action while the lighter phase, i.e., oil, would accelerate. The flowrate of each phase that is entering the inclined section from the horizontal section does not change, because there is no discharge and/or intake. Therefore, by using mass conservation arguments, water, which has slowed down, occupies more area of the pipe cross-section, while oil, which has accelerated occupies less area of the pipe cross-section. This results in increase in the water holdup without actually having any increase in the water cut. [4]

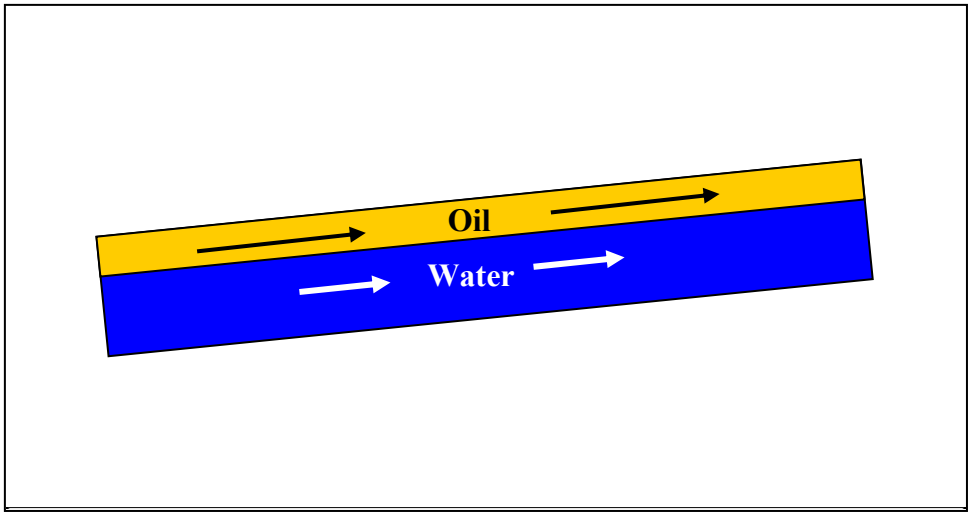


Figure 2.2: Water/ Oil Layer formation in rising inclined pipe

Figure 2.3 shows the holdup increasing as the pipeline inclination is increased from 0° to 1° to 10°. The water cut in this example is approximately 50% and remains constant, while the holdup is significantly increasing. This is only an illustrative sketch, since other factors contribute to holdup change other than inclination, such as physical properties and interfacial friction between the two phases. [4]

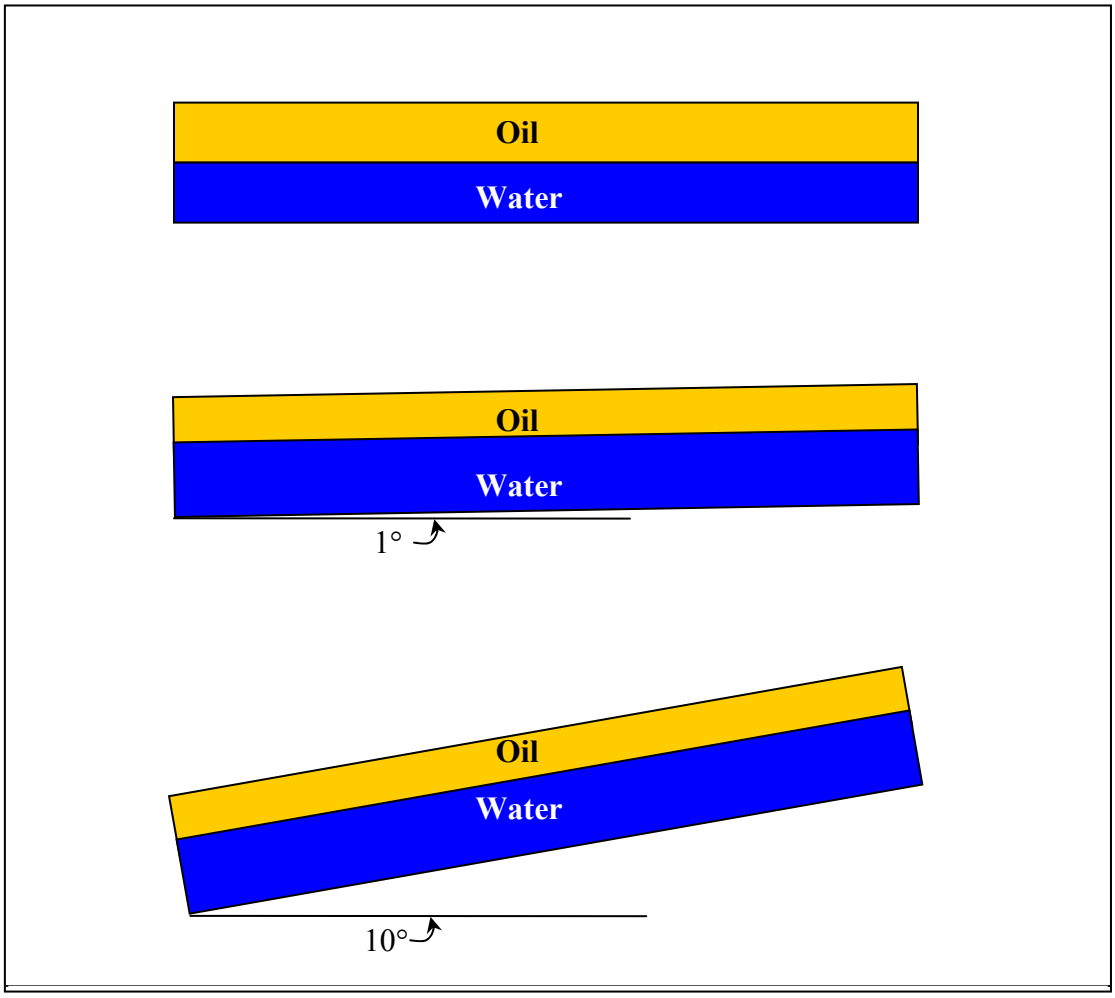


Figure 2.3: Water interface hold-up increasing as the slope of pipe increases

It is observed for the third phenomenon (Figure 2.4) that the holdup of the slower phase in a pipe inclined uphill is independent of its own flowrate. Unexpectedly, it is dependent on the flowrate of the faster phase and the inclination. This phenomenon is the key point for this inline water/oil separator. Based on this phenomenon, the water holdup is independent of the water flowrate, but is rather dependent on oil flowrate and pipe inclination. The other aspect of this phenomenon is that if we are to extract water from any extraction port in the inclined section of the pipe where there is water, the water

holdup will not change, which would help in avoiding oil coning into the extraction port. [4]

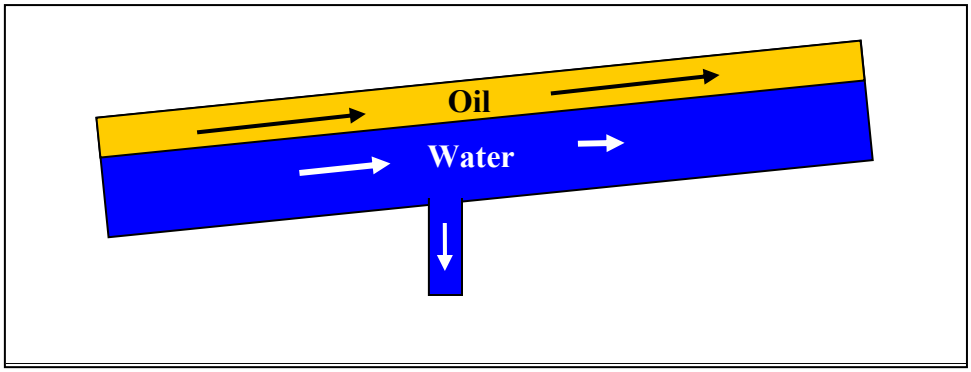


Figure 2.4: Water extraction does not affect the hold-up in inclined pipe

If water extraction does not affect the thickness of the water layer (or the holdup) in the inclined pipe, then the question is: Where is water extraction manifested? Since the water holdup does not change with water flow rate change (both extraction and/or addition), then the only possible explanation is that the axial extent of the water sump tail must change with changing water flow rate. Therefore, if the water flow rate increases, the axial extent of the water layer will advance, whereas if the water flow rate decreases, the axial extent of the water layer will retreat. Figure 2.5: (a), and (b) illustrate the effect of varying water flow rate on the axial extent of the water layer front. [4]

Thereafter, if a water extraction port is installed near the lower point of the pipeline, it will be possible to extract the net incoming water out of the system. If this case is achieved, the water layer will be dynamically stable, i.e., the water layer front will be within controlled limits. On the other hand, if the water influx is more than the extraction rate, the water will advance. And, if the extracted water flow rate is more than the water influx, the water layer will retreat. [4]

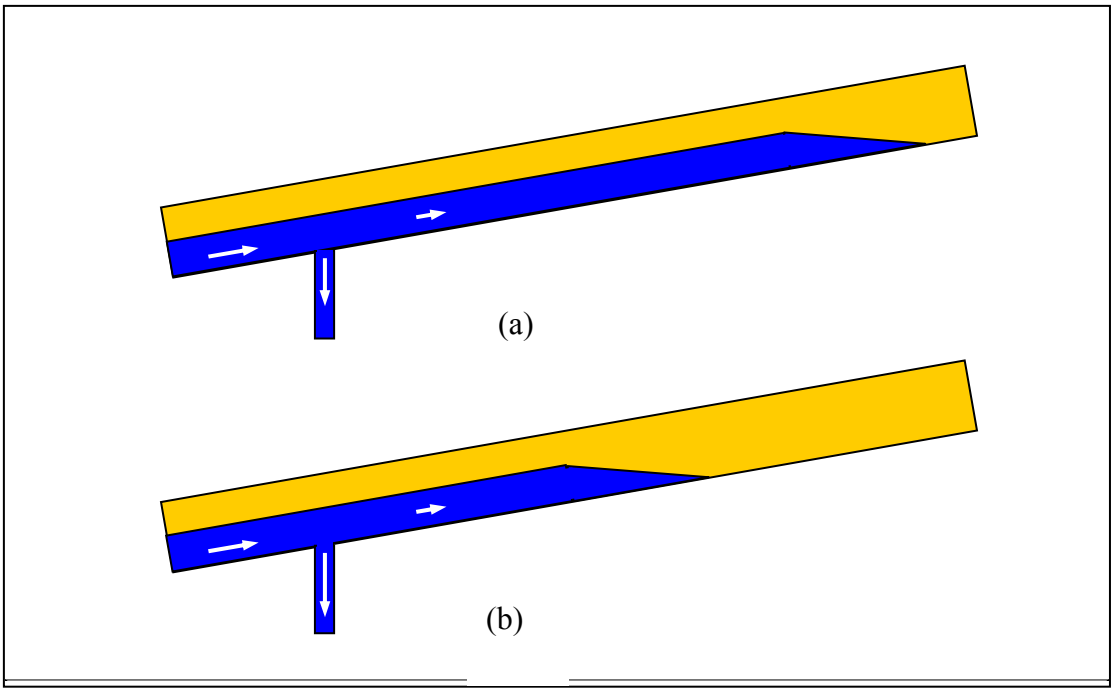


Figure 2.5: Effect of water extracted from pipe. a) under extracting makes the axial movement advanced (b) Over-extracting makes the axial movement retreats.

We may form an analogy between the new inline water/oil separator and a tank having inflow and outflow of water. A more realistic analogy would have the tank boundary changing as the oil flow changes. For simplicity, we will designate the water layer front axial movement in the pipe as the water level. [4]

2.4 Experiment Setup

In order to develop the water extraction control system of the inline water/oil separator, a pilot plant is developed in order to test the phenomena and to develop the control system. In this section, a discussion is given of the mechanical set-up and the electronic set-up of the system. [4]

2.4.1 Mechanical Setup

The experimental set-up is shown in Figure 2.6. It is made up of liquid delivery and recovery systems, which include an oil tank, water tanks and a two-stage separator water/oil gravity separator. Both the oil and the water tanks are connected with a pump at their discharge. The discharge of the oil pump is connected to a filter. The oil line and water line are joined at a T-joint. This causes sufficient mixing in order to simulate the actual mixture of oil produced from the oil fields either exiting the choke at the wellhead of the manifold. [4]

The pipeline after the T-joint is inclined downhill followed by a horizontal section, which in turn is followed by the main extraction pipe section, which is inclined uphill. The three sections (downhill, horizontal and uphill) are installed in order to simulate the phenomena described in the previous section. Following the extraction section of the pipe is the instrumented section, where the presence and axial extent of the water in the oil are detected. The outlet of this last section is routed towards the two-stage gravity separation tank. [4]

The extraction ports drain the extracted water under the action of gravity into the main water storage tanks, via motor-operated-valves (MOVs). The water in these tanks is fed to the main water tank using a level-controlled pump, which would fill the main water tank whenever the water level in this tank drops below a certain predefined level. [4]

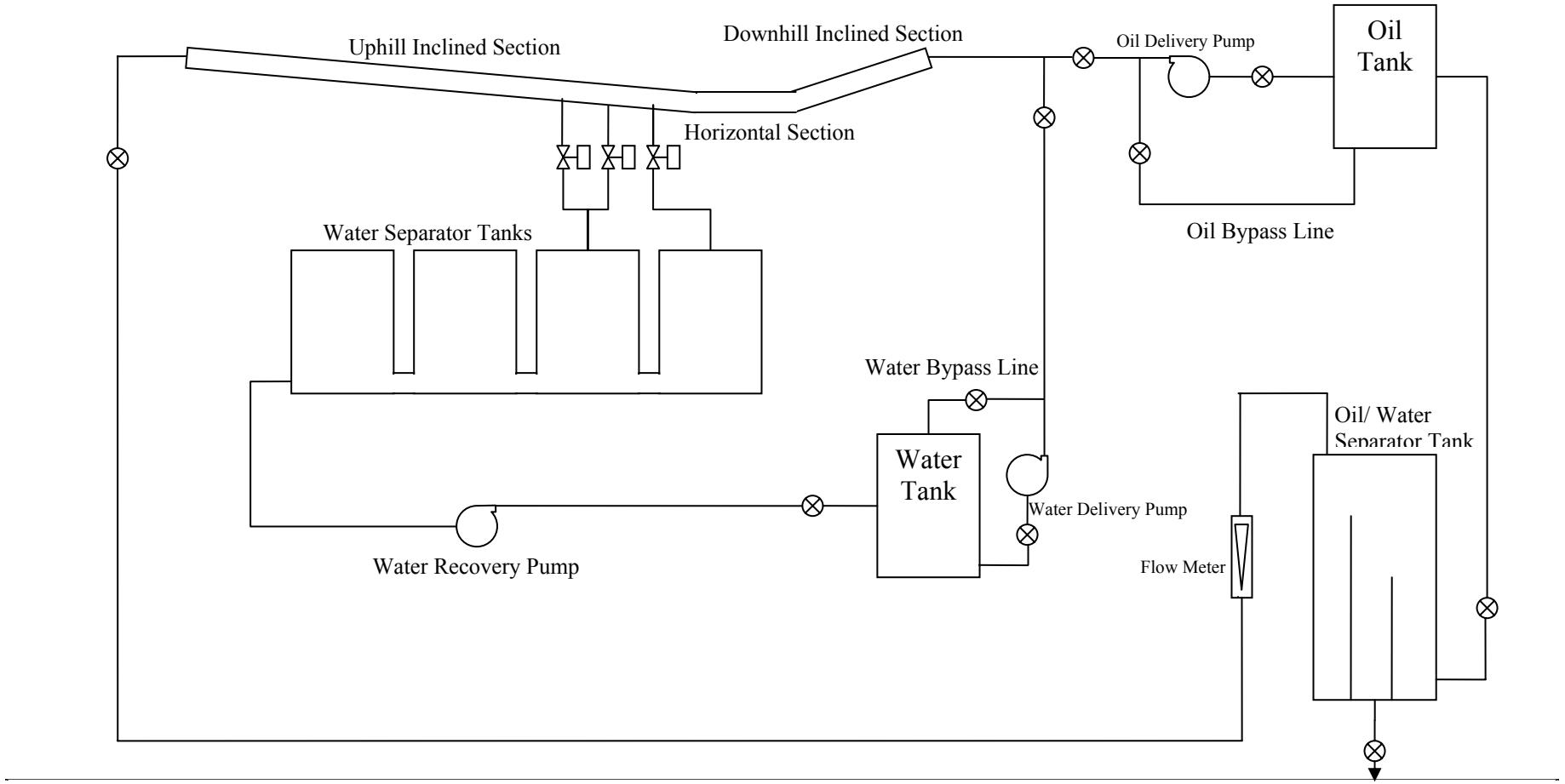


Figure 2.6: Mechanical Set-Up for the In-line water/Oil separator

2.4.2 Instrumental Setup

Now the challenging part is how to detect the axial movement of the water layer front, and how to use this information to control the water extraction rate such that the separator will be dynamically stable. We will describe here the principle of measuring the axial water movement, while the other part will be explained later in Chapter 4.

The loop is equipped with conductivity-based probes to detect the axial water layer movement by detecting the presence of water-in-oil. It is basically composed of a pair of electrodes that penetrate the pipe wall and are immersed into the liquid stream. An AC (or DC) signal is sent in parallel through these electrodes, and the voltage is measured across the pair of electrodes. Then, if water passes over them (they are placed near the bottom of the pipe), the electrical conductivity increases (resistance decreases). On the other hand, if oil covers these electrodes, the conductivity decreases (resistance increases) and the reading will be of an open circuit. This principle is used to differentiate between the presences of water from oil.

In order to build a meter to detect the axial water layer movement, a series of electrodes are connected along the pipe as shown in Figure 2.7. The water layer advances or retreats according to the difference between water influx and water extraction rates. The electrodes are connected to a Data Acquisition System (the details of the connection are provided in Chapter 4, Figure 4.2). The signals are then routed to the control system where they will be used in the control decisions. The output from the control system is

then processed through the Data Acquisition System and a signal is sent towards the motor-operated-valves that control the extraction ports.

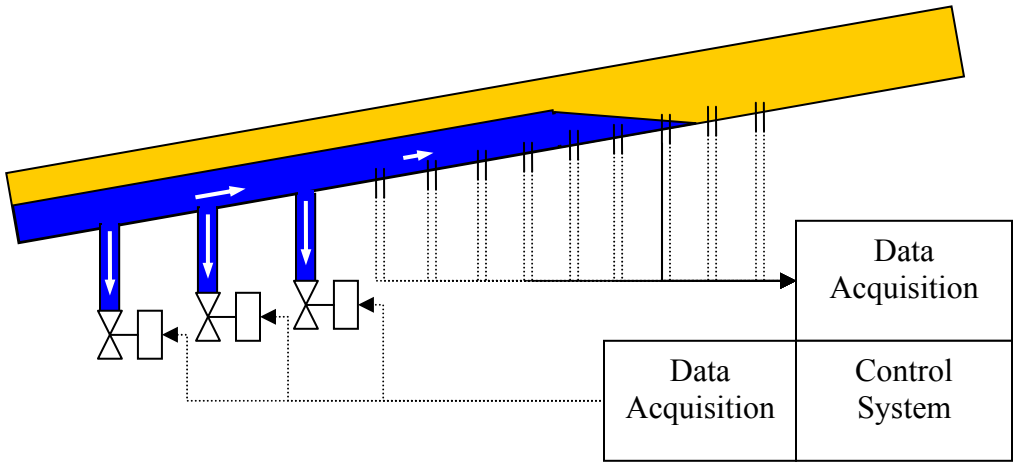


Figure 2.7: In-Line Water Separator Instrumental Set-Up

2.5 Principle of Operation

As discussed earlier, two stratified (separate) layers of water and oil flow concurrently in the uphill section of the pipe in order to separate water from oil by extracting all the water that is entering the system at the extraction zone using the measured parameters from the instrumented pipe section.

At the instrumented section, conductivity probes detect water presence and the read signal will determine the location of the water layer front (Process Variable ***PV***), and

then passes it on to the control system where it will be compared with a set-point (***SP***) value. The error between the set-point value (***SP***) and the Process Variable value will be used to make the decision in the control system. In case (***SP-PV***) is positive, which means that the water layer front is below the set point, signal will be sent to close the motor-operated-valves (***MOV***) to reduce water extraction. However, in case the (***SP-PV***) is negative, which means that the water layer front is above the set-point, the signal will be sent to open the ***MOV*** in order to increase water extraction.

The objective of the control system is, therefore, clear. It is to control the axial extent of the water layer front within an operative range near the set point.

Chapter 3

ARTIFICIAL NEURAL NETWORKS

3.1 Introduction

Many approaches are developed to identify system relation input/output. Most of the methods based on linear system approach like Least Square Methods, PEM, Maximum Likelyhood ..etc. These methods have many limitations to identify non-linear system because most of industrial processes are non linear processes [5] [6]. Now days Artificial Neural Network (ANN) is one of the popular methods used to identify non linear system because of its capability to identify almost any nonlinear function. The research field of ANN was conducted by two extremes of sciences: people who are interested in solving problems (like engineers, computer scientists, and people in industrial sector), and people who are interested in understanding living system (like biologists, psychologists, philosophers, mathematicians, and physicists) [2].

ANN applications are widely used in engineering, for example, in function approximations, pattern recognition, system classification, communications, signal & image processing, power systems, and adaptive controllers. Neural Network's potential in

learning linear and non linear systems is because of its non-linear and multi-stage structure [2] [3] [7].

3.2 Biological Understanding

In biology, any living animals have nervous system composed of a few of hundreds of neurons in simple creatures or hundreds of billions in the human brain. Neuron sometimes called nerve cell process the information. The neuron is consists of: cell body (soma), and two types of branches: the axon and the dendrites. The cell body composed of nucleus that holds the information of the hereditary traits and plasma which is responsible for producing material needed by neuron.[8] [7]

As operational, the neuron receives the impulse signal from other neuron through its dendrites (receiver), where it is processed by the neuron nucleus. The signal is then transmitted along the axon (transmitter) through its stranded branches. The terminal of the strands called synapses which are an elementary structure of the functional unit between two neurons (axon and dendrites). When the impulse signal reaches the axon synapse, certain chemical called neurotransmitters are released. This chemical defuses across the synaptic gap in order to enhance or inhibit the receptor neuron's tendency to emit electrical impulses. The effectiveness of the synapse is adjusted by the signal adjusted by the signal passes through. Therefore, the synapses can learn from the activities they contribute. This is called history of the activities which is possibly the responsible for the human memory. Figure 3.1 shows a sketch of biological neuron.[7]

The cerebral cortex in human is flat sheet of about 2~3 millimeters thick having surface area of about 2,200 cm^2. It contains about 10^{11} neurons that are massively connected. Each neuron is connected to 10^3 to 10^4 other neurons. The human brain contains approximately 10^{14} to 10^{15} interconnections.[7]

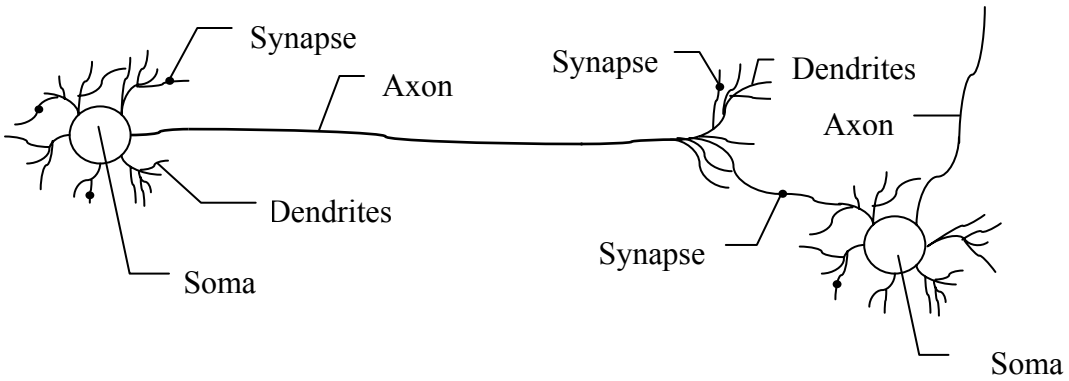


Figure 3.1: Biological neuron

3.3 ANN History Overview

The work in Artificial Neural Networks started in the early 1943. Since that time till today, two pair of revolutions were developed in this field of research. The first revolution started in 1943, two American scientists Warren McCulloch and Walter Pitts published their startling results in Network constructed by Binary Decision units (BDNs). Each of the inputs (typically in form of vecotor **X**) to the BDN is multiplied with its associated synapse weight (**W** in the form of vector) to obtain the effect of that input. $\mathbf{X}=[x_1, x_2, \ldots x_n]$, and $\mathbf{W}=[w_1, w_2, \ldots, w_n]^T$ The result of the BDN ***net(X)*** is the summation of each term over all the inputs over an activation function:

$$net(\mathbf{X}) = f(\mathbf{X}.\mathbf{W}+w_0)$$

$$= f(\sum w_i.x_i) \quad \text{, where } i=0\sim n,\ x_i=1 \qquad (3.1)$$

Where, w_0 is bias, used in some of the calculation and $f(.)$ is an activation function in this case is unit step function having 0 threshold. The positive weights match up with *excitatory* synapses, while the negative weights model *inhibitory* synapses. They showed that these networks can perform any logical function on its inputs. In analogy with biological neuron, the wires and interconnections models the axon and dendrites, connection weights model the synapses, and the threshold activiation function represents the activities of the cell body (soma). This was supporting the start for the development of human thought prototype [2] [7]. Figure 3.2 shows McCuloch-Pitts model of neuron.

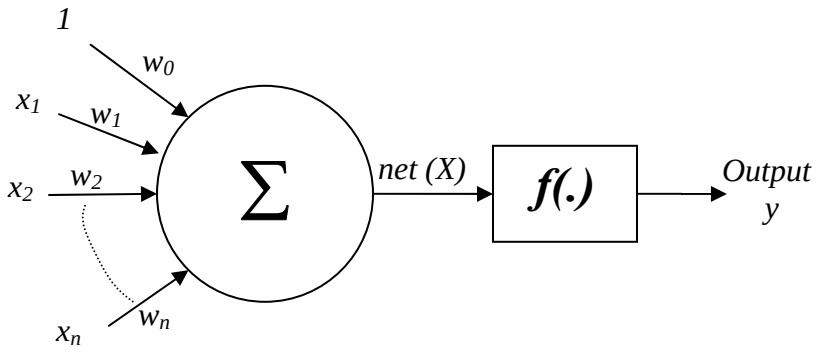


Figure 3.2: BDN computation- Graphical representation

It follows in 1960, the algorithm of LMS was discovered as training rule for adaptive elements [3]. The beauty of these types of networks is that they were presented the model of sort of nerve cell used in human brain to support the idea of thinking.

In 1962, Frank Rosenblatt, and several of his colleagues showed the training method of the BDNs, where they called it “perceptron”. The training method used what are called the connection weights. In this case, the connecting weight is the most important term in the neural networks calculation and their modification (training) is one of the open researches till the present day. So far, the most effective training algorithm is not established and yearly many proposals are derived to train these weights [2].

The basic idea of training this network is by changing the value of the connection weights which will lead to the output result to be improved to be closer to desire value. When Rosenblatt introduced this idea he showed that this model is duplicating to some extent the activity of the human brain. While McCulloch and Pitts indicated that the BDNs could solve any logical task, Rosenblatt has added that these networks can also, be trained to classify any pattern set [2].

By 1969 many people lost their attraction in Neural Networks when Marvin Minsky and Seymour Papert showed that perceptrons are limited and can not solve some very simple pattern classification like the XOR function i.e separating the binary pattern (-1,-1),0; (1,1),0; from the pattern (1,-1),1; (-1,1),1 Figure 3.3.To solve such problem, it is essential to have neurons which are not seen in the out world. These neurons called “Hidden Neurons” which is difficult to train by making their output closer to the desired values given by the training set. So, if the case of linearly inseparable problem there must be hidden layer of neurons in order to convert it to linearly separable problem [2].

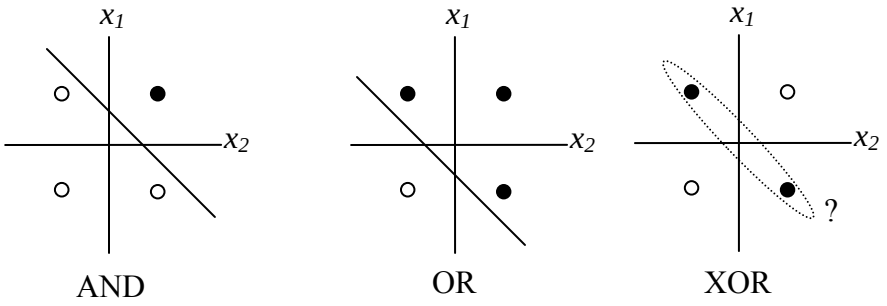


Figure 3.3: XOR problem with BDN classification. Geometric presentation of the linearly separable for the AND and OR function, and non-linearly separable function XOR.

In addition to that, Minsky and Papert has also showed that training such networks especially with hidden layer takes very long time and it is directly proportion with number of neurons in hidden layers [2].

The researches in Neural Networks then went into relatively slow growth since Minsy and Papert discovery. The second revolution started when the power of computers increased and allowed simulation with complicated calculations. Meanwhile, the difficulties of training hidden neurons was solved by the Backpropagation algorithm which was introduced by Paul Werbos in 1974, and rediscovered by Parker and LeCun in 1985 [2] [3]. Backpropagation allowed the error between the Neural Network result and the desired result to be propagated backwards from output Neuron towards input neuron passing through the hidden neurons. This method gives very precise modifications of the weights on the hidden neurons [2].

Another source for further researches in Neural Networks was the result from the paper of John Hopfield in 1982 and the related work of Grossberg and collaborators in 1983. The analyz the dynamics of networks by introducing powerful method based on Lypunov functions. It showed how the BDNs networks can be tied to each other and asynchronously updated and can be seen to develop time as if the system was decreasing the energy mount to find a minimum. Hopfield showed the possibility to form the energy landscape in order to find out the desired set of minima [2].

3.4 Neural Network Types

From the above we can understand that there are two extreme of the architectures of the networks: feedforward networks and recurrent networks. In the first type of network, the inputs move through the network to become an output. However, the recurrent networks (like Hopfield Network) have constant feedback from the neurons of the network to each other [2] [3]. Each of these types of networks can be analogized with two different topologies the first one is line networks and the other is circular networks [2] [7]. Figure 3.4 shows the classification of ANN family.

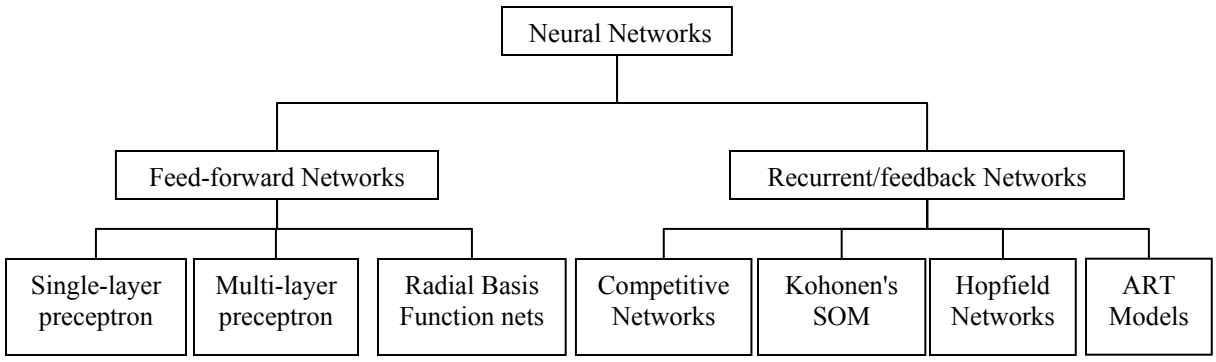


Figure 3.4: Classification of the feed-forward, and recurrent/feedback Networks

3.4.1 Training of Neural Networks

Training of Neural Networks can be classified in three types supervised, reinforcement, and unsupervised. The supervised type of training is like Backpropagation where the targets are known and the networks tries to find out the proper weight by back propagating the error between the actual network outputs and the target output throughout the hidden layer. Once the target became equal to the network outputs the training will stop and the unit will be ready to use. In other words usually the supervised training has set of inputs and outputs [2].

The second type of training is reinforcement training algorithm, some time they call it “punish/ reward” or “bootstrapping”. This type of training can solve problems when uncertainty in the error signal where supervised training method can not be applied due to non available of set of input and output data [2][3][8]. In this case, the Neural Networks has a reinforcement signal which is indicating how the neural network is good.

This signal found out using “critic”. After that, neural networks will use this information in the training procedure to update the concerned weight [8].

In unsupervised training algorithm, the network adjusts its weights by itself using the input only. The target data are not used, so that it cannot find out the errors as a feedback for learning. Furthermore, the output neurons are trained to respond to clusters of patterns from the inputs based on some criteria. In this case, the neural networks discover the statistical features on the input and will update the weights to find out the output. If there are one or more criteria, the networks can find out any existing regularities, patterns, classifications ...etc. So, unsupervised learning the network adapts itself so that similar inputs produce the same representative outputs [2] [3] [8].

In this thesis, supervised training will be demonstrated using Radial Basis Function Networks.

3.5 Feedforward Networks

In general, ANN is Multilayer network consist of input layer, output layer and one or more hidden layers. Each layer consists of number of neurons which each of them is connected with all neuron of previous and next layer. Each connection has its associated weight. In addition, each neuron consists of linear calculation followed with nonlinear function at its output end [2] [8] [9] [10]. As the architecture of the ANN become complex, as its capability to learn nonlinear function become better. But in this case, the

training of these neural networks became very difficult [5][6][2]. Fortunately, it has been demonstrated that with no more than one hidden layer can map almost any nonlinear function

3.5.1 Single-Layer Preceptron

Single Layer Preceptron is the simplest form of the Neural Networks which is used for the classification of linearly separable set of data. The organization of this method is in Figure 3.5. As mentioned earlier the computation of this method is similar the BDN calculation:

$$net(\mathbf{X}) = \mathbf{X}.\mathbf{W}+w_0$$

$$y = f(\sum w_i.x_i) \quad \text{, where } i=0\sim n,\ x_i=1 \qquad (3.2)$$

The of the activation function is *f(.)* is always a nonlinear function and its selection depends on the application of the Neuron. In other ward, it might be function of threshold of zero if the desire output is binary (0, or 1).

$$f(x) = 0 \quad \text{if} \quad x<0$$

$$= 1 \quad \text{if} \quad x>0 \qquad (3.3)$$

However if the output of the ANN is a real number, the activation function might be a sigmoid function. [7] Other typical types of activation function can be seen in the Figure 3.5

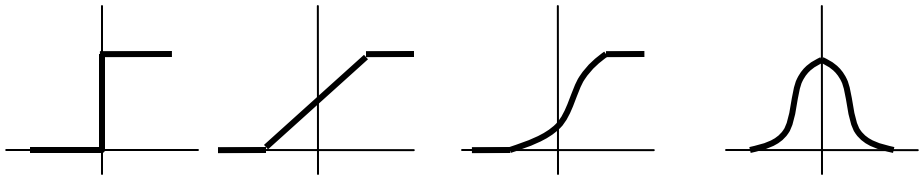


Figure 3.5: Typical activation function (a)Threshold, (b)piecewise linear, (c)sigmoid, and (d) Gaussian

As mentioned in the previous section the training algorithm for the weight is opened for the research and there are many algorithms is defined to train the weights: like the Sequential Least Square SLS, or Recursive Least Square RLS methods. In this section we will explain the training algorithm using the Normalized Least Mean Square NLMS.

Before we start speaking about NLMS we have to stop a while in Least Mean Square (LMS), which is one form of the family of stochastic gradient algorithms, based on deterministic gradient method. This technique was introduced by Widrow & Hoff in 1959. LMS technique is widely used in online estimation of taps weights that solves the Wiener Hopf equation. Steepest descent technique is used in the estimation of Weiener filter [3], [8], [9]:

$$\begin{aligned}\mathbf{w}(n+1) &= \mathbf{w}(n) + \frac{1}{2}\eta\left[-\nabla J(n)\right] \\ &= w(n) + \eta[\mathbf{p} - \mathbf{R}\mathbf{w}(n)]\end{aligned} \tag{3.4}$$

Where **R** is the instantaneous auto-covariance and **p** is instantaneous is the cross-covariance between the input and the desired output. So, the steepest descent algorithm becomes:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta e(n)\mathbf{x}(n) \qquad (3.5)$$

Where $\mathbf{x}(n)$ is the input vector

$\mathbf{w}(n)$ is weight vector.

η is learning rate

$e(n)$ is the error from the previous estimate

Normalized Least Mean Square NLMS was introduced to adapt η such that changes in of tap coefficients is minimized. In this case is η adjusted so that $w(n+1)$ minimizes the error signal of $x(n)$ so that the disturbance reduces to the minimal [9]. So, in this case η became:

$$\eta = \frac{\tilde{\eta}}{\alpha + \mathbf{x}^T(n)\mathbf{x}(n)} \qquad (3.6)$$

where $0<\tilde{\eta}<2,$ and $\alpha \geq 0$

α is regularization factor that will stabilize the calculation numerically. Therefore, LMS became the famous equation of NLMS.

In this type of algorithm each neuron of the Preceptron layer has a simple NLMS calculation with nonlinear function at its output. NLMS can be describes as below [9]:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\eta e(n)\mathbf{x}(n)}{\alpha + \mathbf{x}^T(n)\mathbf{x}(n)}$$
$$y(n) = \mathbf{w}^T(n)\mathbf{x}(n) \qquad (3.7)$$
$$\phi(n) = f(y(n))$$
$$e(n) = d_i - \hat{y}$$

Where *f(.)* is nonlinear function, which commonly uses sigmoid function.

Figure 3.6 shows the block diagram of the hidden neurons and outer neurons calculations.

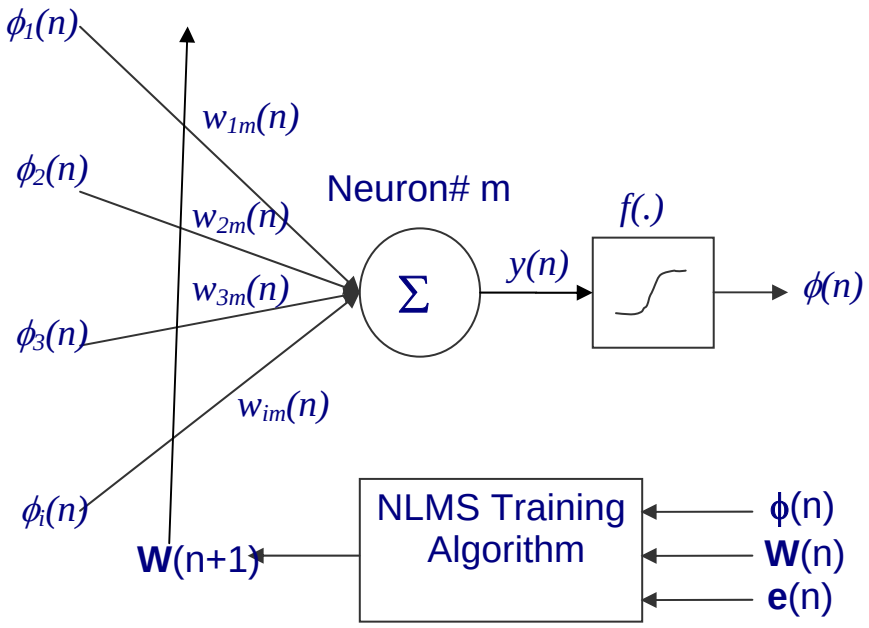


Figure 3.6: NLMS Training algorithm

3.5.2 Multilayer Preceptron

Multilayer perceptron (MLP) composed of one or more hidden layer. The generalized structure of MLP is shown in the next Figure 3.7.

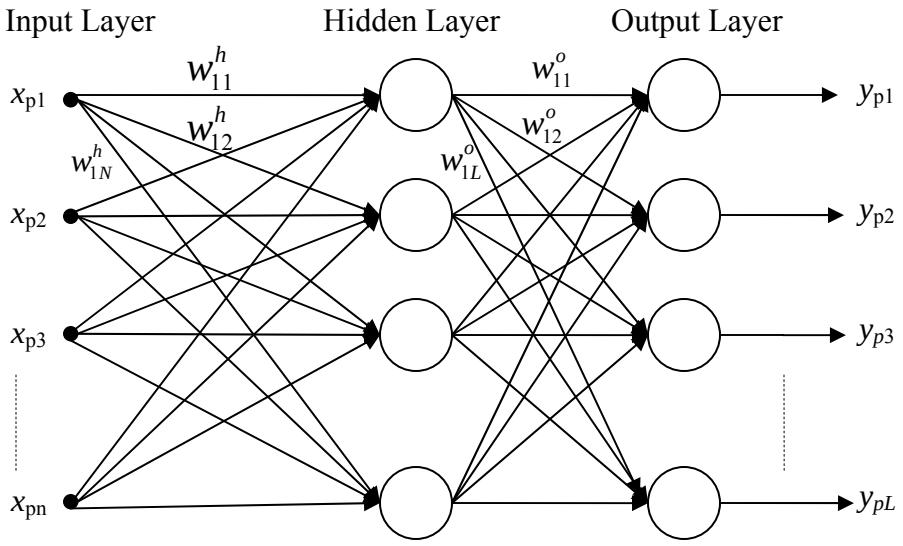


Figure 3.7: Multi-Layer Preceptron. It shows the Input Layer, Hidden Layer and Output Layer

The most common training method for the MLP Networks is the Backpropagation which is based on supervised learning algorithm. The algorithm is based on reduction of cost function which leads to the gradient descent procedure. The inputs to outputs calculation is the forward computation. The error between the target value and the calculated value is used to update the weight in the backward calculation; in other ward the error is backpropagating from the outputs toward the inputs to update the weights. This type of weight training algorithm is called Backpropagation. The general technique for the Backpropagation is as below [11] :

1. Calculate the output vector by applying set of the input vector. Using the equation of the BDN calculation
2. Compare the actual value with the desire value and calculate the error
3. Determine the direction of weight updating in the (+ ve) or in the (– ve) direction

4. Determine the amount of change of each of the weights
5. Apply the new correction to each of the weights
6. Repeat the 1 to 5 to all the training set until the error for all vectors is reduced to the minimum.

To define the exact procedure of the Backpropagation, step 6 asks for criteria for the minimizing the error. In the Backpropagation uses the Generalized Delta Rule GDR. The cost function to minimize, in this case, is the sum of square errors.

$$E_p = \frac{1}{2}\sum_{k=1}^{M} \delta_{pk}^2 \tag{3.8}$$

The Backpropagation method can be summarized as below [11]:

1. Get the Training set of the input values as a vector $\mathbf{x_p}$ = [x_{p1}, x_{p2}, … x_{pN}]', and output vector $\mathbf{y_p}$ = [y_{p1}, y_{p2}, … y_{pL}]'
2. Initialize the weight with random numbers.
3. Compute the input values to the hidden layer:

$$net_{pj}^{h} = \sum_{i=1}^{N} w_{ij}^{h} x_{pi} + w_0 \tag{3.9}$$

4. Compute the output from the hidden layer

$$i_{pj} = f_j^{h}(net_{pj}^{h}) \tag{3.10}$$

5. Compute the net-input value the to each neuron of the output layer

$$net_{pk}^{o} = \sum_{j=1}^{L} w_{kj}^{o} x_{pj} + w_0 \tag{3.11}$$

6. Compute the output from the output layer:

$$o_{pk} = f_k^{o}(net_{pk}^{o}) \tag{3.12}$$

7. Compute the error from the output neuron:

$$\delta_{pk}^{o} = (y_{pk} - o_{pk}) f_{k}^{o/}(net_{pk}^{o}) \quad (3.13)$$

8. compute the error for the hidden neuron:

$$\delta_{pj}^{h} = f_{j}^{h/}(net_{pj}^{h}) \sum_{k} \delta_{pk}^{o} w_{kj}^{o} \quad (3.14)$$

9. Update the weight in the output layer

$$w_{kj}^{o}(n+1) = w_{kj}^{o}(n) + \eta \delta_{pk}^{o} i_{pj} \quad (3.15)$$

10. Update the weight in the hidden layer:

$$w_{ji}^{h}(n+1) = w_{ji}^{h}(n) + \eta \delta_{pj}^{h} x_{i} \quad (3.16)$$

11. Compute the sum of square error, if the value reaches to reasonable value, stop the algorithm and use the calculated weight for the prediction or function approximation.

The Backpropagation algorithm is used only if we have set of input/output data. These data normally divided into two groups. The firs group is used to train the network till satisfactory Sum of Square Error reached. Furthermore, the other group is used for the validation of the model. Therefore, the training of the MLP using the Backpropagation training technique is always performed offline. [7] [11]

3.5.3 Radial Basis Function Networks RBFN

Radial Basis Function Networks (RBFN) is one the ANN techniques that has to do with distance criterion with respect to center, like circle, ellipse or Gaussian. The main advantage of the RBFN over the MLP is that it has simpler structure and straight forward

training algorithm. The RBFN network has three layers: the input layer, one hidden layer, and output layer. We can say it is a special class of Multilayer feed-forward Network. The activation function is always Radial Basis Function (RBF) and it is available only at the hidden layer. The common function used is the Gaussian Function. There are many method for training this Networks some of them are online and the others offline. [7] [12]

As mentioned in previous paragraph, RBFN is different from MLP in its structure. It consists of single hidden layer, while MLP may consists of one or more hidden layers. In RBFN networks, the transfer functions from the input layer to the hidden layer are non-linear, and from the hidden layer to the output layer are linear. In the other hand, in MLP, the transfer functions of each hidden layer with the previous layer are nonlinear, and from the hidden layer to the output layer can be linear or non-linear. Furthermore, the RBF networks consist of radial basis function at the hidden layer where it will compute the distance between the input and the center of that RBF unit. However, MLP calculate the dot product between the input vector and the associated synaptic weights. Figure 3.8 shows the structure of the RBF. [12] [13]

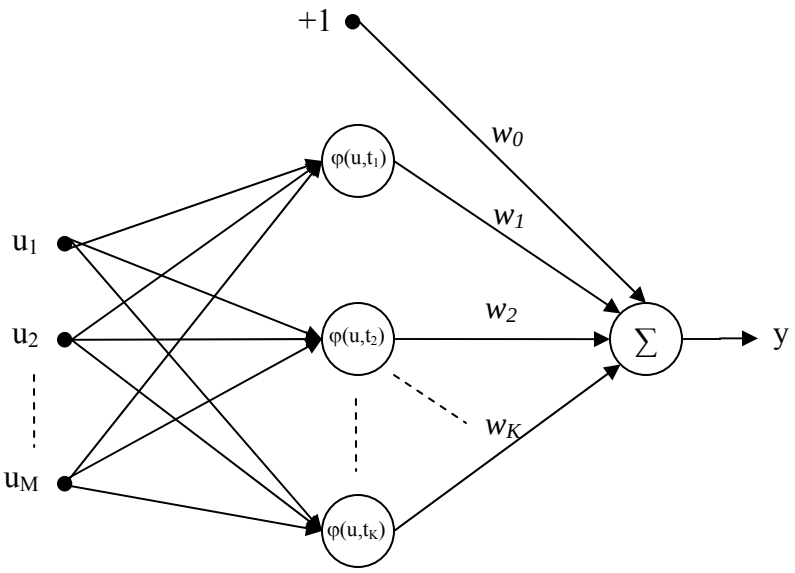


Figure 3.8: RBF Network

From the Figure, RBF have single output form. Therefore, the output function of the RBF will be:

$$y = \sum_{k=1}^{K} w_k \varphi(u; t_k) + w_0 \qquad (3.17)$$

Where $\varphi(u;t_k)$ is the k^{th} radial basis function that will calculate the distance between the input vector $\mathbf{u}$ and the center $\mathbf{t_k}$. Typical radial basis function unit is Gaussian function, but other function like the thin-plate spline function my be used. The Gaussian function is in the following format:

$$\varphi(\boldsymbol{u};\boldsymbol{t_k}) = exp[-(\boldsymbol{u}-\boldsymbol{t_k})^H \boldsymbol{C_k}(\boldsymbol{u}-\boldsymbol{t_k})], \quad where\ k=1, 2, 3, \ldots, K \qquad (3.18)$$

Where $\boldsymbol{t_k}$ is the center of the k^{th} radial basis function, and $\boldsymbol{C_k}$ is the width or smoothing factor. The superscript H is the Hermitian transposition. $\boldsymbol{C_k}$ can be chosen as follows:

$$\boldsymbol{C_k} = \frac{1}{\sigma_k^2}\mathbf{I} \tag{3.19}$$

Where **I** is the identity matrix and σ_k is the width of the Gaussian.

Therefore the Radial basis function becomes [12] [13]:

$$\varphi(\boldsymbol{u};\boldsymbol{t_k}) = exp(-\frac{1}{\sigma_k^2}||\mathbf{u}-\mathbf{t_k}||^2) \tag{3.20}$$

The overall equation of the RBF Networks becomes

$$y = \sum_{k=1}^{K} w_k \, exp(-\frac{1}{\sigma_k^2}||\boldsymbol{u}-\boldsymbol{t_k}||^2) \tag{3.21}$$

3.5.3.1 Fixed center training

The simplest method in the design of the RBFN is to select a set of fixed centers for the radial basis function units [12] [13]. These centers may be chosen randomly from the set of the training data. This approach is considered logical since the randomly selection would distribute the center according to its probability density function of the training data set. In this case the radial basis function will can be written as:

$$\varphi(\boldsymbol{u};\boldsymbol{t_k}) = exp(-\frac{K}{d_{\max}^2}||\mathbf{u}-\mathbf{t_k}||^2), \quad \text{where k=1,2,…,K} \tag{3.22}$$

Where K is the number of centers (RBF units in the hidden layer), and d_{max} is the maximum distance between the selected centers. Therefore, the width σ_k is fixed for all the RBF units:

$$\sigma = \frac{d_{\max}}{\sqrt{K}} \tag{3.23}$$

This selection will ensure that the there are not too sharp or too flat function of the RBF units.

Further to that, there are many methods used for the selection of the weight in for the output layer. Least square might be used for the selection of the weight values. In order to describe this method, we will define set of training data set $\{\mathbf{u}_i, d_i\}$, where $\mathbf{u}_i$ is the input vector and d_i is the desire output value at given input vector. Then we might define the interpolation matrix as bellow:

$$\boldsymbol{\Phi} = \begin{bmatrix} 1 & \varphi(\mathbf{u}_1;\mathbf{t}_1) & \varphi(\mathbf{u}_1;\mathbf{t}_2) & \dots. & \varphi(\mathbf{u}_1;\mathbf{t}_K) \\ 1 & \varphi(\mathbf{u}_2;\mathbf{t}_1) & \varphi(\mathbf{u}_2;\mathbf{t}_2) & \dots. & \varphi(\mathbf{u}_2;\mathbf{t}_K) \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ 1 & \varphi(\mathbf{u}_N;\mathbf{t}_1) & \varphi(\mathbf{u}_N;\mathbf{t}_2) & .... & \varphi(\mathbf{u}_N;\mathbf{t}_K) \end{bmatrix} \qquad (3.24)$$

$$\mathbf{d} = \begin{bmatrix} d_1 & d_2 & .... & d_N \end{bmatrix}^T$$

Where $\mathbf{\Phi}$ is the interpolation matrix of (NxK) dimension and $\mathbf{d}$ is the desire output of (Nx1) dimension. Further to that, we can define the weight vector having dimension of (Kx1) using the method of leas square:

$$\mathbf{W} = (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathbf{d} \qquad (3.25)$$

So, we can summarize the algorithm for training the RBFN with fixed center as below: [12]

1. For specific K number of radial basis units, select K centers randomly from the training data set, and compute the fixed width of the RBF units.
2. Define the interpolation matrix $\mathbf{\Phi}$ for the given set of N training data set.
3. Compute the weight vector $\mathbf{W}$ using the least square method technique.

3.5.3.2 Online Training method for RBFN

There are many approaches for training the RBFN online, like Recursive Hybrid Learning Procedure, and Stochastic Gradient Approach. This section will give an overview on the online training method for RBFN using the Stochastic Gradient Approach. [12] [13]

Stochastic Gradient approach is supervised learning process for RBFN which means that it is based on error-correction learning. The principle of this method is similar to that for the Least Mean Square LMS. [13] [12] In order to develop the technique, first we have to define the cost function (Sum Square Error):

$$\varepsilon = \frac{1}{2}\sum_{n=1}^{N}\left[d_n - \sum_{k=1}^{K} w_k \exp(-\frac{1}{\sigma_k^2}\left\|u(n) - t_k\right\|^2) \right]^2 \qquad (3.26)$$

Then we have to take the gradient of ε with respect to t_k and with respect to σ_k separately. This will lead to the following equations [12] [14]:

$$\begin{aligned} \mathbf{t}_k(n+1) &= \mathbf{t}_k(n) - \eta_t \frac{\partial \varepsilon}{\partial \mathbf{t}_k} \\ \sigma_k^2(n+1) &= \sigma_k^2(n) - \eta_\sigma \frac{\partial \varepsilon}{\partial {\sigma_k}^2} \end{aligned} \qquad (3.27)$$

Where $\mathbf{t}_k(n+1)$: centers at hidden layer k at $n+1$ iteration,

and $\sigma_k^2(n+1)$: Variances "Width of the Radial Basis" at k+1 iteration for k layer.

By solving the equation, we will lead to the Stochastic Gradient Algorithm for the design of the RBF. Pleas note that k now stands for the: [12] [14]

$$\mathbf{t}_k(n+1) = \mathbf{t}_k(n) + 2\mu_c e(n)w(n)\phi(n)\frac{\mathbf{x}(n) - \mathbf{t}_k(n)}{{\sigma_k}^2}$$

$${\sigma_k}^2(n+1) = {\sigma_k}^2(n) + \mu_\sigma e(n)w(n)\phi(n)\frac{\left\|\mathbf{x}(n) - \mathbf{t}_k(n)\right\|^2}{\sigma^2} \qquad (3.28)$$

$$e(n) = d_i - \hat{y}$$

Where:

μ_c : training rate of the Centers

d_i : Desire value

$\hat{y}$: Calculated value from the RBFN

$w(n)$: weights from the output layer

$\phi(n)$: Covariance matrix

μ_σ : Training rate for the RBF width

Chapter 4

CONTROL SYSTEM SETUP

4.1 ANN Control System Overview

Artificial Neural Networks has been used in many applications in the field of Computer and Engineering. Many of the applications were for pattern classification, clustering, function approximation, prediction forecasting, image processing, optimization, and control systems. The popularity of ANN is because of its capability to approximate (identify) almost any nonlinear system.

As indicated in Chapter 2, the objective of this work is to develop an adaptive control system to control the Water/ Oil Separator. In order to do so, Neural Network is chosen to be the base of this controller. In this chapter, we will illustrate the methodology and the program used for the development.

Figure 4.1 shows the structure of the System Block Diagram. It consists of Hardware and Software parts. The hardware part is covered in the input probes and output valves blocks. on the other hand, the other blocks: input processing, control

system and output processing blocks are implemented in the software. In the coming sections, more details will be provided for these blocks.

The complete program was implemented using LabView version 7.0. The software written for each block will be shown and explained in detail in its respective section.

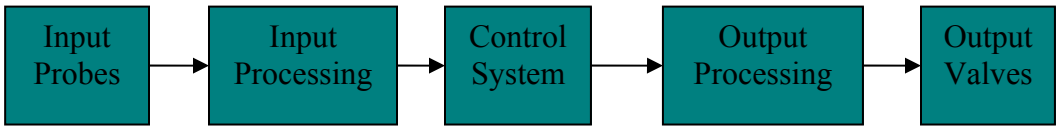


Figure 4.1 System Block Diagram

4.2 Input Probes Block

The first block to be described is the input probes. The overview of these probes has been illustrated briefly in Chapter 2. As described earlier, the conductivity probes are used to measure the conductivity of the liquid in which they are immersed and thus to indicate the axial location of the water in the pipe by knowing which probe is in the water and which one is in the oil. Figure 4.2 shows the circuit diagram for the connection of these probes with the Data Acquisition System CH0 to CHn. In this case, if the probe is immersed in oil it will show high resistance. This high resistance will result in low voltage generated at the analog input card CH0~CHn. On the other hand, if the probe is immersed in water it will show very low resistance which will result in high voltage

generated at the analog input card CH0~CHn. The measured voltage is then processed further to the input-processing block.

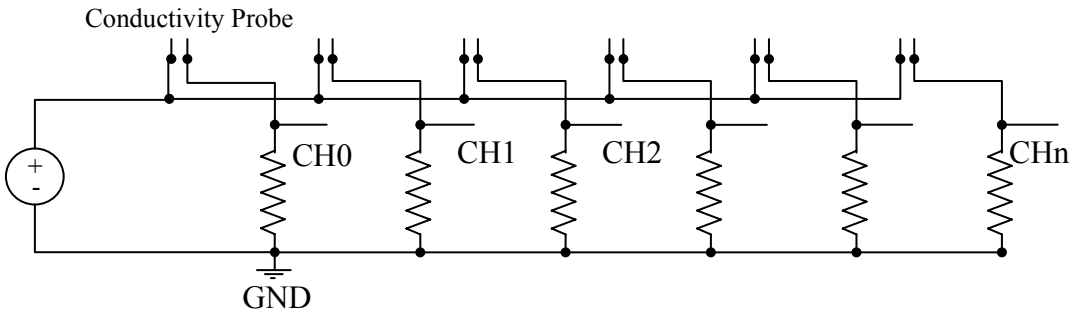


Figure 4.2 Analog Input signal to the Data Acquisition.

4.3 Input Processing Block

Input processing block is block used to recondition the signals coming from the input probes to a signal understood by the control system block. Figure 4.3 shows the general block details for input processing: Signal Conditioning, Fault detection, and the Axial level calculation. Figure 4.4 and Figure 4.5 shows the main programs for the implementation of the input processing block.

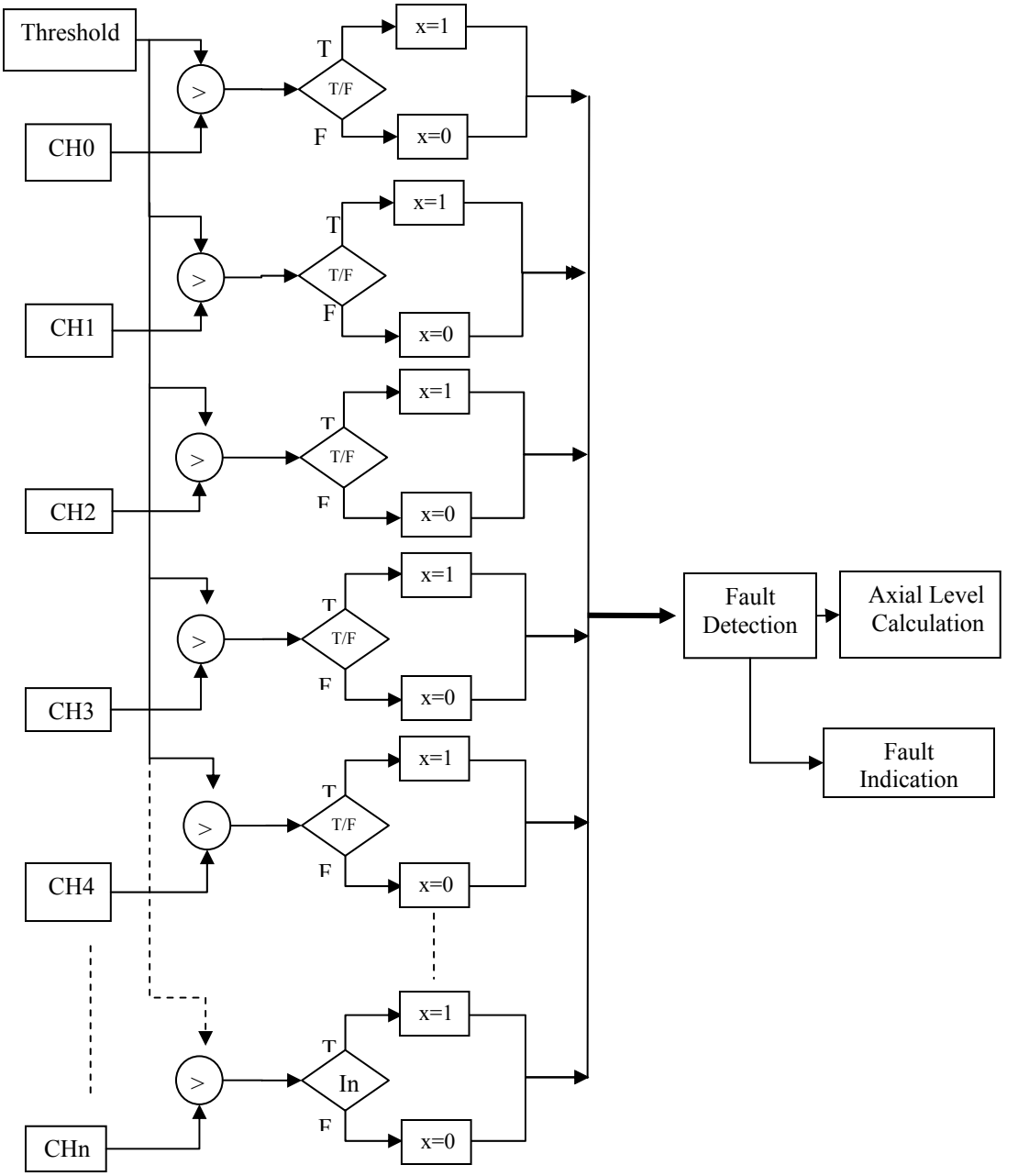


Figure 4.3: General Block Diagram from Input Processing

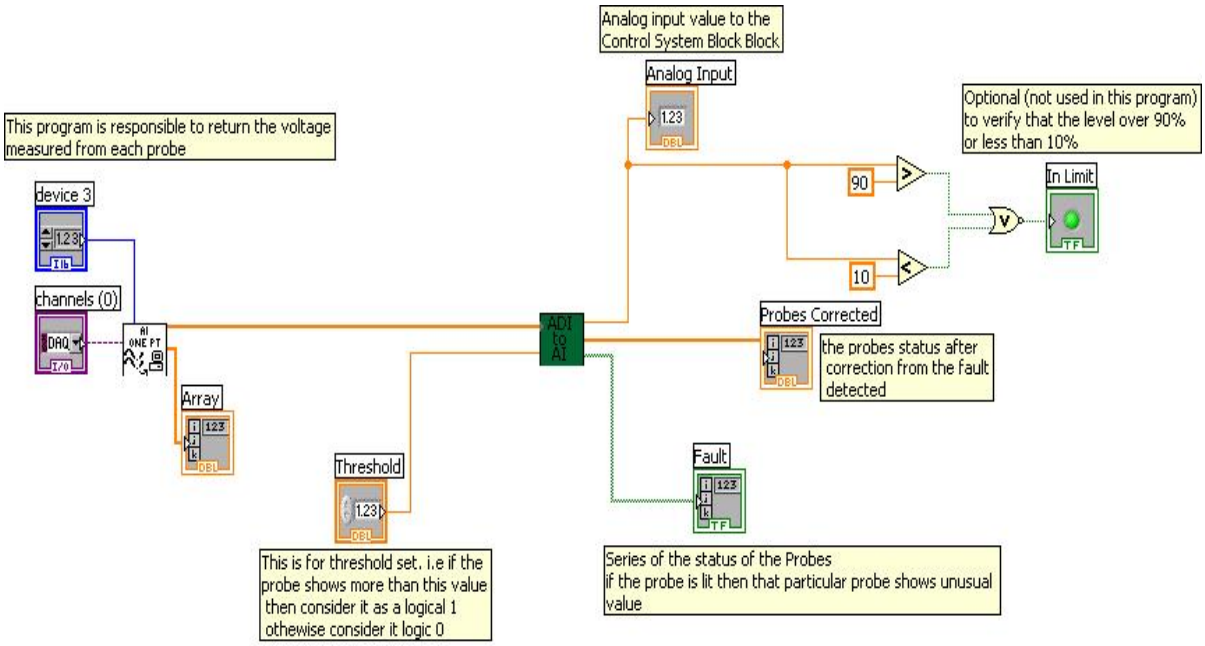


Figure 4.4 Input Feedback "Input Processing" program

Figure 4.4 shows the program for the input processing "Input_read" where it consists of ADI_to_AI subprogram shown in Figure 4.5.

In our case, the control system needs only single analog signal to indicate the axial location of the water. The first part is to extract the useful information from the probes. This is simply done by using a comparator that will give a value of 1 in case the probe voltage is greater than a certain threshold value. This threshold value shall be chosen properly to be grater than the voltage generated in case oil is covering the conductivity probe. The first part of Figure 4.3 shows the block diagram for signal conditioning. Similarly, the first part of the program (ADI_to_AI) in Figure 4.5 shows signal conditioning for the voltage measured. The inputs to this program are the threshold and the series of the measured voltages for the analog channels. The output from this part is a matrix having 1’s or 0’s.

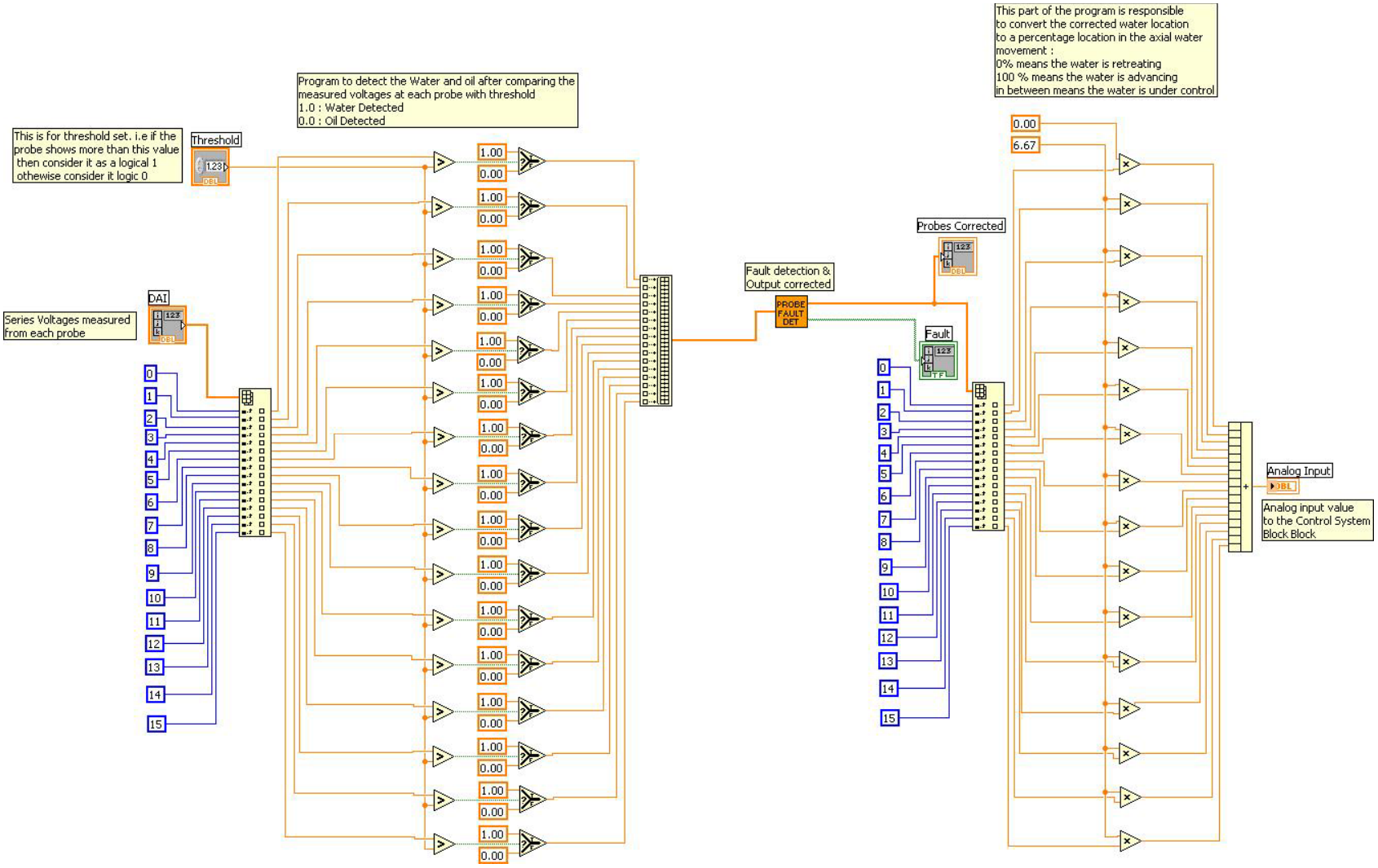


Figure 4.5: ADI to AI sub program. This program is responsible for the calculation of the axial level of water.

As shown, the matrix of the binary numbers is then passed to fault detection algorithm (Figures 4.3, 4.5). The aim of this algorithm is to verify that the values generated from signal conditioning are correct. For example, if probe ***n*** detects water (binary number = 1) whereas probes ***n-1*** and ***n+1*** do not detect water (binary number = 0), then probe ***n*** value shall be changed to no water (binary number = 1) and consider that this probe has failed. Another case, where probe ***n-1***, and ***n*** did not detect water, while probe ***n+1*** did, then probe ***n*** shall continue with no water detection, but a fault shall be indicated for probe ***n***. Table 4.1 shows all possible cases in Truth Table Format, along with its explanation for each case. Note that, there are 2 functions. The first one has the name '**output'** and it will show the new output value after analyzing the conditions at probes ***n-1*** and ***n***. The second function is ***Fault*** where the detected fault will be indicated to the user.

Converting the truth table to a Boolean Function will make the Detection Fault algorithm:

$$Fault = yx' + zy' \tag{4.1}$$

$$Output = xz + xy = x(y+z) \tag{4.2}$$

Figure 4.6 shows the algorithm for fault detection. In this program, a virtual probe, having a binary value of ***1***, is added for the analysis before the first probe. Another virtual probe, having a binary value of ***0***, is added after the last probe. In this program the input are the probes, and the output is made up of the fault, and the corrected probes.

Table 4.1: Fault detection Truth Table

	x	y	z			
S/n	Input(n-1)	Input(n)	Input (n+1)	Output	Fault (n)	Failure Description
0	0	0	0	0	0	No water indication, no failure detected at probe(n)
1	0	0	1	0	1	No water indication, failure indication at (n) since
2	0	1	0	0	1	No water indication since the probe(n-1) and (n+1) did not see flow, consider failure at probe (n)
3	0	1	1	0	1	No water indication since the probe(n-1) did not see level (if studied the case at (n-1) we will see this condition will not appear at any time compare with S/n 1, 5), probe (n+1) shows faulty signal since probe (n-1) did not see the level
4	1	0	0	0	0	No water indication, no failure detected at probe(n)
5	1	0	1	1	1	Water indication shall be available since there is level at probe(n-1) and probe(n+1)
6	1	1	0	1	0	Water indication available, and there is no faulty at probe(n)
7	1	1	1	1	0	Water indication available, and there is no faulty at probe(n)

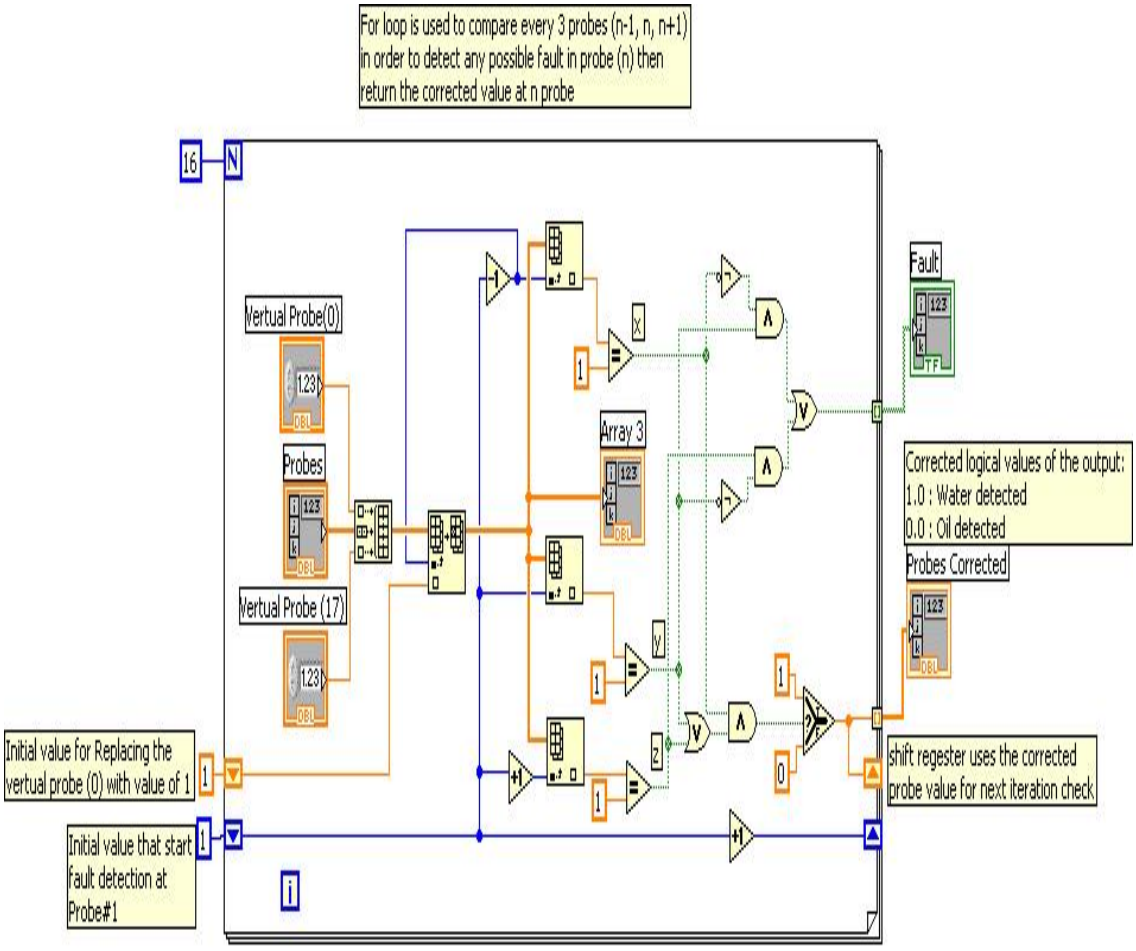


Figure 4.6 Fault Detection Program

The corrected signals measured by the fault detection program ***output*** function are then processed to the axial level calculation shown in Figure 4.5. This part of the program will multiply the digitized value (1, 0) by a scaling factor in order to calculate the axial location of water where it will be used further in the next block (Control System Block). Therefore, if we consider that the probes are equally spaced along the pipe, the total length can be expressed in percentage of the water level to represent the axial water layer movement. Whereas the first probe considered at 0%, the axial length calculation will be equal to:

$$h = \left(\sum_{i=1}^{N-1} \frac{1}{N-1}\right) * 100 \qquad (4.3)$$

4.4 Control System Block

The control system consists of two parts. The first part is the adaptive Neural Networks Controller that controls the system. The aim of this controller is to make sure that the axial level remains in the acceptable range, defined a priori. To achieve this, an anti-wind up algorithm is used along with the NN controller in order to overcome the problems that might arise during the operation of the motor operated control valves. Figure 4.7 shows the block diagram of the control system block. In our application, Feed-Forward Radial Basis Function Neural Network is used because of its advantages over the MLP.

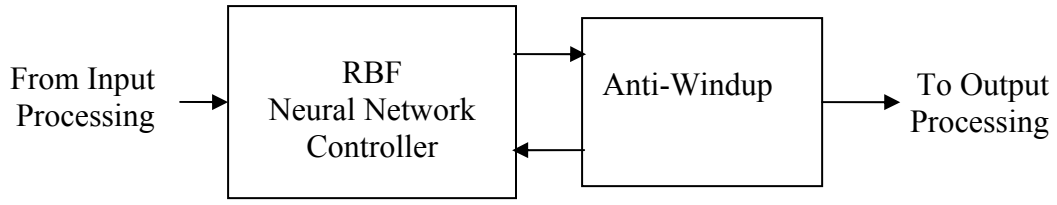


Figure 4.7: Control System Block Diagram

4.4.1 Neural Networks Based Controller

The basic principle of control is to identify the inverse of the plant transfer function. In this case, by cascading the stabilized plant, its inverse will make the loop gain unity. Therefore, the required input (set-point) will be tracked by the output from the stabilized plant. This principle is used to identify the inverse of the plant transfer function using an on-line identification for Neural Networks. Figure 4.8 illustrates the

general block diagram for the adaptive ANN controller. In this case the input to the ANN will be the set-point ***SP*** and the regression values of the axial level (process value ***PV***). The training criterion for the ANN in this case is the error between the ***SP*** and the ***PV***. As shown in the previous chapter, ANN is used to identify almost any non-linear function by minimizing the Sum Square Error ***SSE***. In our case the ***error*** = ***SP- PV***. By doing this, the plant axial level ***PV*** will try to track the ***SP*** when the ANN tries to minimize this error. Furthermore, if the plant characteristics changed because of its non-linearity, the on-line identification will immediately respond to the changes and tries to minimize the new error. From this point of view the ANN based adaptive controller is constructed.

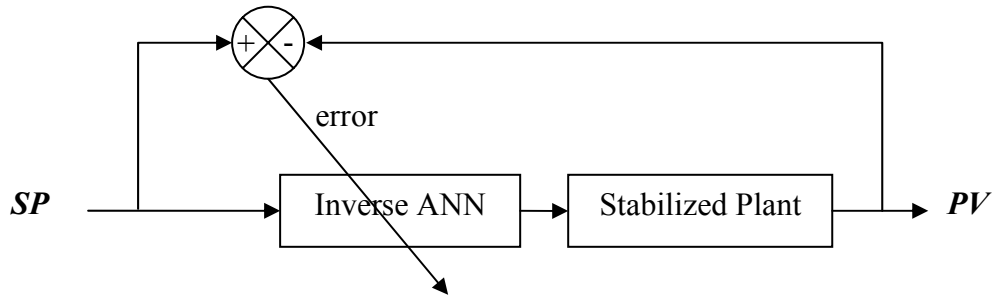


Figure 4.8: Inverse ANN as Adaptive Controller

To construct the ANN controller, block based programming is used. The inverse ANN is decomposed to blocks. The first level block is the adaptive blocks where Centers and RBF width are adjusted. The outputs from them are directly correcting the values in each of the RBF calculations. The calculation used for the RBFN Neural Networks is the same as what is described in section 3.4.3.2 :

$$\phi(n) = \exp(-\frac{\left\|\mathbf{x}(n) - \mathbf{t}_i(n)\right\|^2}{\sigma^2})$$

$$\mathbf{t}_k(n+1) = \mathbf{t}_k(n) + 2\mu_c e(n)w(n)\phi(n)\frac{\mathbf{x}(n) - \mathbf{t}_k(n)}{\sigma_k{}^2} \qquad (4.4)$$

$$\sigma_k{}^2(n+1) = \sigma_k{}^2(n) + \mu_\sigma e(n)w(n)\phi(n)\frac{\left\|\mathbf{x}(n) - \mathbf{t}_k(n)\right\|^2}{\sigma^2}$$

$$e(n) = d_i - \hat{y}$$

The output layer uses the calculation of NLMS where the calculations as below:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\eta e(n)\mathbf{x}(n)}{\alpha + \mathbf{x}^T(n)\mathbf{x}(n)} \qquad (4.5)$$

$$y(n) = \mathbf{w}^T(n)\mathbf{x}(n)$$

Where η is the learning rate for the NLMS.

The structure of the RBFN for 3 Motor-Operated-Valves (MOVs) is illustrated in Figure 4.9. There are 5 inputs to the hidden layer, which are the set-point and the four previous scan values of the axial level of the water layer front. The RBFN is made up of 3 groups of RBF_5X1, which is responsible of controlling each MOV. An anti-windup controller is added to the system in order to stop the ANN in case any MOV is saturated open or close. Further detail of this controller will be given in the next section.

The whole program is written using LABVIEW version 7.0. The coming paragraphs will give an overview of the program and how it is structured.

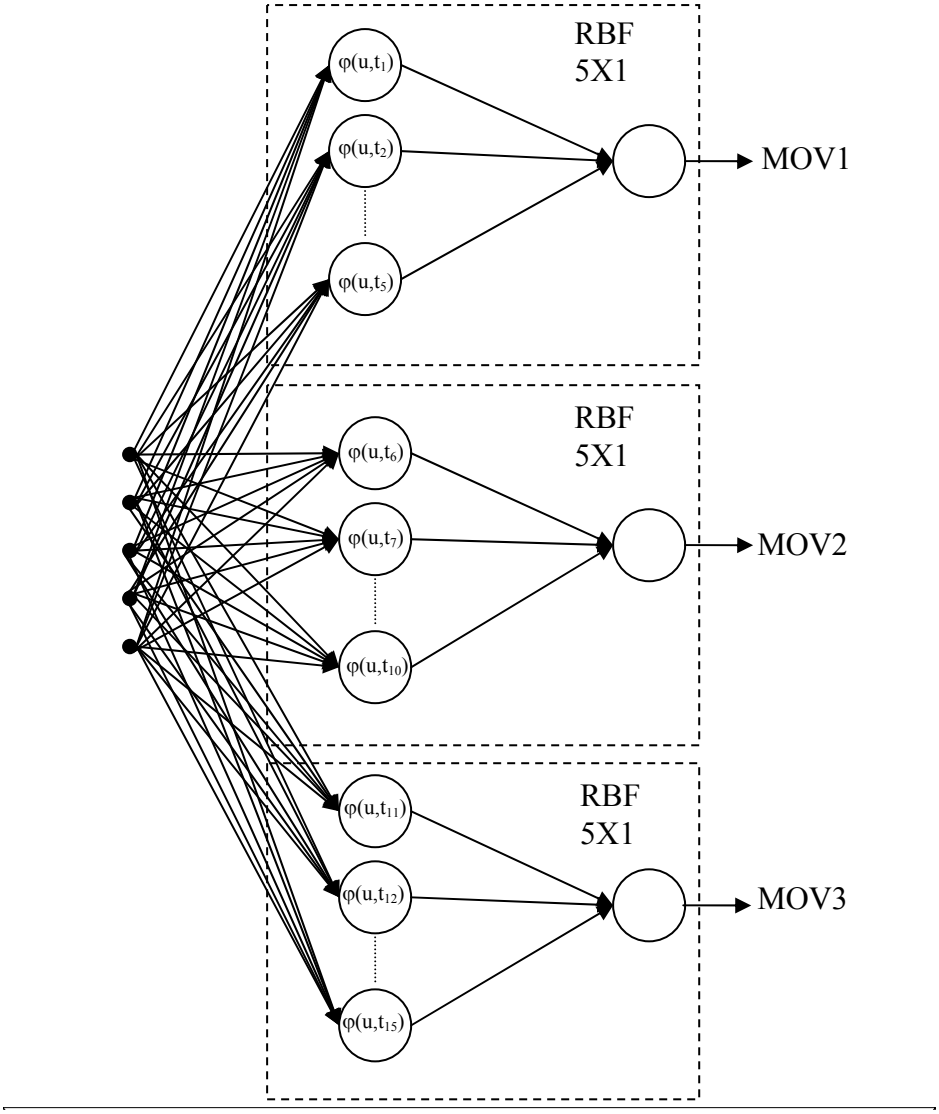


Figure 4.9: RBFN for 3 Valves Outputs

4.4.1.1 LABVIEW Program ANN Controller

As demonstrated earlier about the calculation for each neuron of the ANN, Figure 4.10 shows the overall program for the ANN and the anti-windup. It composed of the following subprograms: RBF_5X1, Level Feedback, MOVs, and Anti-windup. The Anti-windup will be explained in the next section while the MOV subprogram

will be explained in the output processing section. These subprograms are integrated together in order to form the inverse Artificial Neural Network Controller.

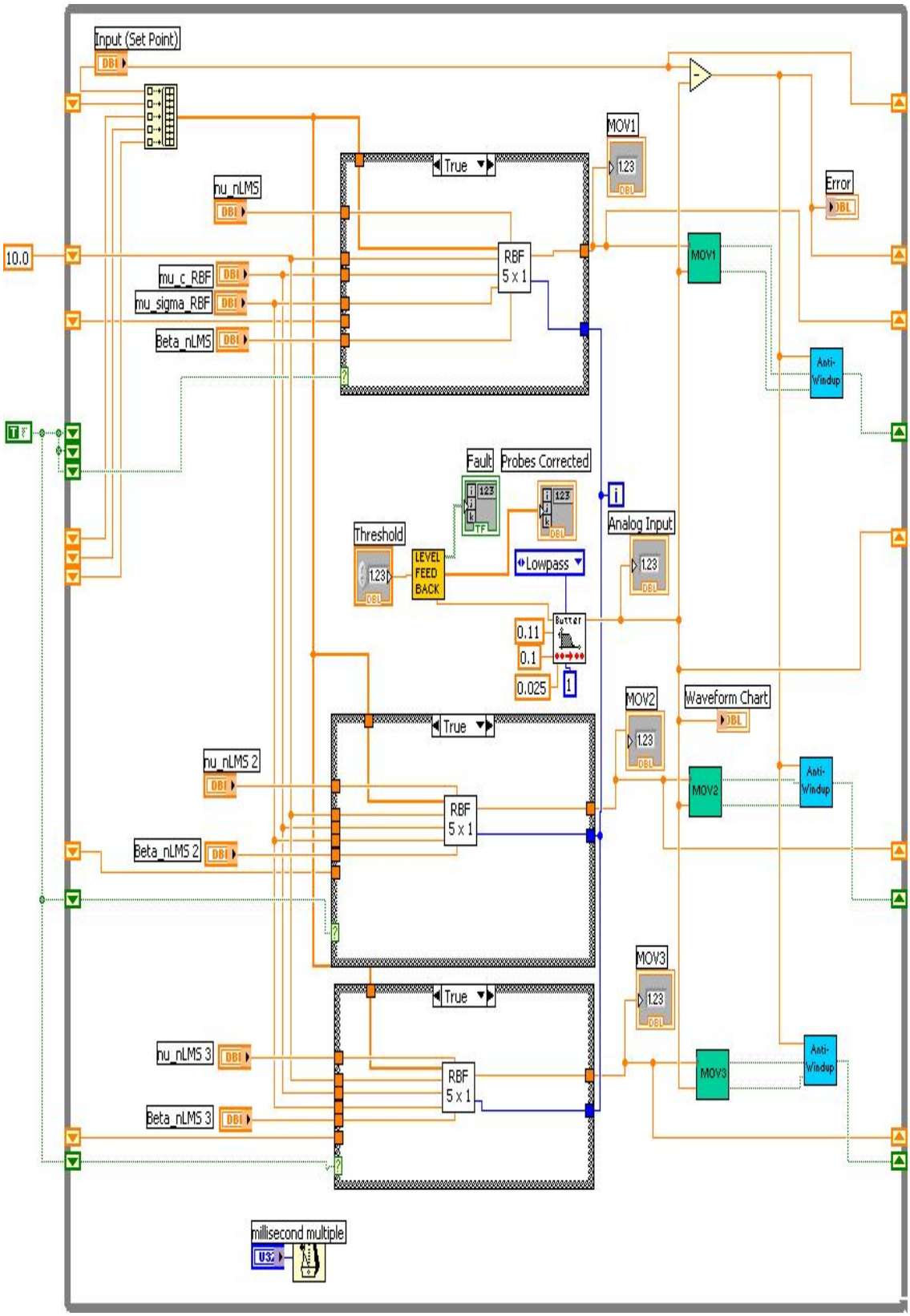


Figure 4.10: Main program of the Inverse ANN controller

In order to explain the program, Figure 4.11 shows the subprogram used for the RBF_5X1. There are three of them where each is responsible for the output to a single MOV. The input to this sub-program is the **input** (input layer: set-point and four of the previous scanned values of the axial level), **error** (between the set-point and the axial water level), **nu_nLMS** (training rate for the output layer), **Beta_nLMS** (regularization factor), **mu_c_RBF** (training rate for the center), **mu_sigma_RBF** (training rate for the RBF width), and the **iteration number**. The output from this program is the **output_NN**. Again, this program has further subprograms that are responsible for the calculations in each of the **RBF** in the hidden layer, and the **NLMS_for** which is responsible for the calculations in the output layer neuron. The **RBF** is the single unit in the hidden layer.

Figure 4.12 shows the program for the RBF. The inputs to this block are **Input** (**x**), **w**(n), **Error**, **mu_c**, **mu_sigma**, and the **iteration number**. The iteration number is used here in order to distinguish the first iteration loop used in order to randomize the weights initially. Again, this block has further subprograms for updating the center **C_cor** and the radial basis width **Sigma_cor** (Figure 4.13 and Figure 4.14).

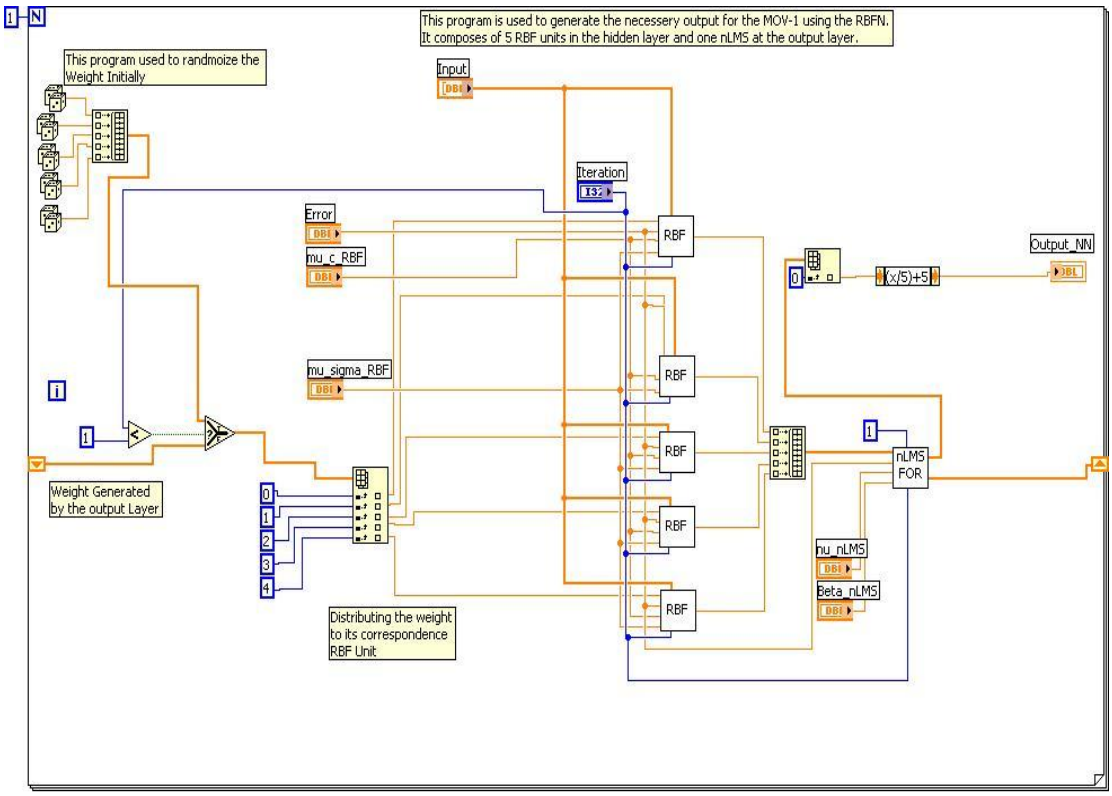


Figure 4.11: Program used for RBF_5X1 that contains the five RBF units in the hidden layer and a single neuron at the output layer nLMS_for.

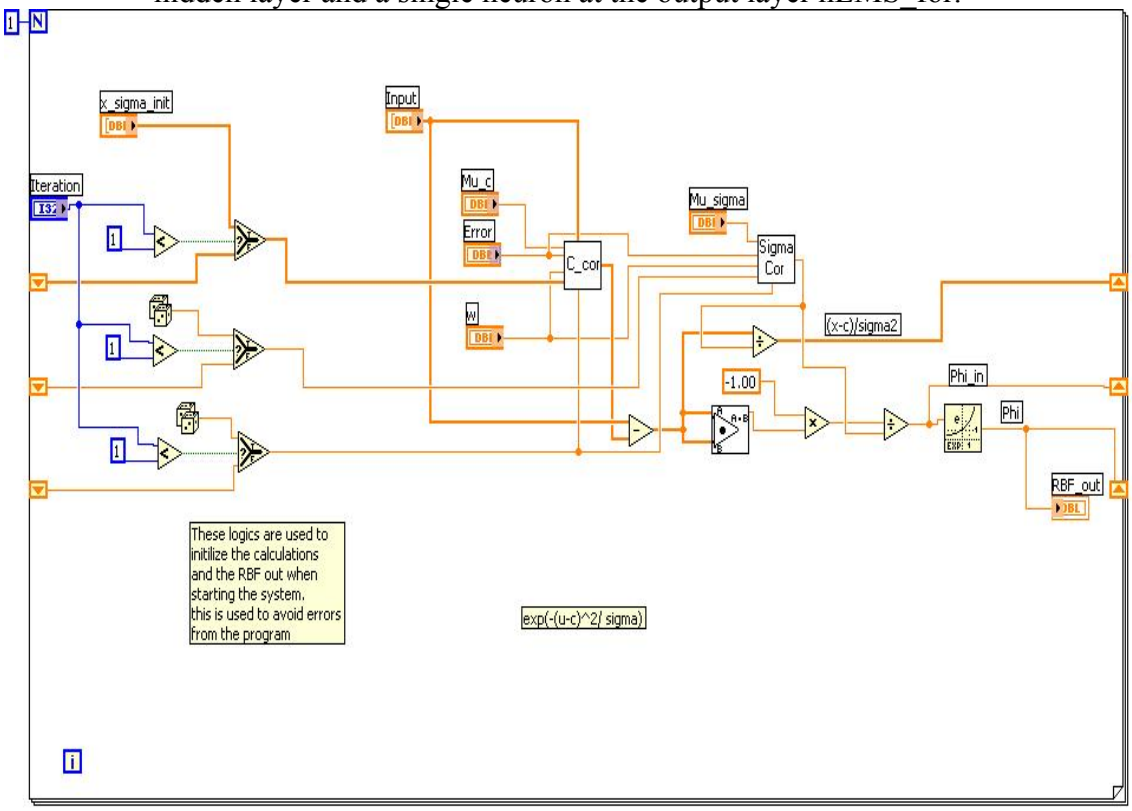


Figure 4.12: Program for Single RBF unit in the hidden layer

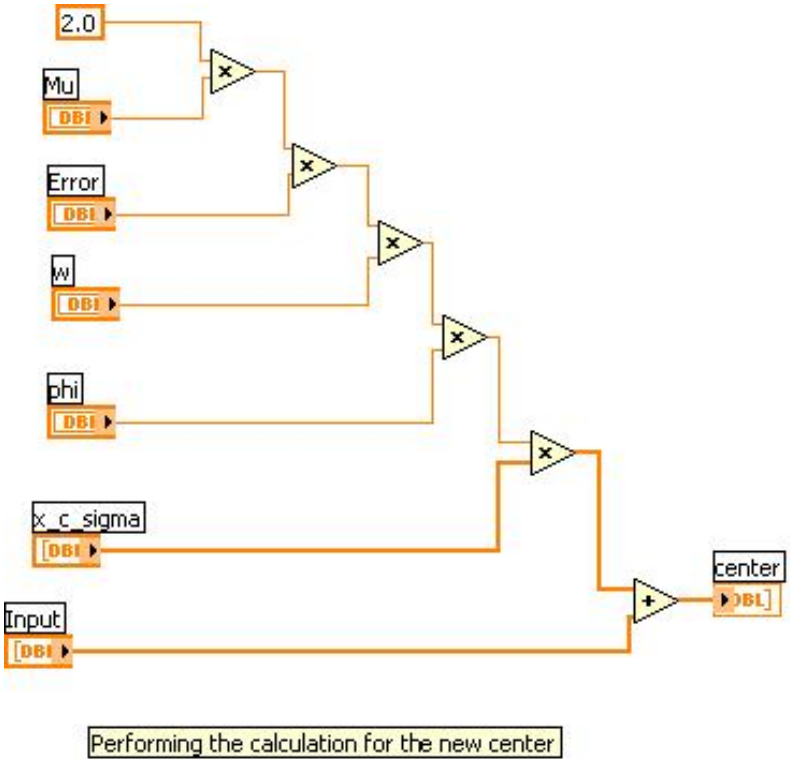


Figure 4.13: Center Correction Subprogram

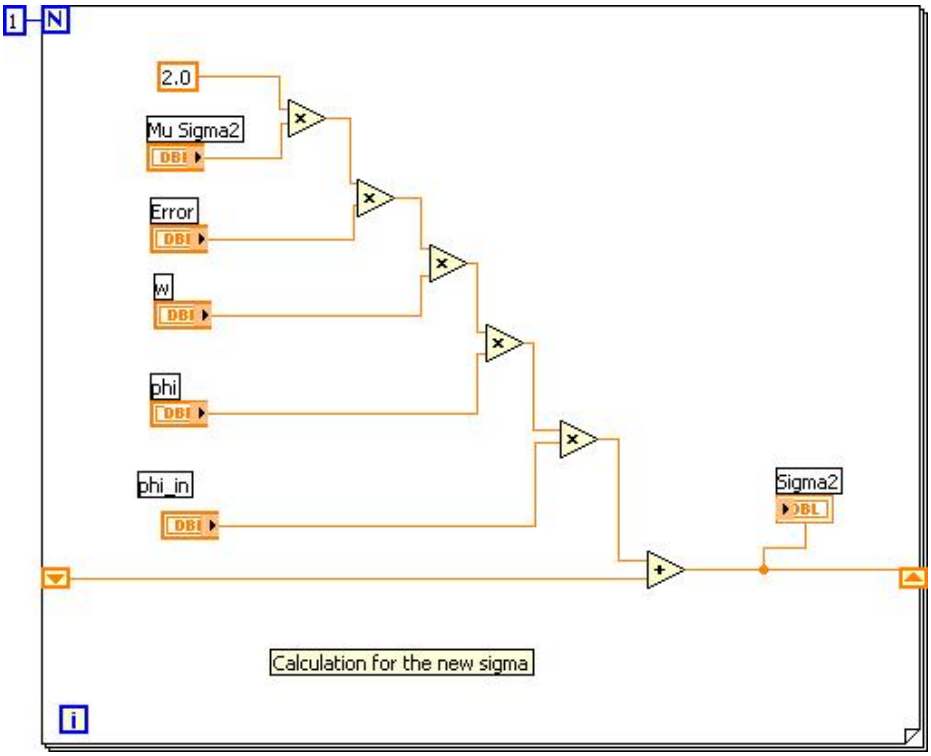


Figure 4.14: RBF width Correction **sigma_cor** for RBF unit

The output layer subprogram NLMS_for is shown in Figure 4.15. The inputs to the subprogram are the **input** (outputs from the RBF hidden layer), the **error**, **nu_nLMS** (learning rate for the weights), and **alpha_nLMS** (regularization factor).

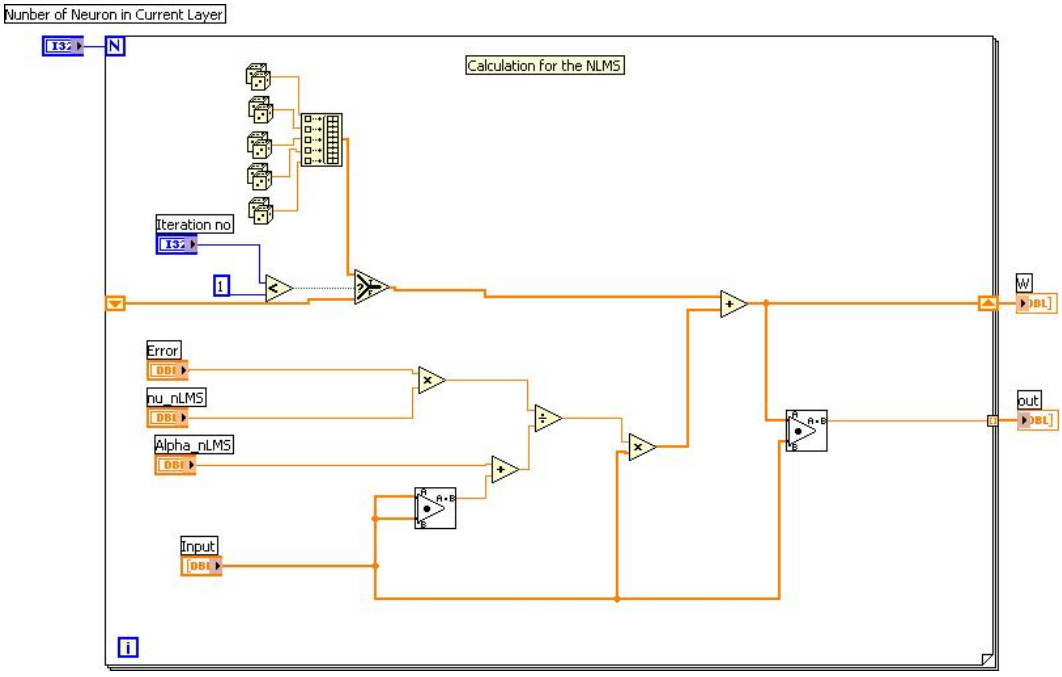


Figure 4.15: Output layer subprogram **NLMS_for**

4.4.2 Anti-windup controller

One of the difficulties facing the controller action is the case of valve becoming saturated either open or close. In this case the controller will continue sending a value to the control valve for controlling the level, although the valve is saturated open or close. This nonlinearity causes delay in the controller if the water level had crossed again the set-point, which is exactly similar to the wind-up problem with a PID controller [15] [16].

Many approaches are developed to minimize windup (Anti-windup). The first mechanism is achieved by measuring the difference between the controller output and

the plant input. This signal will equal zero only if both are the same. Hence, this situation will not affect the state of the controller. However, when this value is not equal to zero, a compensation mechanism is implemented. Another approach is by using nominal dynamic controller that uses feedback to update its stats. In this case, the saturated output is used to update the controllers [15].

We have modified the first approach in order to solve this problem, where the case of saturation and errors from the set point was studied in details. As a result, combinational LOGIC gates are used to identify the saturation in addition to error condition. The outputs from the combinational logic gates are used to stop the controller algorithm and keep the output to the valves in its condition in case saturation is detected. If the level crossed the set point again, the controller will start operating again (resetting the saturation detection).

Table 4.2 shows all of the conditions studied to develop the anti-windup controller. The over range and under range of the MOVs are detected if the output to the MOVs is grater than 100% or Less than 0%. Furthermore, the error is the difference between the set point and the axial level. In order to make the anti-windup controller more flexible, the error was studied in three zones. The first zone is when the error is greater than 10%, the second zone is when it is less than -10%, and the third zone is when the error is between +/- 10%. This way, hysterisis of +/- 10% can be developed around the set point. The results are studied and mapped into the output function **F** which is responsible for starting the algorithm of the ANN controller or sending out the old value to the MOVs. The value of Zero in the Table 4.2 means the condition is not satisfied, while the value of One means that abnormality is detected.

Moreover, if the valve is between 0~100% and the error is within +/- 10%, the algorithm of the ANN shall work till an abnormality is detected. However, if the level is less than the set point by 10 % or is grater than the set point by 10%, while the valve is within its operational range, the ANN controller shall operate in this case. Similarly, if the level is less than the set point by 10% and the valve is saturated open, the algorithm shall operate in this case to close the valve in order to increase the axial level. Finally, if the level is at a location greater than the set point by 10% and the valve is saturated close, the algorithm shall operate in order to open the valve to decrease the axial level.

On the other hand, if the axial level is more than the set point and the valve is saturated open, the ANN algorithm shall stop in order to decrease the level. Similarly, if the axial level is less than the set point by 10% and the valve is saturated close, the ANN algorithm shall stop in order to increase the level. All these cases were studied in detail in Table 4.2.

Table 4.2: Truth Table for the Anti-windup sub-program

S/n	MOV		Error		Output	Description of process
	Under range	Over range	-10%	10%	F	
0	0	0	0	0	1	valve within range and error within range, algorithm shall run
1	0	0	0	1	1	level is less than set point and valve is normal, algorithm shall run
2	0	0	1	0	1	level is more than set point and valve is normal, algorithm shall run
3	0	0	1	1	X	don't care condition
4	0	1	0	0	1	error within range and valve is saturated close, algorithm shall run, tuning shall be careful & error range to be considered
5	0	1	0	1	1	level is less than set point and valve is saturated open, algorithm shall run to close the valve
6	0	1	1	0	0	level is more than set point and valve is saturated open algorithm shall stop to decrease the level
7	0	1	1	1	X	don't care condition
8	1	0	0	0	1	level is normal and valve is saturated open, algorithm shall run, tuning shall be careful & error range to be considered
9	1	0	0	1	0	level is less than set point and valve is saturated close algorithm shall stop to increase the level
10	1	0	1	0	1	level is more than set point and valve is saturated close, algorithm shall run to open the valve to decrease level
11	1	0	1	1	X	don't care condition
12	1	1	0	0	X	don't care condition
13	1	1	0	1	X	don't care condition
14	1	1	1	0	X	don't care condition
15	1	1	1	1	X	don't care condition

Now if we consider MOV's under range to be **W**, MOV's overrange to be **X**, the error less than -10% to be **Y**, and the error grater than +10% to be **Z**, then the truth table can be expressed using a simple Boolean function:

$$F=Z'W'+ZX'+Z'Y' \qquad (4.6)$$

Now, in order to implement the anti-windup controller using LABVIEW, the Boolean function is written as a subprogram. Figure 4.16 shows the anti-windup subprogram. As indicated in Figure 4.6, anti-windup is affecting the operation of the ANN controller algorithm. In this case, if the function **F** is true then the algorithm of the Case Program will operate the ANN algorithm (Figure 4.17), but if **F** is False then the value to be sent to the valve is passed through the case program which will hold the old value of the MOV (Figure 4.18).

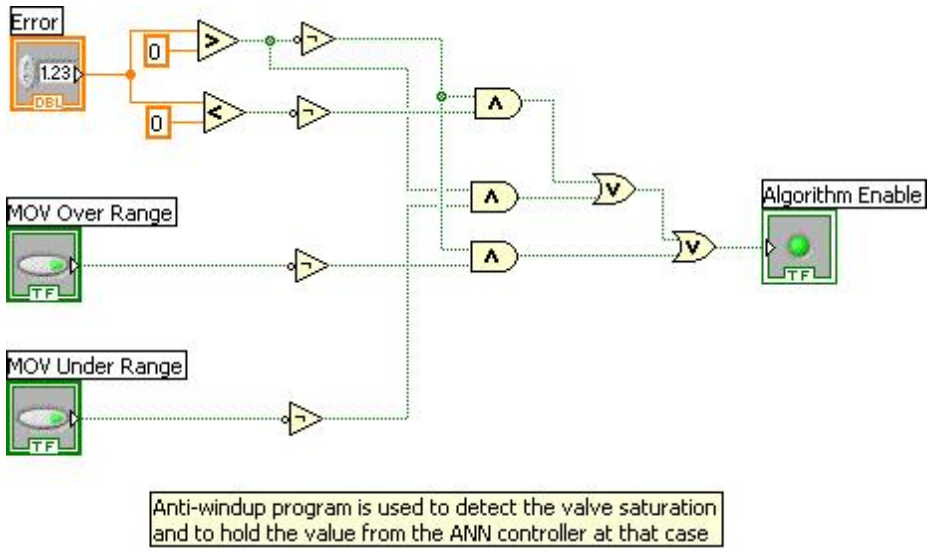


Figure 4.16: Anti-Windup Sub-program

Anti-windup is implemented against each MOV alone, so each RBF_5X1 will work alone. Therefore, in case the MOV1 is saturated open or close while the axial level is abnormal, its associated ANN algorithm will stop. The stoppage of MOV1 Controller does not mean that MOV2 or MOV3 will stop.

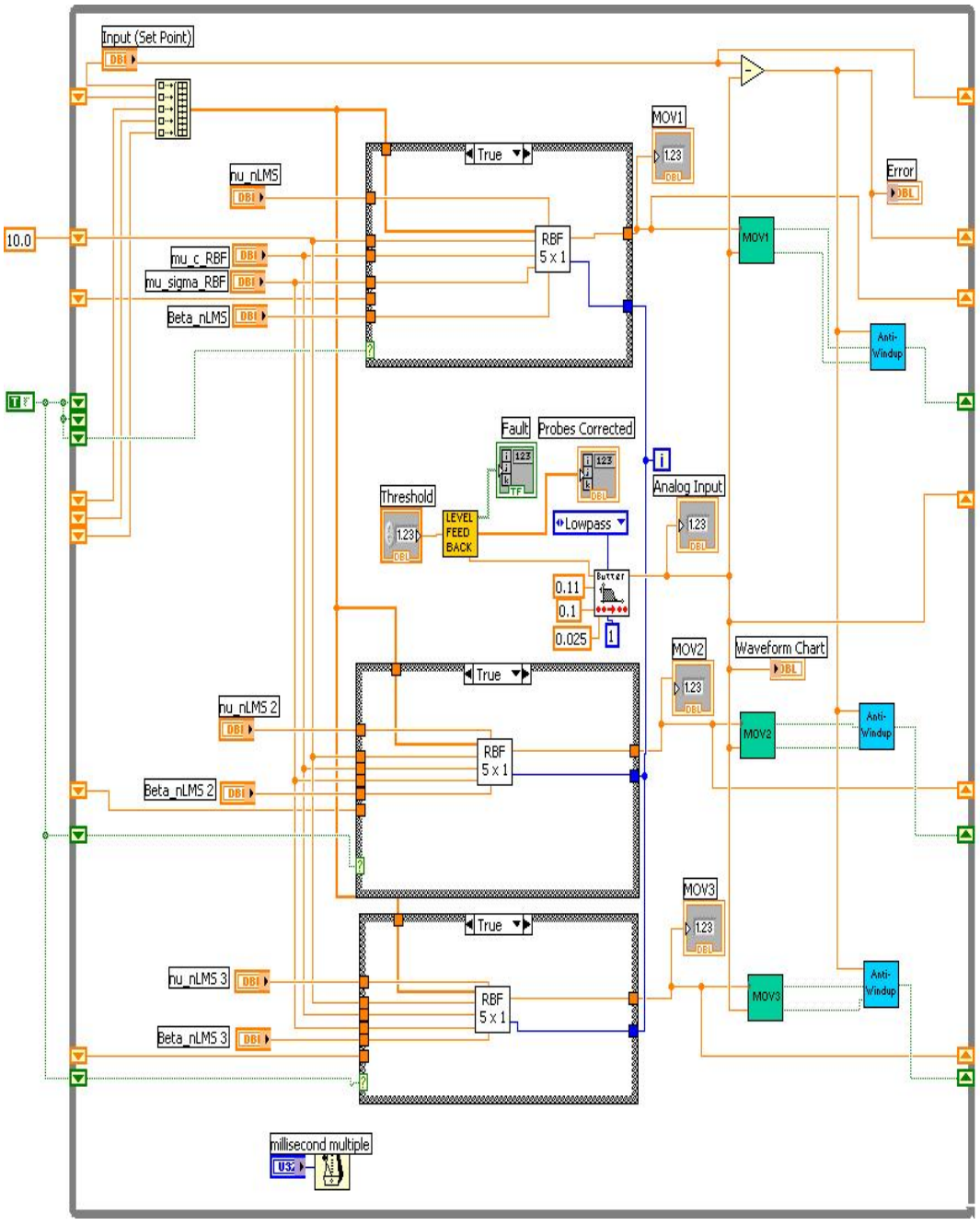


Figure 4.17: Main Program with Normal operation- Anti-windup Function (True)

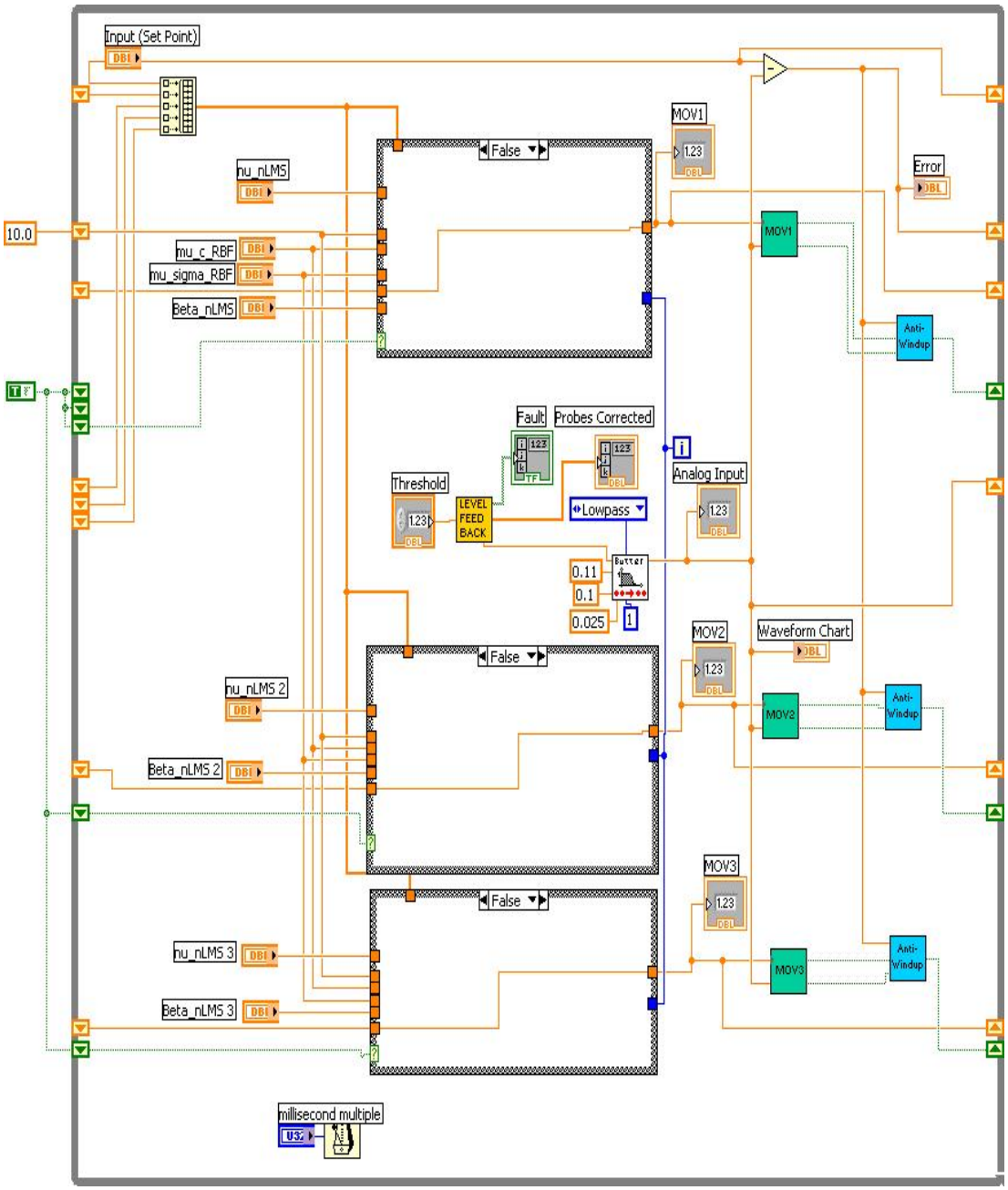


Figure 4.18: Main Program with MOV1 Saturated Anti-Windup Function (False)

4.5 Output Processing

Output processing is a process responsible for converting the output value from the controller to a value understood by the Motor Operated Valve. In our particular case, the output from the controller has an open range between the negative to the positive values. Therefore, the output from the controller is first limited to a value between 0 ~ 100.

In a conventional controller, if the process value increases, the output of the controller decreases like, for example, a temperature controller (if the temperature increases, the output to the heater decreases). This is called reverse action controller. Our case is different, if the axial level increase, the output needs also to increase to further open the extraction valve (MOV) to decrease the axial level. This is called a direct action controller. To change the action of the controller, the output from the controller should be reversed by applying the following equation. (Figure 4.19 shows the subprogram for the MOV):

Output= 10-X (4.7)

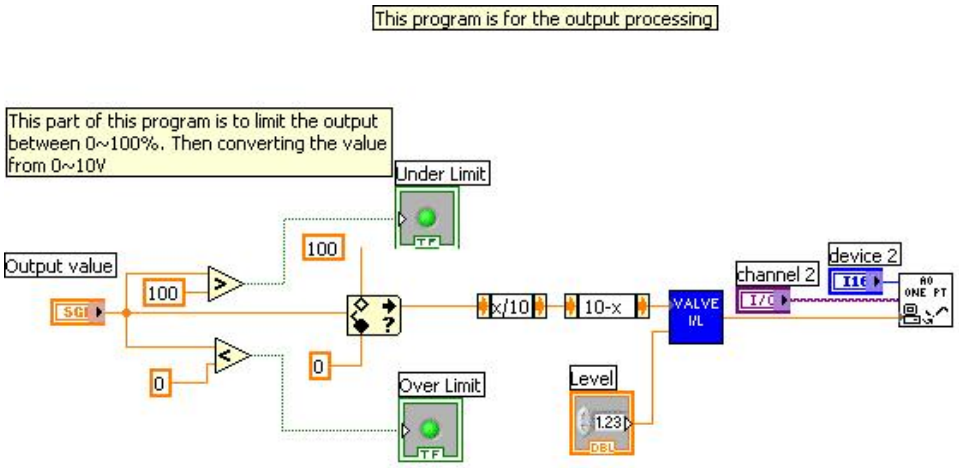


Figure 4.19: Output Processing: MOV Subprogram

Now, if there is a tuning problem in the ANN controller, the axial level will be out of control. If the axial level were to decrease below 0%, oil would drain through the extraction MOV. Similarly, if the axial level increases over the 100%, water would be carried over to a downstream location. In order to overcome such a problem, a subprogram is developed to interlock the operation of the MOV. The aim of the program is to monitor the status of the axial level and maintain its location in the operational range of 2~95%. In this case, if the level increases over 95% all three MOVs will open 100% to decrease the axial level to become within the operational range. On the other hand, if the level decreases below 2%, all three MOVs will close causing the level to increase to become within the operational range. Figure 4.20 shows the subprogram for the MOV interlock.

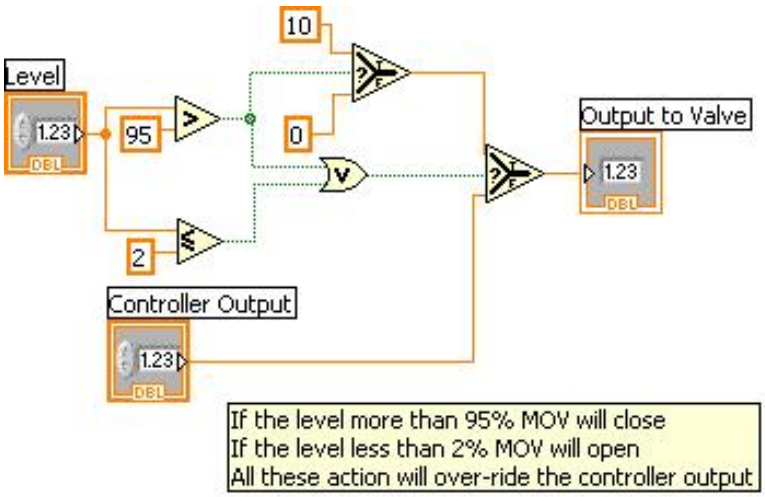


Figure 4.20: Output processing: MOV Interlock subprogram

4.6 Output Valves

Before we explain how the Motor Operated Valves operate, we shall stop a little bit to talk about the valves. The valves consist of two main parts: Body and Actuator. The body is the lower portion of the assembly, i.e., the ball valve, where fluid passes through, while the actuator is the upper portion of the assembly, i.e., a step-motor plus gears etc., which opens and closes the ball valve according to an input control signal [17].

The actuator may be operated manually like the conventional hand operated valves, or pneumatically, hydraulically, or by motor. The selection of the actuator depends on the type of the process forces, and the availability of power resources. In our case, Motor Operated Valves MOVs are chosen because of the availability of power sources. [17]

Those MOVs are used to control water extraction flow rate. Before deciding on the type of control valves to be used, we shall go through valves characteristics and types [17].

The main purpose of any control valve is to regulate flow through the valve by changing the position of the valve's plug via a proper signal. This signal will define the valve opening and consequently the flow through it. The governing equation for the control valve is given by [17]:

$$f = C_v . f(x) \sqrt{\frac{\Delta p_v}{G_f}} \qquad (4.8)$$

where:

f is the volumetric flow through valve

C_v is the valve capacity factor in $m^3 / \sqrt{Bar}$

f(x) is the inherent characteristic function of the valve

x is the valve opening position (0~100%)

Δp_v is the pressure drop across the valve

G_f is the specific gravity

The valve inherent characteristic *f(x)* determines the capacity flow across the valve as the valve opens. Figure 4.21 shows the three typical control valve inherent characteristics. **Quick Open** valves will allow more flow at small openings. The typical application is in the pressure relief processes where it requires very fast action to relieve pressure. [17]

A small opening of the **Equal Percentage (=%)** valves leads to low flow through the valve. The typical application is in fast processes like flow control applications. [17]

Linear valves have a ratio of one to one between the valve opening and the valve capacity. The typical application is in level control or pressure control system.

To select the proper valve to use we have to study the rangeability of each type of valve. The valve rangeability can be defined by this equation:

$$Rangeability = (Flow\ at\ 95\%\ Lift)/(Flow\ at\ 5\%\ Lift) \quad (4.9)$$

By applying this formula to each type, the typical inherent rangeability of Quick Open valves is less than 3, which is poor. Linear valves rangeability, however, is 19. On the other hand, the Equal Percentage valves' rangeability is grater than 30. In practice, the Equal Percentage valves are built with rangeability between 50~100, which will enable fine regulation of flow across the control valve [17].

Based on the above, the valve that has the capability to control the opening of the water extraction port without causing oil conning into the extraction port should be chosen. Therefore, the best type to conform to this criterion is the equal percentage valve.

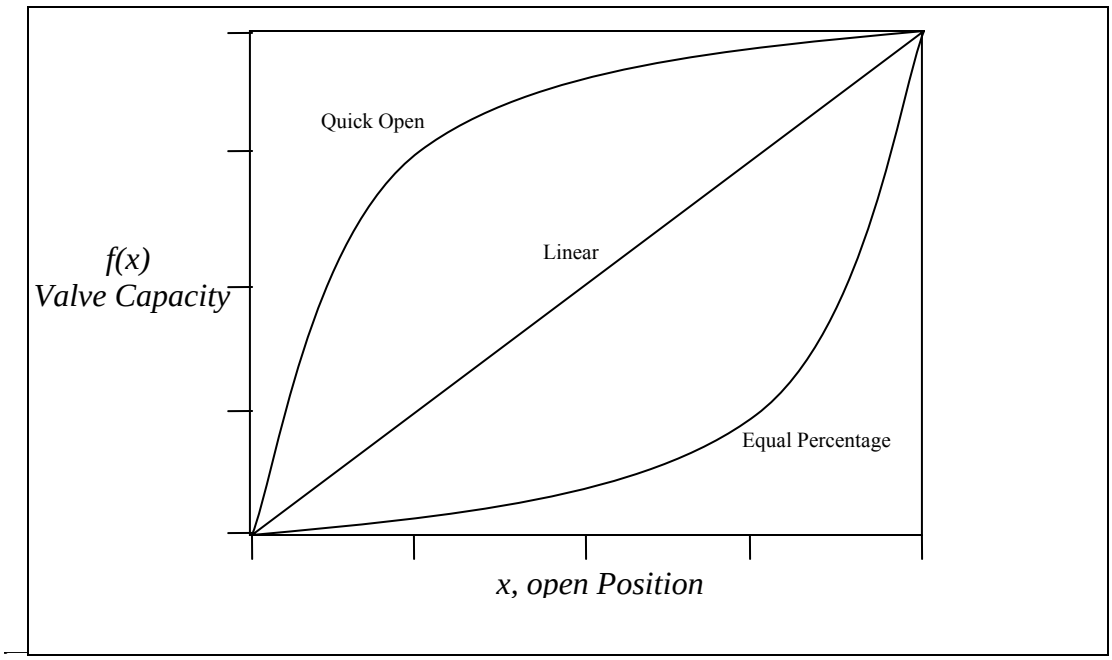


Figure 4.21: Inherent Valve Characteristics

Chapter 5

Operating Procedure and Result Discussion

5.1 Operation Procedure

This chapter will describe the methodology to be followed for operating the inline water/oil separator. The operator of the separator shall put in his consideration two parts: Physical Operation and Software Operation.

5.2 System Preparation

Before starting unit operation, some precautions shall be taken to assure the safety of the people involved and the working environment, as well as integrity of the system along with its components. These precautions cover the Mechanical, Electrical and Electronics parts of the unit.

From a mechanical point of view, the following checks must be made before starting the unit (Appendix 1 shows equipment list along with their models):

1. Verify that the water loop flow path is unimpeded by checking the valve sequencing:
    i. Water tank discharge valve is fully open.
    ii. Discharge valve from the water delivery pump is fully open.
    iii. Water tank delivery pump kick back (bypass valve) is open slightly (this valve is used to assure water flow across the pump in case of discharge line sudden closure, to regulate the flow, and to prevent cavitations of the pump impeller).
    iv. Water flow indicator discharge valve is fully open.
    v. All other valves in the water line are fully open.
    vi. The discharge isolation valves of the MOVs are open (the opening percentage depends on the total required water extraction rate).
    vii. The discharge valve of the Water Recovery pump is open. This loop assures that the main water tank is continuously filled.
2. Verify that the oil loop flow path is unimpeded by checking the valve sequencing:
    i. Oil tank discharge valve is fully open.
    ii. Discharge valve from the oil delivery pump is fully open.
    iii. Oil tank delivery pump kick back (bypass valve) is open slightly (this valve is used to assure water flow across the pump in case of discharge line sudden closure, to regulate the flow, and to prevent cavitations of the pump impeller).

iv. Oil filter valve is fully open (if filtering is required, otherwise it is closed).

v. Oil filter bypass valve is fully closed (if filtering is required, otherwise it is open).

vi. All other valves in the water line are fully open.

The second precaution is for the electrical safety:

1. Verify that the ON/OFF switches for the MOVs power are in the ON position.
2. Verify that the power is available to all pumps: Oil delivery Pump, Water Delivery Pump, and the Water Recovery Pump.
3. Verify that Water tank level switch is connected and operating well. (This switch is very important for auto-start of the Water Recovery Pump).
4. Verify that the motor speed controllers of the water and oil delivery pumps are connected and working properly.

The third checklist is for the electronic parts along with the software:

1. Verify the connections between the computer and the analog input and output Data Acquisition Boards. Appendix 1 shows the list of the electronic components of the data acquisition system.
2. Verify the physical integrity of the probes connections.
3. Verify the operation and the calibration of the MOVs according to the vendor standards and precautions. (Appendix 2)

4. Verify the operation and calibration of the signal conditioner cards (0~10 VDC to 4~20 mA converters) according to the vendor standards and precaution (Appendix 3)

5.3 System Start-up

The system is now ready to start. The first step is by starting the oil delivery pump. From this step we will observe the oil is circulating in the oil loop and by this we are ready to go the next step.

Similarly, the second step involves starting the water delivery pump and the water recovery pump. The water flowrate should not exceed 50% of the oil flowrate, due to flow loop limitations. Oil and water will be mixed and flow concurrently in a downhill, horizontal and uphill sections of the inline separator, respectively. Therefore, a stratified flow of water and oil is generated and will advance in uphill section of the pipe.

5.4 Software Operation

Before explaining unit operation, a brief description will be given for the components of the software Graphical User Interface GUI.

The program is developed using the LABVIEW 7.0 environment and all virtual instruments developed for this program are stored in the following program:

"Application_ANN_RBF_for_5X2_antiwindup_individual control_rev1.vi". The interface of this program is composed of input fields, output fields, and information fields as shown in Figure 5.1.

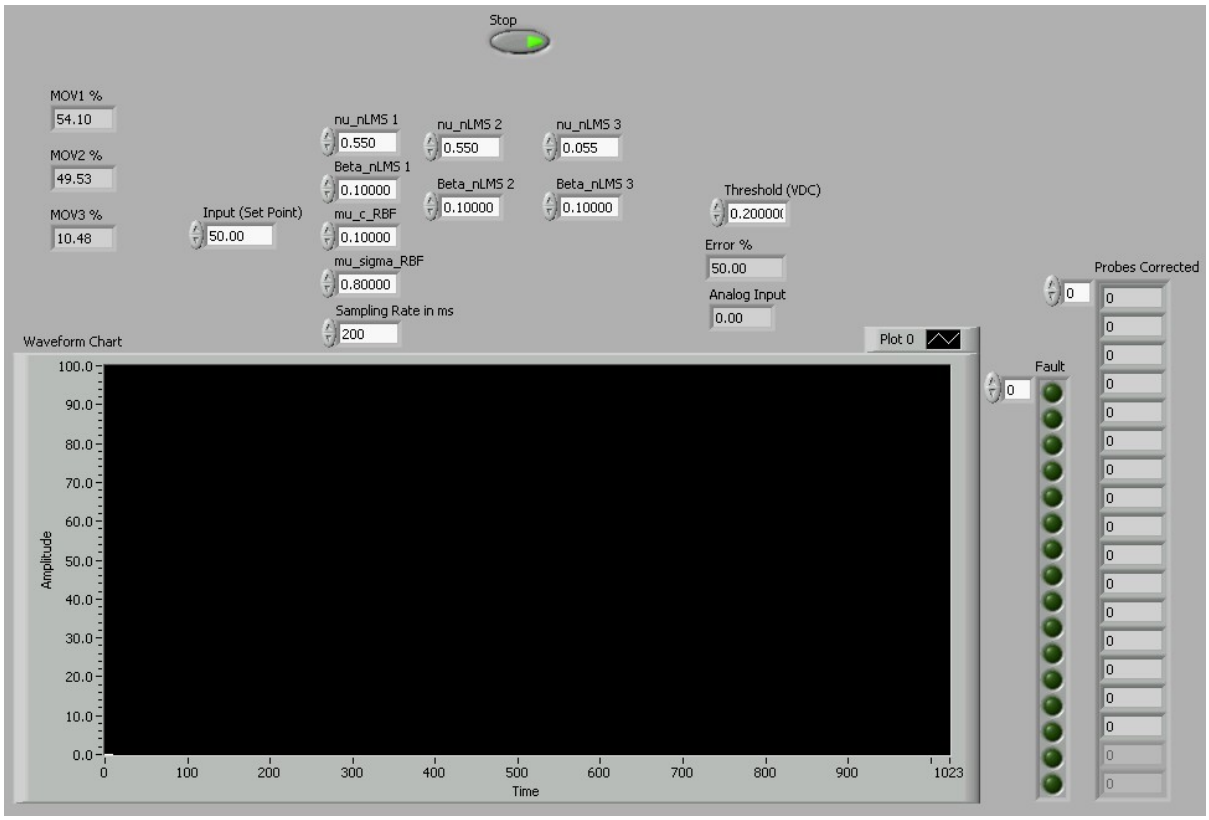


Figure 5.1. Capture of the screen of the Graphical User Interface GUI.

5.4.1 Graphical User Interface signals

A. The input field consists of the following:

1. Stop: To stop the program from running
2. Input (Set Point): This field is to set the desired normal operational level in the pipe.
3. Threshold: This field is to set the desired threshold for the comparator calculation in the signal conditioning process.

4. Sampling Time: The period of time where the sample of the continuous signals been sampled. In this case the sampling time is the time period between two successive executions of the program.
5. Tuning sets for MOV1, MOV2, and MOV3
    i. nu_nLMS 1: The training rate for the output layer MOV-1
    ii. Beta_nLMS 1: The regularization factor for the output layer
    iii. nu_nLMS2: The training rate for the output layer MOV-2
    iv. Beta_nLMS 2: The regularization factor for the output layer
    v. nu_nLMS3: The training rate for the output layer MOV-3
    vi. Beta_nLMS3: The regularization factor for the output layer
    vii. mu_c_RBF": The training rate for the center at the hidden layer
    viii. mu_sigma_RBF: The training rate for the width of the hidden layer

B. The output field consists of the following:

1. MOV-1: The percentage command going to the MOV-1 from the control system (0~100%). 0% being fully open and 100% being fully closed.
2. MOV-2: The percentage command going to the MOV-2 from the control system (0~100%). 0% being fully open and 100% being fully closed.
3. MOV-3: The percentage command going to the MOV-3 from the control system (0~100%). 0% being fully open and 100% being fully closed.

C. The information fields consists of the following:

1. Error: This is the difference between the set point and the axial level position in percentage.
2. Axial Level: The axial level of water in the measured range (0~100%).

3. Fault: A series of LEDs displaying faulty probes.
4. Probe Corrected: A series of digitized values (0 or 1) after the fault been corrected by the fault detector algorithm.
5. Waveform Chart: A chart that shows graphically the trend of the axial level movement in order to evaluate the performance of the control system.

5.4.2 Software Settings and results

Now, the unit is ready to start from the software side by just starting the software:" Application_ANN_RBF_for_5X2_antiwindup_individual control_rev1.vi" The main data entered into the software are indicated in Table 5.1.

Table 5.1: Software Parameter Setting for the Water/ Oil Separation Program

S/n	Input	Input description	Value
1	Input (Set point)	The target Axial level in Percentage	50 %
2	nu_nLMS 1	Training Rate for output Layer of MOV1 (0~1)	0.550
3	Beta_nLMS 1	Regularization Factor for the output Layer ($0<\beta<2$) of MOV1	0.100
4	Nu_nLMS 2	Training Rate for output Layer of MOV 2 (0~1)	0.550
5	Beta_nLMS 2	Regularization Factor for the output Layer ($0<\beta<2$) of MOV2	0.100
6	Nu_nLMS 3	Training Rate for output Layer of	0.055

		MOV3 (0~1)	
7	Beta_nLMS 3	Regularization Factor for the output Layer (0<β<2) of MOV 3	0.055
8	mu_c_RBF	Training rate for the center of hidden layer (RBF units). This value is set for all hidden layers of MOV 1, MOV 2, and MOV 3.	0.100
9	mu_sigma_RBF	Training rate for the width of the Gaussian of hidden Layer (RBF units). This value is set for all hidden layers of MOV1, MOV2, and MOV3.	0.800
10	Threshold (VDC)	to set the desire threshold for the comparator calculation in the signal conditioning process	0.200 VDC
11	Sampling Rate	The period of time where the sample of the continuous signals been sampled. In this case the sampling time is the time period between two successive executions of the program.	200 mSec

The results can be demonstrated as in the following Snapshoot figures. Figure 5.2 shows the start-up of the in line water-oil separator. In this case, the process been stabilized within 160 sec.

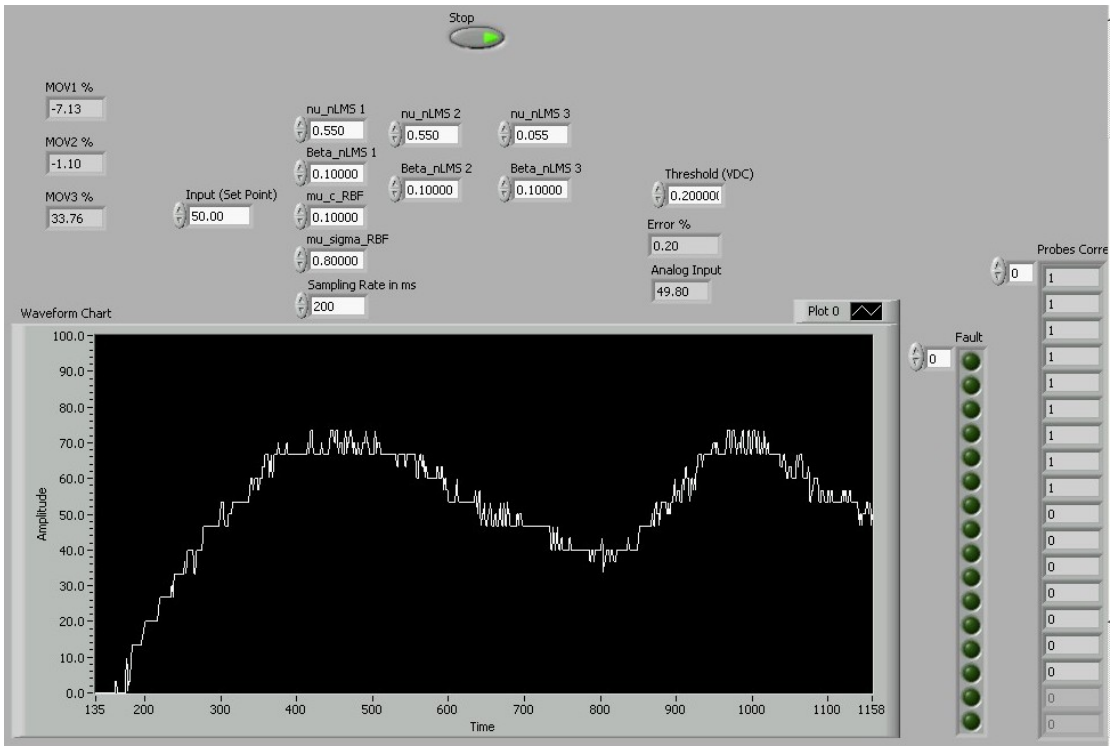


Figure 5.2: Trend shows the Start-up of the Software

In Figure 5.3: shows the stabilized controlling of the level. It is shows the variation between the 70% and 40%. This means that the variation is between +/- 2 probes. This achieved by keeping the MOV-1 and MOV-3 isolation valves open to a certain position while MOV-2 isolation valve is in crack open. The maximum allowable opening of the isolation valve is determined by observing the oil bubbles are not extracted from the MOVs at their maximum opening.

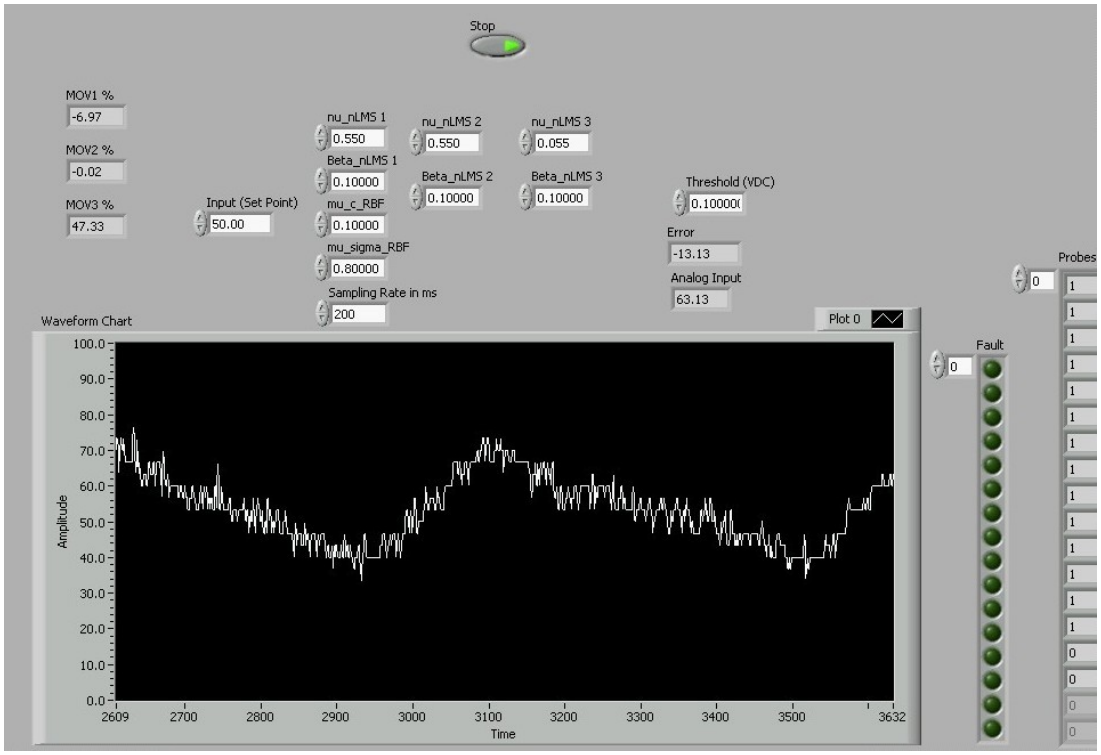


Figure 5.3: Stabilized axial level control

The next figure demonstrates a transient in which the water flow rate was suddenly stopped (Figure 5.4). In this case, the valves were fully closed. When water flowrate is stopped and the level goes below zero, the interlock program reacts and forces the valves to fully close.

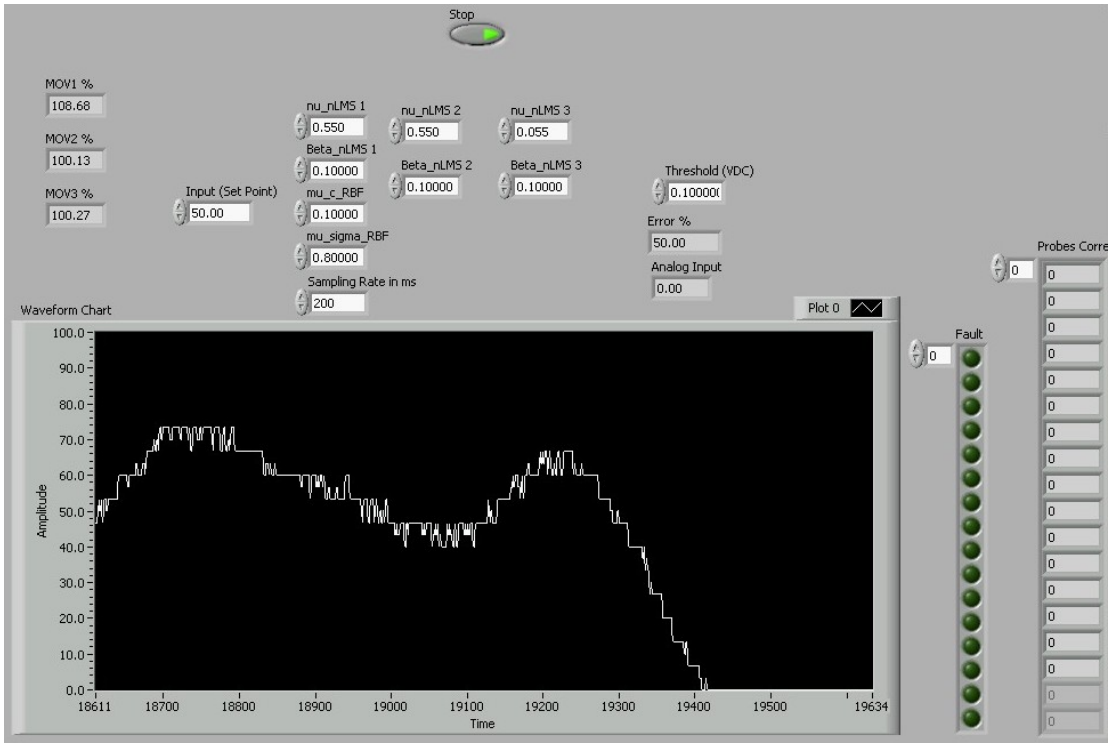


Figure 5.4: Sudden Stoppage of water. The valves in this case is on fully close

When water flow resumes back the valves will stay in their closed condition till the water axial level moves past the set point. This action is due to the anti-wind up controller. This is to assure that there will be no further output signal to close the valve while the actual axial level is below the set point. Figure 5.5 shows the transient of the water axial level when the water resumes flowing back through the system and the controller takes its action. It took 186 seconds to stabilize.

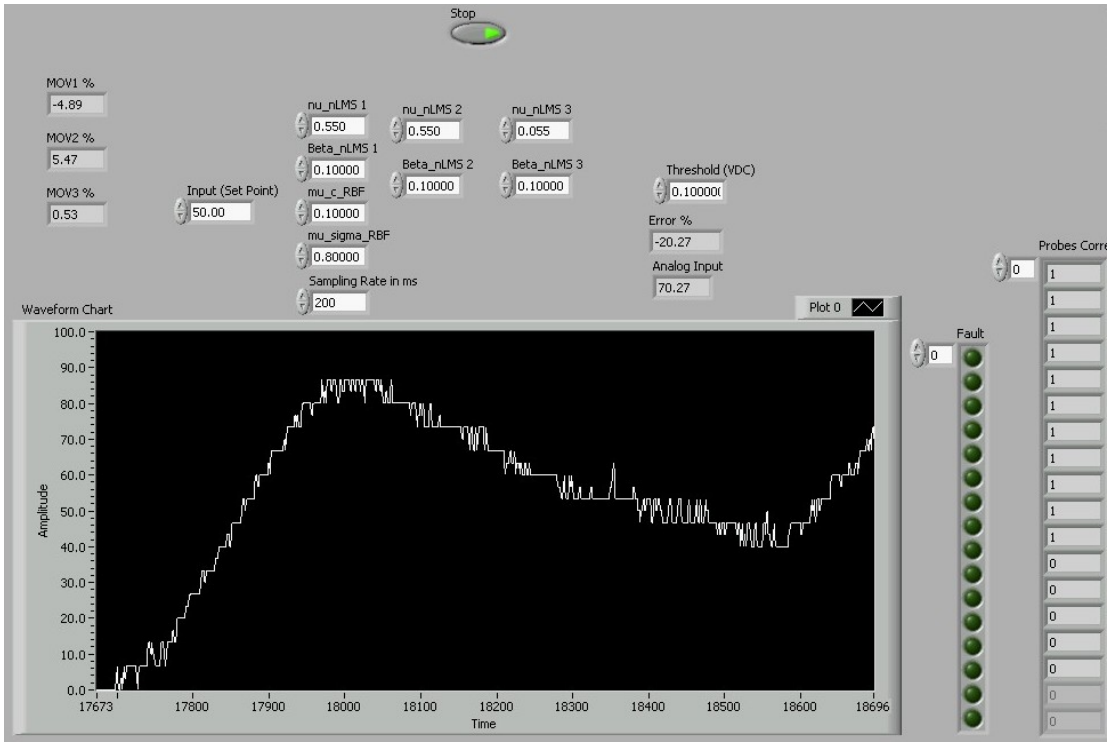


Figure 5.5: Water resuming back to the system and it took 186 seconds to stabilize the axial level

Figure 5.6 shows that the cycle of variation is relatively slow. This is a characteristic of the system dictated by the time it takes for the water to drain under gravity from the water layer. In Figure 5.6, however, the variation was artificially accelerated because MOV2 isolation valve was opened more. This case is considered as under-fitting case that causes this variation. The cycle is approximately 60 Sec and the variation is between 30% and 75% (approximately +/-7 probes around the set point). However, this variation is still within the acceptable operational limit (2~95%).

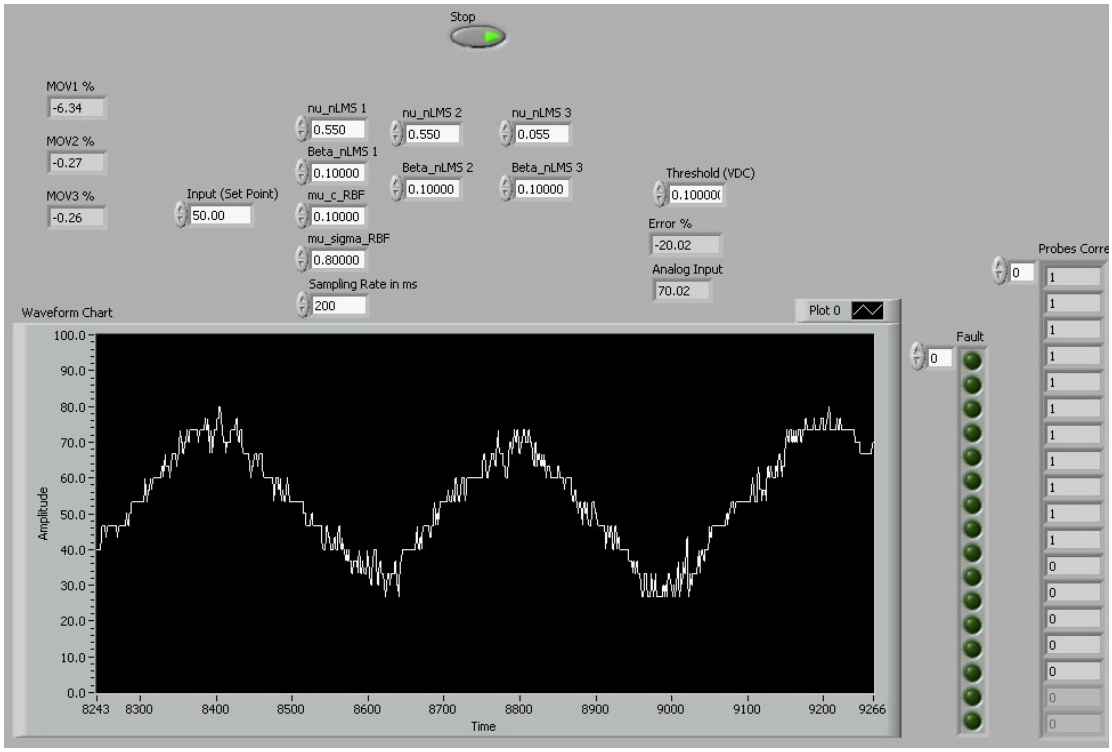


Figure 5.6: High variations with lower cycle (under fitting case)

Figures 5.7, 5.8, and 5.9, show the transient case of the valves failing to open due to a power failure and then opening following power resumption. In this case, the axial level advanced over the 100 %. When the power resumed back, the interlock program immediately opened all MOVs fully. As the level retreated back to the normal operating range (i.e., 0~100%), the normal control routine takes over. It took about 220 Seconds for the control system to stabilize the process.

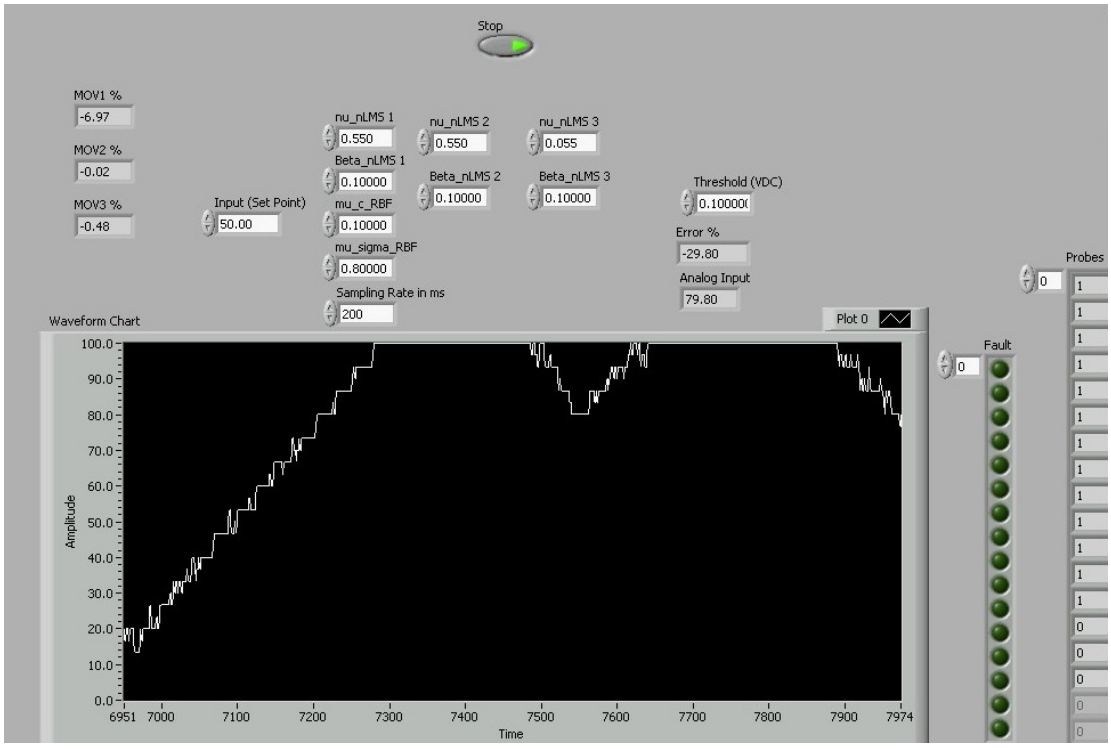


Figure 5.7: Power failure and the effect on the axial level

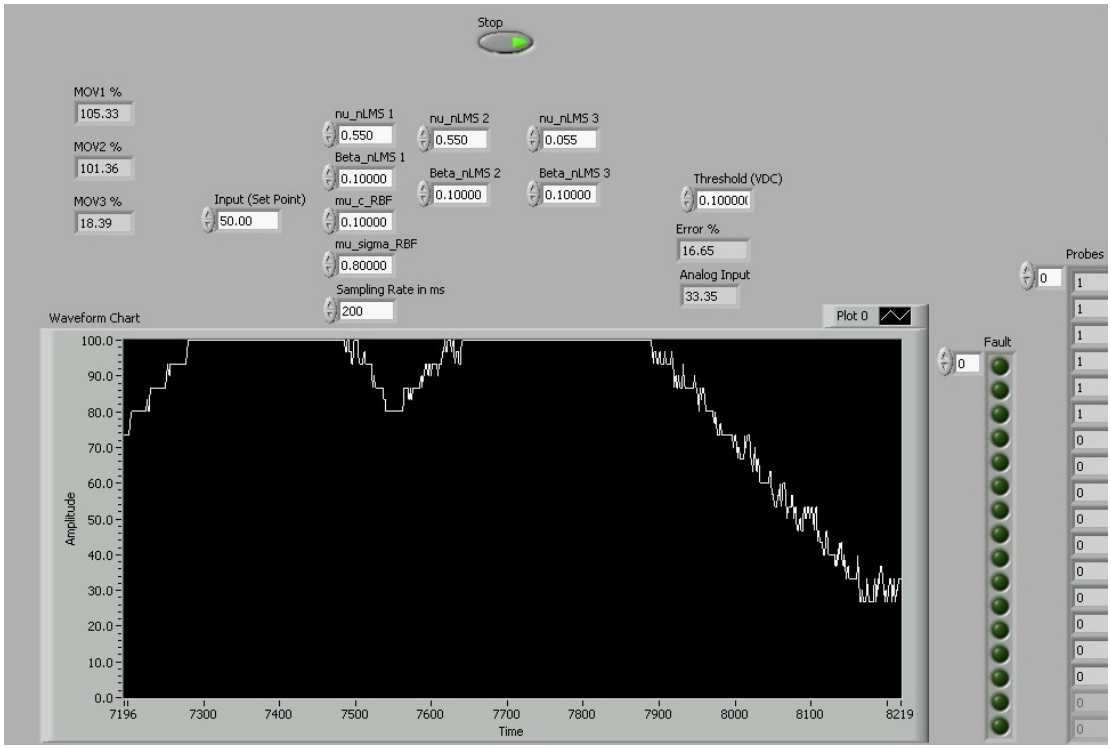


Figure 5.8: Power resuming back and the unit starts to control

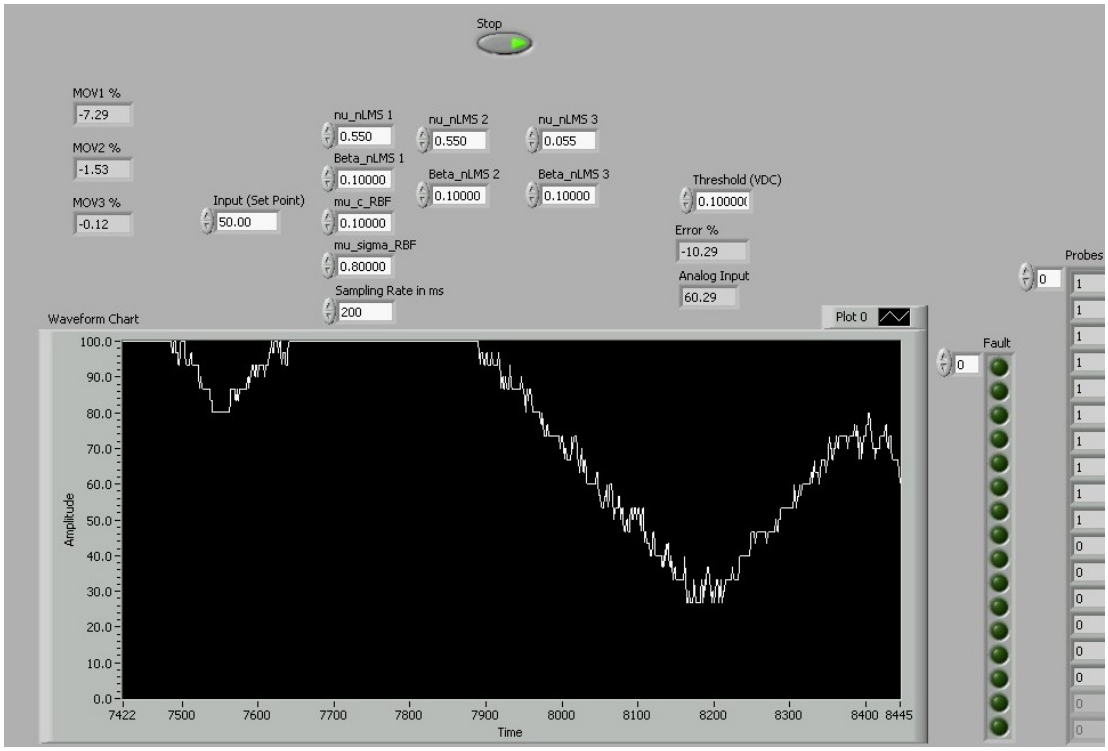


Figure 5.9: Controlling of the water took about 220 Seconds

A probe fault is simulated in Figure 5.10. The water level is shown to be at 63.6% while probe #1 failed. In this case the axial level does not change. Figure 5.11 shows two probes that are not adjacent, e.g., Probes #2 and #4, failing. Here the level is kept unchanged at 63.6%. Finally, Figure 5.12 shows probes 2, 4, and 5 failures. In this case, probe# 3 makes the correction in probe# 2, while the successive probes 4 & 5 makes the correction of oil to probe 4. Therefore, water is assumed to stop at Probe #3 and the level is 13.3%.

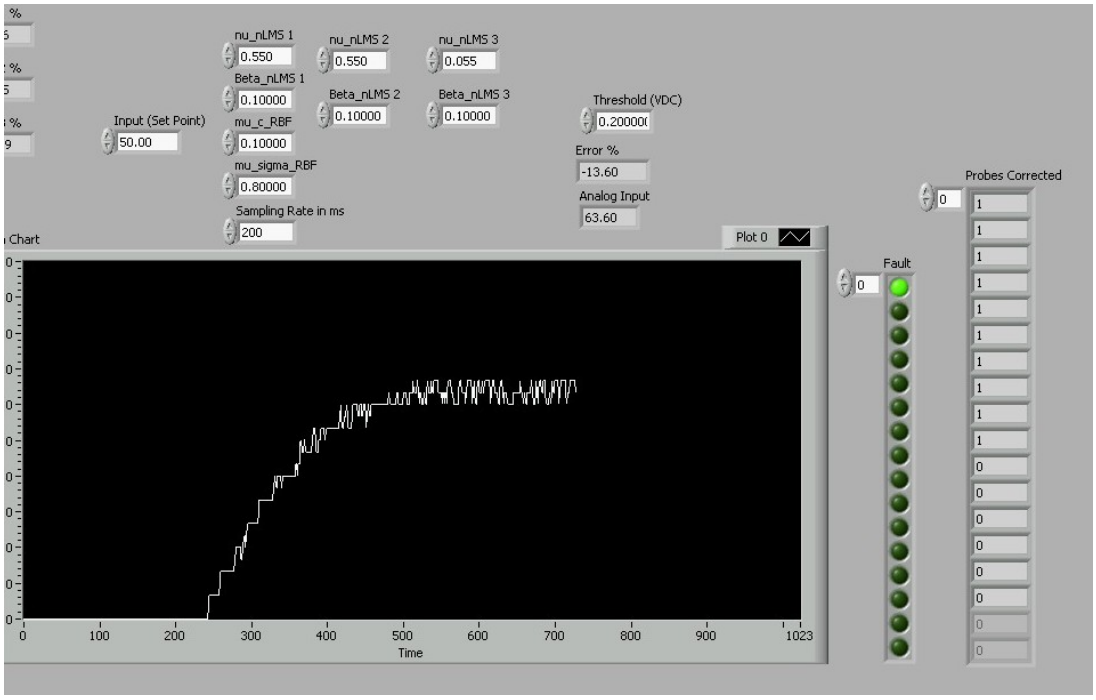


Figure 5.10: One probe failure (Probe#1) does not effect the measurement

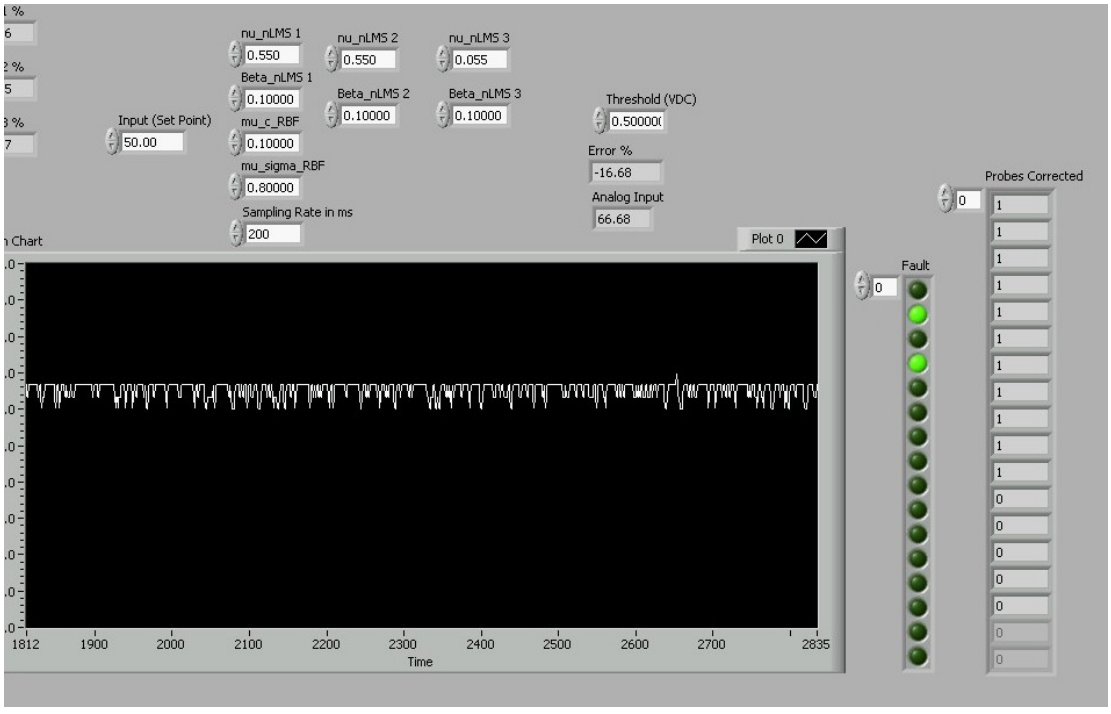


Figure 5.11: Two alternative probes (Probes 2, 4) failure does not affect the reading

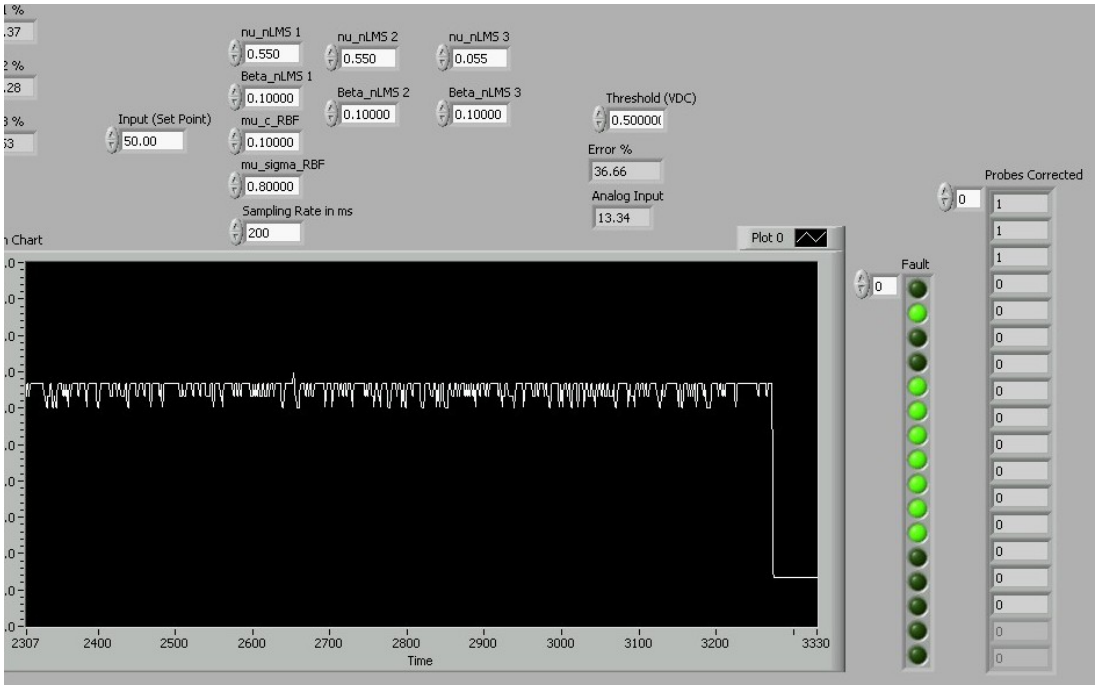


Figure 5.12: Probe#2 and two successive probes (4, 5) Failure causes the reading to change (As real water low). The water will be assumed covering Probe3.

Video recording has been used to document the performance of the controller. In the Video-1 a demonstration for the operation of the control system is presented. It shows the stabilized operation of the controller with brief description of the effect of the anti-wind up controller. Video-2 shows another demonstration about the GUI in the LABVIEW. It shows the three parts of the GUI: Input Section, Output Section and the Information Section. The demonstration show the constants used for the input section and their link with the control system block. The Information section is been elaborated for the description of the ISOWS operation. In addition, Video-3 shows another demonstration of the stabilized operation. Video-4 demonstrates the controller action when water sopped and the water retreating below 2%. The Valves closed fully and the water stopped from retreating. When water resumes back the controller takes action after the anti-windup reseated when level crossed the set point. Moreover, valve shutoff is

demonstrated in Video-5, when the axial water tail reaches below 2%. It is shows that the valve closes immediately regardless of the controller output.

CHAPTER 6

CONCLUSION

6.1 Conclusion

The objective of this thesis is to develop a software program for a practical application designed by Schlumberger. The new Inline Water/ Oil Separator designed by SLB is an innovative as most of the water/oil separators used are bulky tank separators. The new system is very compact and can be placed at or near the wellhead, since the separation process takes place while the production stream is flowing in the pipeline. The separation and extraction is made possible by allowing water and oil to stratify inside a near-horizontal pipeline. By extracting water from a location downstream of the lowest point (either absolute or local) in the pipeline, the water layer that extends from the water-accumulated sump will change its length according to a balance between the water influx and the extraction rates. The movement of the sump tail (also called the water layer front) is what makes the system amenable to control.

The software is developed to control the axial movement of the tail by putting it in a controllable range. The software is developed by breaking the problems into blocks.

The first block is the input block where it consists of the probes and circuitry for the data acquisition. The second block is the input processing where the analog inputs measured from the probes are checked and converted (conditioned) to a signal acceptable by the control system.

The third block is the control system, which is also broken further into two sub-blocks: ANN controller and Anti-windup controller. The ANN is used as the core program for the control system by using its power of identification to identify the inverse of the non-linear system, which is the inline water/oil separator system in this case. The Anti-wind up program is very essential in our case as the system is using MOVs for the control of water extraction. The saturation of the valves either opening or closing can cause problems in the control system when the deviation is below or greater than the set point, which can cause the controller output to cross the saturation region and thus resulting in the ANN controller to not act when the level is returning to the MOV controllable range. The aim of the Anti-wind up controller is to identify the saturation regions and to stop the inverse ANN controller from acting further.

Following the control system block is the output-processing block. This block consists of two sub-programs: Output conditioning and interlock programs. The output conditioning is the block responsible for reconditioning the signal generated from the control system to a signal acceptable by the MOVs. In addition, the interlock program is responsible for monitor the axial water level in the pipeline. So, if the axial level is advancing more than 100% the MOVs will be forced to open regardless of the controller

output, and if the axial level is retreating below 0% the MOVs will be forced to close regardless of the controller output. When the water axial level returns to the operating range, the controller output will pass to the MOVs.

Finally, the output valve block consists of the output data acquisition and signal converter where it will convert the signals generated to a signal acceptable by the MOVs.

Furthermore, the operation procedure and the result of operating the ANN controller for the ISOWS are also demonstrated. Different operation scenarios are performed to show the effectiveness of the control system. The start-up scenario shows that axial water level resumes to the stabilized operation with small shooting. The stabilized operation shows that the variation within probes is very minor. In addition to that, the response of the control system during the transients was tested. During the sudden stoppage of water, the axial level is retreating back to the zero percentage. In this case the MOVs are close regardless of the controller signal. In the other hand, when the water resume back to the normal operating range, the controller takes care to resume the axial water tail to the stabilized condition. Similarly, power failure to the MOVs been simulated in order to demonstrate the performance of the controller when the axial level accedes the instrumental section. When power resumes back all the MOVs opened fully regardless of the ANN controller sending. However, when the water returns to the operating range, the controller immediately takes its action to resume the axial water tail to the stabilized condition.

All the tests and scenarios were documented by snap shot trends of the axial level and the video recording.

NOMENCLATURE

Abbreviations

ISOWS	Inline Surface Oil/Water Separator
WLRs	water-liquid-ratios
ANN	Artificial Neural Network
SDCR	Schlumberger Dhahran Carbonate Research
MOVs	Motor-Operated-Valves
PV	Process Variable
SP	Set-Point Value
BDNs	Binary Decision units
LMS	Least Mean Square
SLS	Sequential Least Square
RLS	Recursive Least Square
NLMS	Normalized Least Mean Square
MLP	Multilayer Perceptron
RBFN	Radial Basis Function Networks
RBF	Radial Basis Function
SSE	Sum Square Error
GUI	Graphical User Interface

English Symbols

W	Weights Matrix

w_i	Wight element
R	Instantaneous Auto-Covariance
p	Instantaneous is the Cross-Covariance
x(n)	Input Vector at iteration n
$e(n)$	Error between the target and the calculated value at iteration n
$\boldsymbol{x}_p$	Training vector p of input vector $\boldsymbol{x}$
$\boldsymbol{y}_p$	Training vector p of output vector $\boldsymbol{y}$
w_{ij}^h	Weight assigned from input number i towards neuron number j in the hidden layer
w_{ij}^o	Weight assigned from neuron number i towards the neuron number j of the output layer.
E_p	Sum Square Errors for all output layer neurons for training set number p.
net_{pj}^h	The input value for neuron j at the hidden layer for training set number p.
i_{pj}	The output value from neuron j at the hidden layer for training set number p.
$f_j^h(.)$	Activation function at neuron j of the hidden layer.
net_{pk}^o	The input value for neuron k at the output layer for training set number p.

o_{pk}	The output value from neuron *k* at the output layer for training set number *p*.
$f_k^o(.)$	Activation function at neuron *k* at the output layer for training set number *p*.
t_k	Center at k^{th} neuron in the hidden layer of the RBFN.
$\boldsymbol{C_k}$	Width vector of the k^{th} Radial Basis Function neuron
H	Hermitian transposition
K	The number of centers in RBFN
d_{max}	Maximum distance between the selected centers in fixed center training.
d_i	Desire output for training set *i*
$\mathbf{u}_i$	Input vector for training set *i*
nu_nLMS	Program constant stands for the training Rate for output Layer of MOV
Beta_nLMS	Program constant stands for the regularization Factor for the output Layer
mu_c_RBF	Program constant stands for the training rate for the center of hidden layer (RBF units)
mu_sigma_RBF	Program constant stands for the training rate for the width of the Gaussian of hidden Layer (RBF units).

Greek Symbols

η	Learning Rate

α	Regularization Factor
δ_{pk}	Error at output layer neuron unit *k* for training set number *p*.
δ_{pj}^{h}	Error at the hidden layer neuron *j* for training set number *p*.
$\varphi(\boldsymbol{u};\boldsymbol{t}_k)$	k^{th} radial basis function which calculate the distance between input vector $\boldsymbol{u}$ and its own center $\boldsymbol{t}_k$
σ_k^2	Width of the k^{th} Radial Basis Function neuron
Φ	Interpolation matrix
ε	Sum square error for RBFN
μ_C	Training rate for the Centers of RBF neuron calculation
μ_σ	Training rate for the width RBF neuron calculation.

APPENDICES

Appendix A

The list of equipments used for this project and explained in chapter 2 are:

1. Oil Delivery & Recovery System
    a. Pumps
    b. Storage Tank
    c. Seperator Tank
    d. Piping
    e. Manual Valves
2. Water Delivery & Recovery System
    a. Pumps
    b. Storage Tanks
    c. Piping
    d. Manual Valves
3. Test Section “Seperator”
    a. Water Extraction Uphill Section
        i. Three Extraction Ports
        ii. Manual Valves
    b. Instrumented Uphill Section
    c. Downhill section
    d. Horizontal Section
4. Control System
    a. MOVs

b. Signal Converters

c. Conductivity Probes

d. Personal Computer having LABVIEW VER. 7.0

e. Electronic equipment and control:

   i. Data Acquisition for Analog Input: PCI-6071E, and SCB-68 manufactured by National Instrument Inc.

   ii. Data Acquistion for Analog Output: PCI-6713, and CB-68LP manufactured by National Instrument Inc.

   iii. Signal Converter 1~10VDC to 4~20mADC , model# PXU-20, manufactured by BRODERSEN controls.

   iv. Motor Operated Valves, Model# QA1-POS-XMIT, manufactured by CHEMLINE Plastics Limited.

Appendix B

Converter Manual (1~10VDC to 4~20mAVDC)

PXU-20

Process Signal Converter

Fig. 1

Input	Output
0-5V DC 0-10V DC 0-10 - +10VDC 0-20mA DC 4-20mA DC	4-20mA DC 0-20mA DC 0-10V DC 2-10V DC

Block Diagram

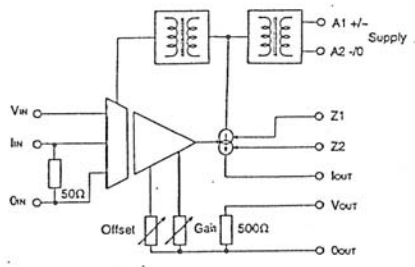


Fig. 2. Mounting

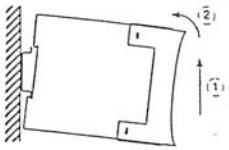
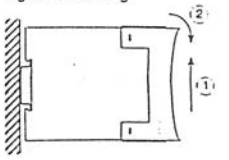


Fig. 3. Demounting



UK

User Instruction

Process signal converter for conversion of one type of analogue process signal to another and/or to provide galvanic isolation. Galvanic isolation between supply, input and output. (Fig. 1)

Mounting/Wiring:

See fig. 2, fig. 3 and wiring diagram.

Configuration

Remove cover by use of a screwdriver - fig. 4. Choose input range via selector switch. Output is selected by using the specific terminals - see wiring diagram. If wanted the offset and the gain of the output signal can be adjusted ± 5% by using the potentiometers - fig. 5 and 6.

Screw Terminals

Screw terminals with dual compartment. Up to 2 x 2,5mm² wire (2 x 1,5mm² with ferrules). Recommended torque 0,5Nm, max. 0,7Nm.

Technical Data

Input:

Input signal	Impedance	
0-5V DC	100kOhm	U_{max} = 50Vp-p
0-10V DC	100kOhm	
-10-+10V DC	100kOhm	
0-20mA DC	50 Ohm	I_{max} = 50mA
4-20mA DC	50 Ohm	

Selected via switches on the front panel.
Upper critical frequency 30 Hz.

Output:

Voltage:	500Ohm via intern shunt
Current:	R_{max} 500Ohm
Accuracy:	< 1%, without adjustment
Linearity:	< 0,05% of full scale
Temp. Coefficient:	< 0,02%/°C
Ripple (RMS):	< 0,1%

Supply Voltage:

924M = 12-48V AC/DC	10,5- 60V AC/DC
924 = 24V DC	20,4-27,6V DC
230 = 110- 230V AC	95-265V AC
Consumption:	DC = 2W AC = 2W

Isolation:
Input/Output/Supply: 3,75kV AC 1min.

Indication: Green LED = Operation voltage
Yellow LED = Input less than 5-10% of FS.

Temperature:
Ambient Temp.: -20-+55°C.
Storage Temp.: -40-+80°C.

D

Bedienungsanleitung

Prozeßsignalkonverter mit linearer Kennlinie zur Umwandlung eines Prozeßsignals in ein Anderes. Galvanische Trennung zwischen Eingang, Versorgung und Ausgang. (Fig. 1)

Montage/Anschluß:

Siehe fig. 2, fig. 3 und Anschlußdiagramm.

Konfiguration

Entfernen Sie die Frontabdeckung mit einem Schraubendreher - fig. 4. Einstellung des Eingangsbereiches durch Bereichsschalter auf der Gerätefront. Die Auswahl des Ausgangssignals erfolgt über die Anschlußklemmen - siehe Anschlußdiagramm. Offset und Gain lassen sich im Bereich von ± 5% mit Potentiometern auf der Gerätefront einstellen. - fig. 5 und 6.

Klemmen

Kombischlitzschrauben 0,5-0,7Nm. Anschlußkabel bis 2 x 2,5mm² oder 2 x 1,5mm² mit Aderenhülsen.

Technische Daten

Eingang:

Eingangssignal	Impedance	
0-5V DC	100kOhm	U_{max} = 50Vp-p
0-10V DC	100kOhm	
-10-+10V DC	100kOhm	
0-20mA DC	50 Ohm	I_{max} = 50mA
4-20mA DC	50 Ohm	

Die Selektion erfolgt über Schalter an der Gerätefront.
Obere kritische Frequenz 30 Hz.

Ausgang:

Spannung:	500Ohm interner Shunt
Strom:	R_{max} 500Ohm
Genauigkeit:	< 1%, Ohne Einstellung
Linearität:	< 0,05% Skalenwert
Temp. Koeffizient:	< 0,02%/°C
Ripple (RMS):	< 0,1%

Versorgungsspannung:

924M = 12-48V AC/DC	10,5- 60V AC/DC
924 = 24V DC	20,4-27,6V DC
230 = 110- 230V AC	95-265V AC
Leistungsaufnahme:	DC = 2W AC = 2W

Isolation:
Eingang/Ausgang/
Versorgung: 3,75kV AC 1min.

Anzeigen: Grüne LED = Versorgungsspannung
Gelbe LED = Eingangssignal 5-10% des maximalen Signals.

Temperatur:
Umgebungstemp.: -20-+55°C.
Lagertemp.: -40-+80°C.

DK

Brugervejledning

Signalkonverter til konvertering af én type analog processignal til et andet, og/eller direkte til at opnå galvanisk adskillelse. Galvanisk adskillelse mellem indgang, udgang og driftsspænding. (Fig. 1)

Montage/Tilslutning

Se fig. 2, fig. 3 og tilslutningsdiagram.

Konfiguration

Fjern dæksel med skruetrækker - fig. 4. Vælg input område på omskifter. Udgangssignal vælges via terminalvalg - se tilslutningsdiagram. Hvis ønsket kan udgangssignalets offset og gain justeres ± 5% på de to potentiometre - fig. 5 og 6.

Skruterminaler

Skruterminaler med dobbelt kammer. Op til 2 x 2,5mm² ledning (2 x 1,5mm² med hylser). Anbefalet tilspændingsmoment 0,5Nm, max. 0,7Nm.

Tekniske data

Indgang:

Indgangssignal	Impedans	
0-5V DC	100kOhm	U_{max} = 50Vp-p
0-10V DC	100kOhm	
-10-+10V DC	100kOhm	
0-20mA DC	50 Ohm	I_{max} = 50mA
4-20mA DC	50 Ohm	

Udgang:

Udgangssignal:	500Ohm via intern shunt
Strøm:	R_{max} 500Ohm
Nøjagtighed:	< 1%, uden finjustering
Linearitet:	< 0,05% af fuld skala
Temp. koefficient:	< 0,02%/°C
Ripple (RMS):	< 0,1%

Driftsspænding:

924M = 12-48V AC/DC	10,5- 60V AC/DC
924 = 24V DC	20,4-27,6V DC
230 = 110- 230V AC	95-265V AC
Effekt forbrug:	DC = 2W AC = 2W

Isolation:
Indgang/udgang/
driftsspænding: 3,75kV AC 1min.

Indikatorer: Grøn LED = Driftsspænding
Gul LED = Input mindre end 5-10% af FS.

Temperatur:
Driftstemperatur.: -20-+55°C.
Lagertemperatur.: -40-+80°C.

BRODERSEN controls

Export:
Tel: +45 46 74 00 00
Fax: +45 46 75 73 36
E-mail: bc@brodersencontrols.com

United Kingdom:
Tel: +44 020 8546 4283
Fax: +44 020 8547 3628
E-mail: bcs@brodersen.co.uk

Germany:
Tel: +49 208 46954-0
Fax: +49 208 46954-50
E-mail: bs@brodersen.de

PXU-20

Process Signal Converter

Wiring Diagram

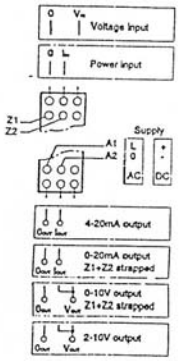


Fig. 4.

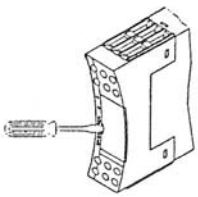


Fig. 5.

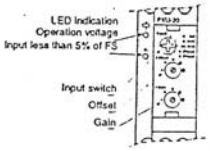
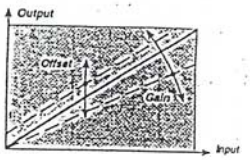


Fig. 6.



F

Manuel d'instructions

Le convertisseur de procédé transforme un signal analogique de procédé en un autre signal analogique isolé. L'isolation se fait entre alimentation, entrée, sortie. (Fig. 1)

Montage/Cablage

Voir fig. 2, fig. 3 et les schémas de raccordement.

Configuration

Enlever le couvercle avec un tournevis –fig.4. Choisir une échelle d'entrée par le commutateur. Choisir une sortie en prenant le cablage adequat. Voir schéma de raccordement.Il est possible d'ajuster le gain et l'offset su signal de sortie de ± 5% par les potentiomètres –fig. 5 et 6.

Borniers à vis

Les borniers possedent deux emplacements.Ils acceptent des fils jusqu'à 2 x 2.5mm² (2x1.5mm² avec embout).Un serrage de 0.5Nm,max.0.7Nm est ecommandé.

Caractéristiques

Entrée:

Signal d'entrée	Impédance	
0-5V DC	100kOhm	
0-10V DC	100kOhm	U_{max} = 50Vp-p
-10-+10V DC	100kOhm	
0-20mA DC	50 Ohm	I_{max} = 50mA
4-20mA DC	50 Ohm	

Selectionnable par switch en face avant Fréquence critique haute 30 Hz.

Sortie:

Tension:	500Ohm par shunt interne
Courant:	R_{max} 500Ohms
Précision:	< 1%, sans réglage
Linearité:	< 0,05% de la PE
Coeff. de temp.:	< 0,02%/°C
Ondulation (RMS):	< 0,1%

Alimentation générale:

924M = 12-48V AC/DC	10,5- 60V AC/DC
924 = 24V DC	20,4-27,6V DC
230 = 110- 230V AC	95-265V AC
Consommation:	DC = 2W AC = 2W

Isolation:

Entre entrée/sortie et alimentation:	3.75kV AC 1min.

Indication :	LED verte = Sous tension LED jaune= Entrée inférieur à 5-10%de la pleine échelle.

Temperatures:

Fonctionnement :	-20-+55°C.
Stockage :	-40-+80°C.

E

Instrucciones para el usuario

Convertidor de señal de proceso de un tipo de señal de proceso analógica a otra y/o para proporcionar aislamiento galvánico. Aislamiento galvánico entre la alimentación, la entrada y la salida. (Fig. 1)

Montaje/Cabelado:

Ver fig. 2, fig. 3 y esquema de conexionado.

Configuración

Quitar la tapa con un destornillador - fig. 4. Elegir el rango de entrada con el selector. La salida se seleciona usando las bornas específicas - ver esquema de conexionado. Si se quiere el ajuste fino y la ganancia de la señal de salida pueden ajustarse ± 5% usando los potenciómetros - fig. 5 y 6.

Bornas de tornillo

Bornas de tornillo con doble compartimento. Para cable de 2 x 2,5mm² (2x1,5mm² con bornas de conexión). El momento de torsión recomendado es de 0,5Nm, max. 0,7Nm.

Información técnica

Entrada

Señal de entrada	Impedancia	
0-5V DC	100kOhm	
0-10V DC	100kOhm	$U_{máxima}$ = 50Vp-p
-10-+10V DC	100kOhm	
0-20mA DC	50 Ohm	$I_{máxima}$ = 50mA
4-20mA DC	50 Ohm	

Se selecciona mediante los interruptores en el panel frontal.

Por encima de la frecuencia crítica 30Hz.

Salida:

Voltaje/Tensión:	500Ohm via shunt interno
Corriente:	R_{max} 500Ohm
Precisión:	< 1%, sin ajuste
Linealidad:	< 0,05% de fondo de escala
Coeficiente de temp.:	< 0,02%/°C
Factor de rizado (RMS):	< 0,1%

Voltaje de alimentación:

924M = 12-48V AC/DC	10,5- 60V AC/DC
924 = 24V DC	20,4-27,6V DC
230 = 110- 230V AC	95-265V AC
Consumo:	DC = 2W AC = 2W

Aislamiento:

Entrada/Salida/ Alimentación:	3,75kV AC 1min.

Indicación LED:	LED verde = Con tensión LED amarillo = Baja señal en la entrada del 5-10% del fondo de escala.

Temperatura:

Temp. ambiente:	-20-+55°C.
Temp. almacenaje:	-40-+80°C.

I

Instruzioni per l'uso

Il convertitore di segnali di processo per la conversione di un tipo di segnale analogico di processo in un altro, e/o per isolamento galvanico. Isolamento galvanico fra alimentazione, ingresso e uscita. (Fig. 1)

Montaggio/Cablaggio:

Vedere fig. 2, fig. 3 e lo schema elettrico.

Configurazione

Rimuovere la protezione servendosi di un cacciavite - fig. 4. Scegliere il campo d'ingresso tramite il selettore. Selezionare l'uscita per mezzo dei terminali specifici - vedere lo schema elettrico. E possibile regolare lo scostamento e il guadagno del segnale d'uscita di ± 5% per mezzo dei potenziometri - fig. 5 e 6.

Terminali a vite

Terminali a vite a camera doppia. Conduttore fino a 2 x 2,5mm² (2 x 1,5mm² con ghieretta). Coppia consigliata 0,5Nm, max. 0,7Nm.

Specifiche tecniche

Ingresso

Segnale di ingresso	Impedanza	
0-5V DC	100kOhm	
0-10V DC	100kOhm	U_{max} = 50Vp-p
-10-+10V DC	100kOhm	
0-20mA DC	50 Ohm	I_{max} = 50mA
4-20mA DC	50 Ohm	

Selezionato via interruttori sul pannello frontale
Frequenza critica superiore 30 Hz

Uscita:

Tensione	500Ohm mediante derivazione interna
Corrente	R_{max} 500Ohm
Precisione	< 1%, senza regolazione
Linearità	< 0,05% della scala completa
Coeffciente di temp.	< 0,02%/°C
Ondulazione (valore efficace)	< 0,1%

Tensione di alimentazione:

924M = 12-48V AC/DC	10,5- 60V AC/DC
924 = 24V DC	20,4-27,6V DC
230 = 110- 230V AC	95-265V AC
Assorbimento:	DC = 2W AC = 2W

Isolamento:

Ingresso/Uscita/ Alimentazione:	3,75kV AC 1min.

Indicatori:	LED verde = tensione d'esercizio LED giallo = Ingresso basso 5-10% dell'FS (slittamento di frequenza)

Temperatura:

Temp. ambiente:	-20-+55°C.
Temp. di conservazione:	-40-+80°C.

BRODERSEN controls

Export:
Tel: +45 46 74 00 00
Fax: +45 46 75 73 36
E-mail: bc@brodersencontrols.com

United Kingdom:
Tel: +44 020 8546 4283
Fax: +44 020 8547 3628
E-mail: bcs@brodersen.co.uk

Germany:
Tel: +49 208 46954-0
Fax: +49 208 46954-50
E-mail: ba@brodersen.de

Appendix C

Motor Operated Valve Manual

CHEMLINE Plastics Limited

Your Pipeline To Quality Valves, Strainers, Flowmeters and Controls

55 Guardsman Road
Thornhill, Ontario
L3T 6L2
Canada
Tel: 905-889-7890
Fax: 905-889-8553

INSTALLATION AND MAINTENANCE INSTRUCTIONS
For Chemline A- & Q-Series Actuator with Positioner

General:

The 4-20 mA positioner circuit board option is designed to provide precise control of actuator position through 90° of rotation or multiple 360-degree rotations for multi-turn valves.

Refer to attached drawings and proceed as follows:

Calibration:

Calibration:

Do not manually override the actuator out of its quadrant. If the actuator is out of quadrant, follow the directions below. All other settings should be correct provided that nothing else was touched.

Power actuator to the fully closed position by applying 120VAC to terminals 1 and 4 at location J1. Re-adjust closed limit micro switch now, if necessary. If the potentiometer does not read approximately 95 ohms between terminals 1 and 2 at location J1, or 500 ohms at midpoint for multi-turn, then adjust the ohm reading by loosening the 3 #8-32 set screws on the large gear on the actuator shaft and rotate it (and therefore the potentiometer gear) until the desired value is obtained, then retighten the 3 screws. Install 4-20 mA input signal to location J3.

Zero adjustment:

Apply a fully closed signal to the positioner's input equal to 4 mA. The actuator should cycle until the minimum signal position limit switch is tripped, stopping the actuator. Adjust the "zero" adjustment (drawing #0003PB) until the actuator stops in the fully closed position and the red indicator LED is off.

a) If the red LED is on, turn the "zero" adjustment clockwise until the LED just turns off.

b) If the red LED is off, turn the "zero" adjustment counter clockwise until the LED is on, then clockwise until the LED just turns off.

Calibration & Adjustment Instructions A- & Q- Series Actuator w/Positioner

Range adjustment:

Apply a fully open signal to the positioner's input equal to 20 mA. The actuator should cycle until the maximum signal position limit switch is tripped stopping the actuator. Adjust the "range" adjustment (drawing #0003PB) until the actuator stops in the fully open position and the green indicator LED is off.

a) If the green LED is on, turn the "range" adjustment clockwise until the LED just turns off.

b) If the green LED is off, turn the "range" adjustment counter clockwise until the LED is on, then clockwise until the LED just turns off.

NOTE: **It may be necessary to repeat steps A&B when calibrating the "zero" and "range" adjustment to ensure proper calibration.**

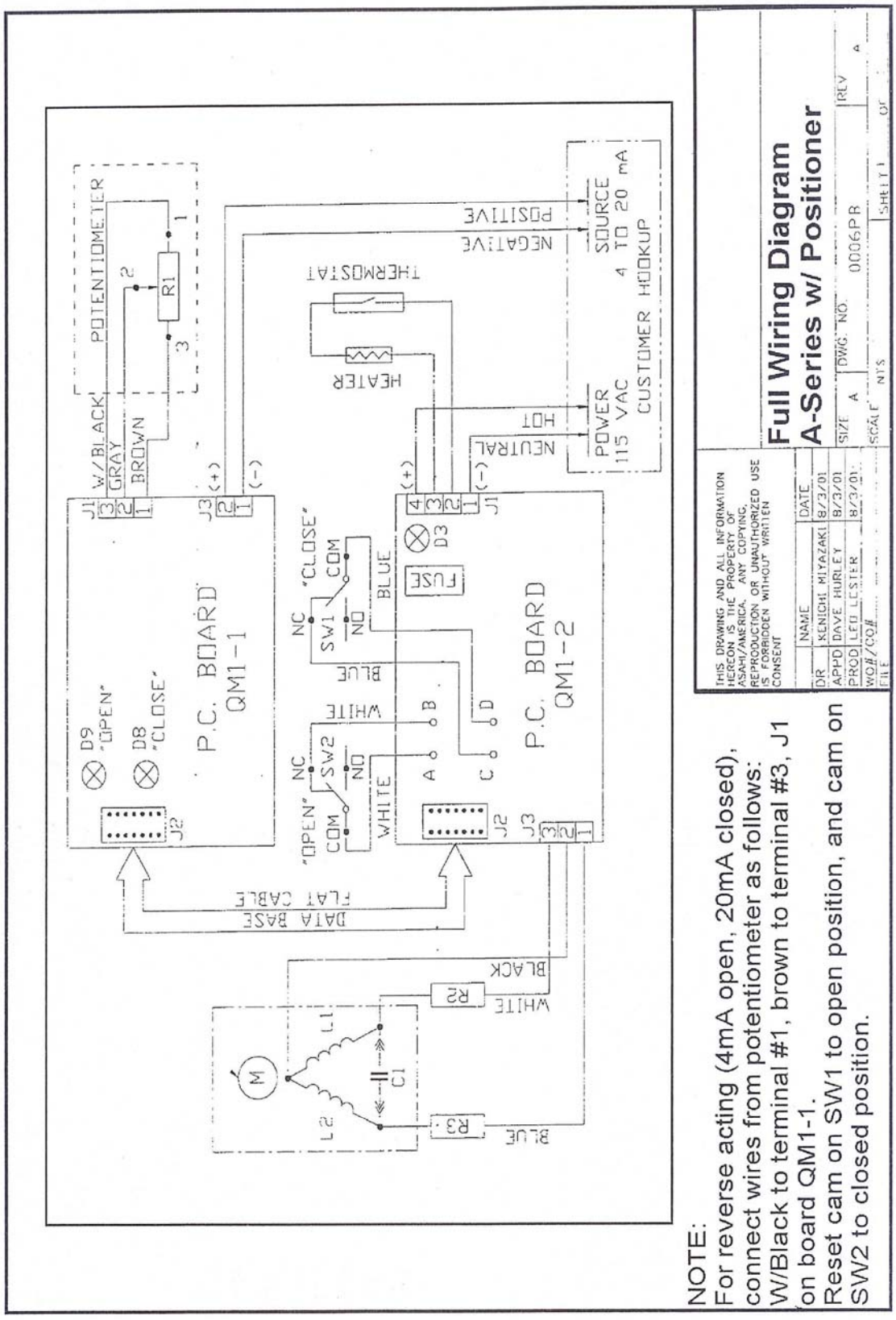
NOTE: **If unit is calibrated for 0 – 10 VDC input, adjustment procedure remains the same, only the control signal changes to 0 - 10 VDC.**

NOTE: **For reverse acting (20 mA closed, 4 mA open) factory modifications are required, however adjustment procedure remains the same.**

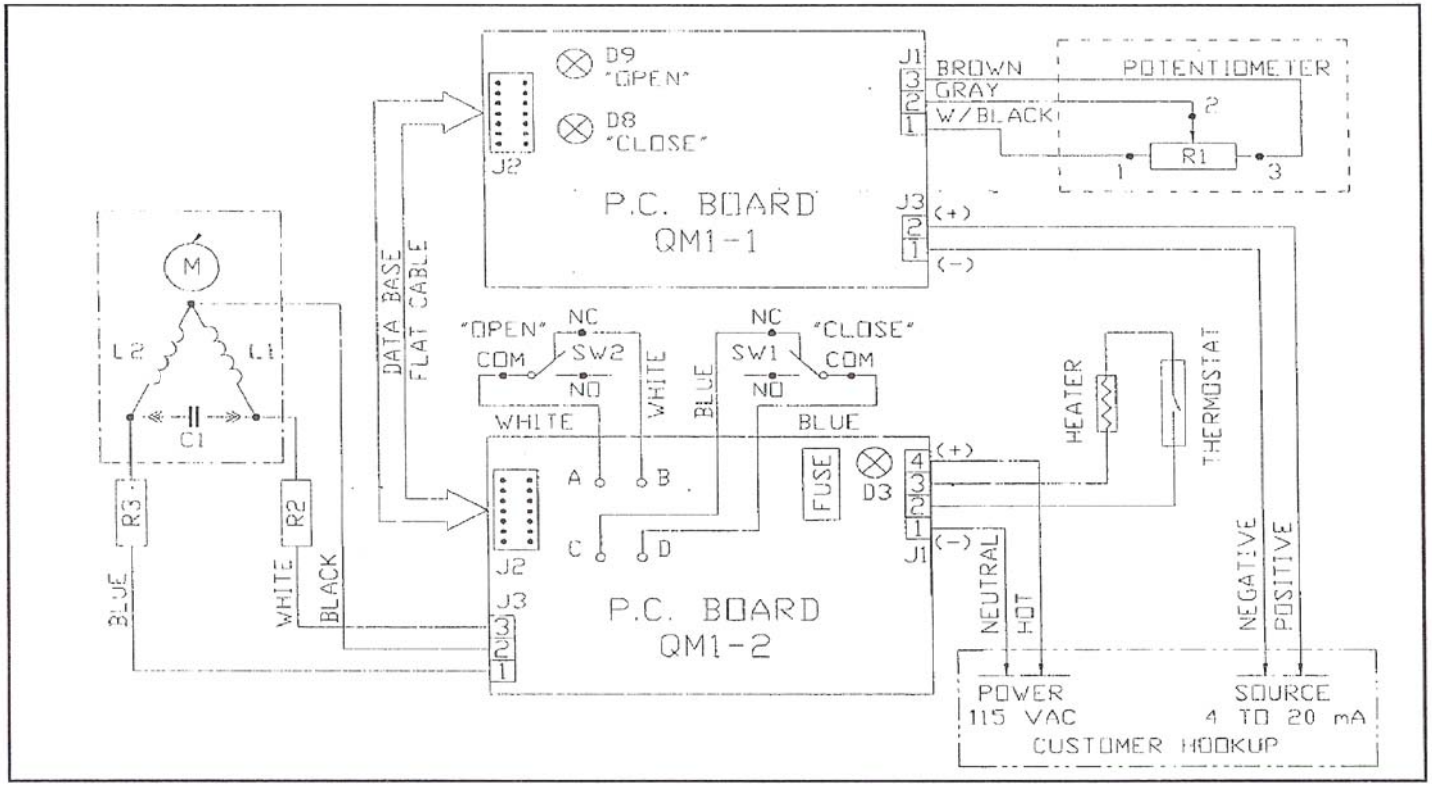
PERFORMANCE CHART Positioner Mounted in Typical Actuator	
Independent Linearity (The maximum deviation of the actual characteristic from a straight line).	0.5% of span
Resolution (Smallest possible change in valve position)	0.7% of span
Deadband (The maximum range through which the input signal can be varied without initiating a change in output shaft position. Adjustable via anti-hunt control).	0.5% min. of span
Hysteresis (The maximum difference in output shaft position for a given input signal during a full range traverse in each direction).	1.2% of span
Temperature Limits (Ambient)	-40° - 115°F
Duty Cycle	Specify 75%
Characteristic (Input/Output relationship)	Linear

PERFORMANCE CHART
Positioner Mounted in Typical Actuator

Current Draw	5 Watts plus Actuator current drain
Load Resistance 150 ohms for 10-50 mA signal 320 ohms for 4-20 mA signal 1100 ohms for 1-5 mA signal	
Adjustments Zero –25% of span Range –25% of span Anti-hunt control 0.25 to 10% signal	
Feedback Potentiometer	1000 ohms
Options Extra Limit Switches Heater/Thermostat Mechanical Brake/Butterfly Valve Application Direct or Reverse Acting Dual Potentiometer Split Ranging	



NOTE:
For reverse acting (4mA open, 20mA closed), connect wires from potentiometer as follows: W/Black to terminal #1, brown to terminal #3, J1 on board QM1-1.
Reset cam on SW1 to open position, and cam on SW2 to closed position.



NOTE:
For reverse acting (4mA open, 20mA closed), connect wires from potentiometer as follows: W/Black to terminal #3, brown to terminal #1, J1 on board QM1-1.
Reset cam on SW1 to open position, and cam on SW2 to closed position.

	NAME	DATE
DR	KENICHI MIYAZAKI	8/3/01
APPD	DAVE HURLEY	8/3/01
PROD	LEO LESTER	8/3/01
WO#/CO#		
FILE		

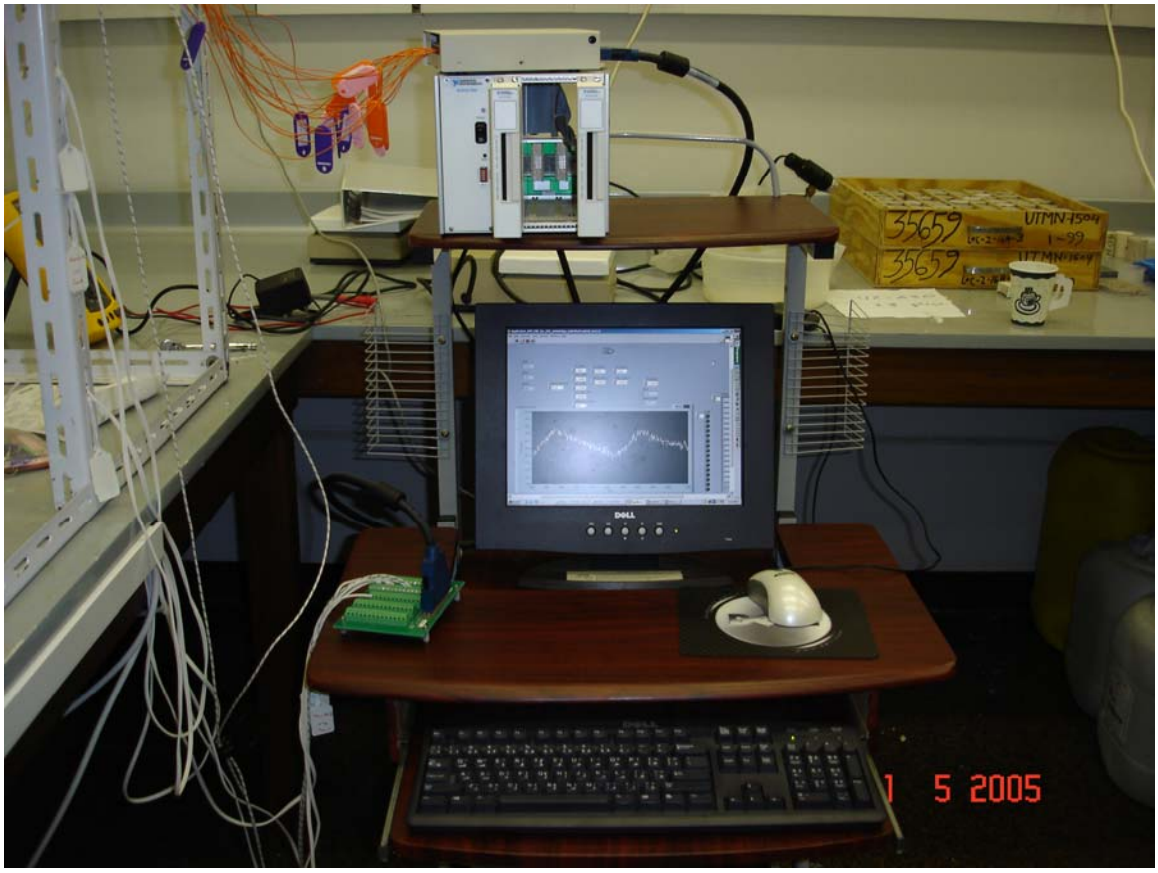
Full Wiring Diagram
Q-Series Actuator w/ Positioner

SIZE	DWG. NO.	REV
A	0006PB	A

SCALE NTS SHEET 1 OF

Appendix D

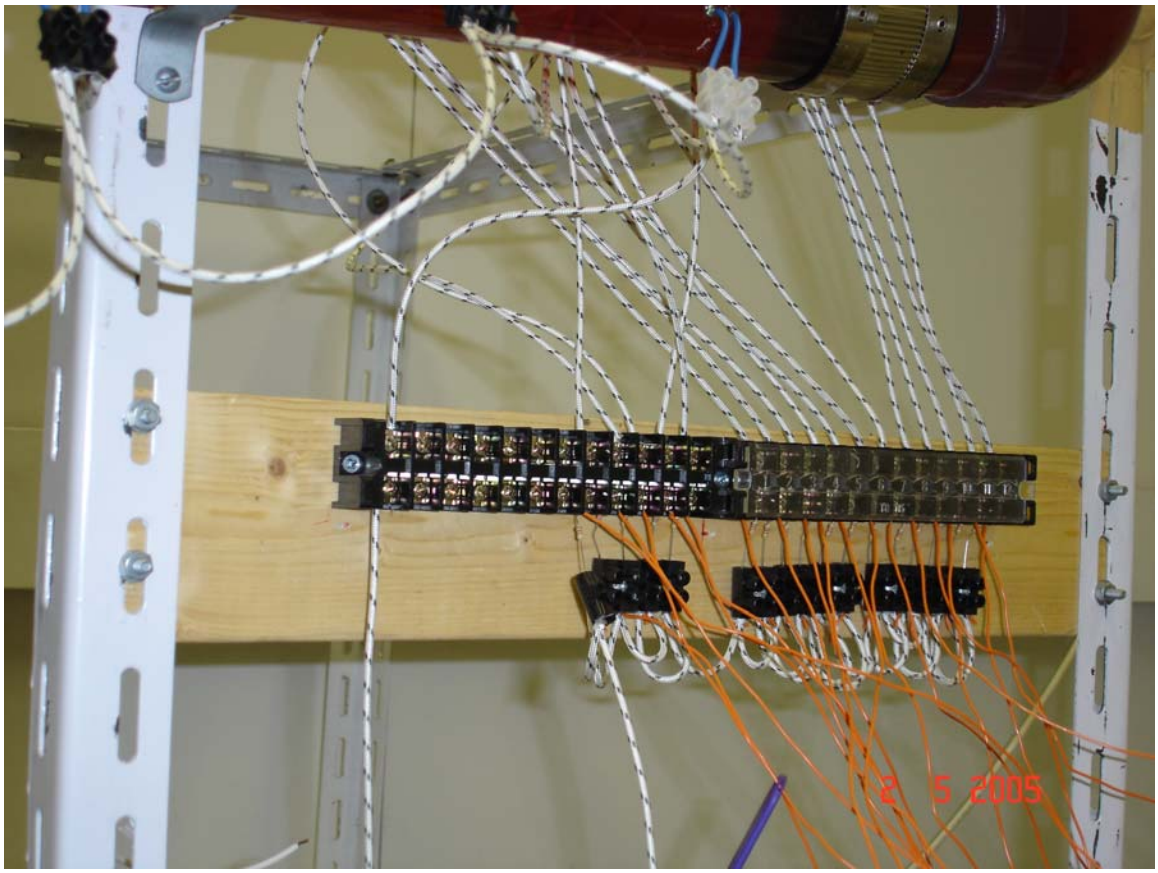
Miscellaneous Experiment Photos



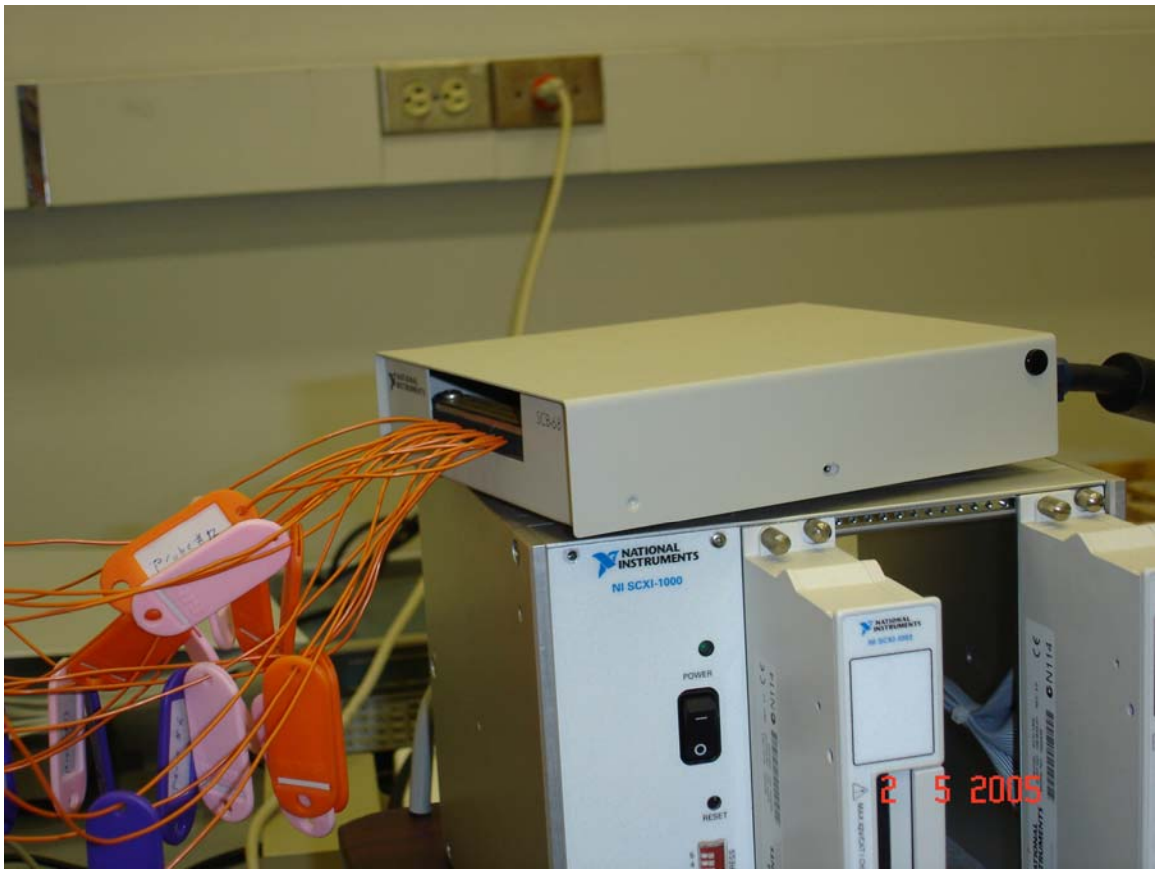
Picture C-1: Computer Setup for the Experiment

Picture C-2 Probes connection at the Uphill Instrumental Section

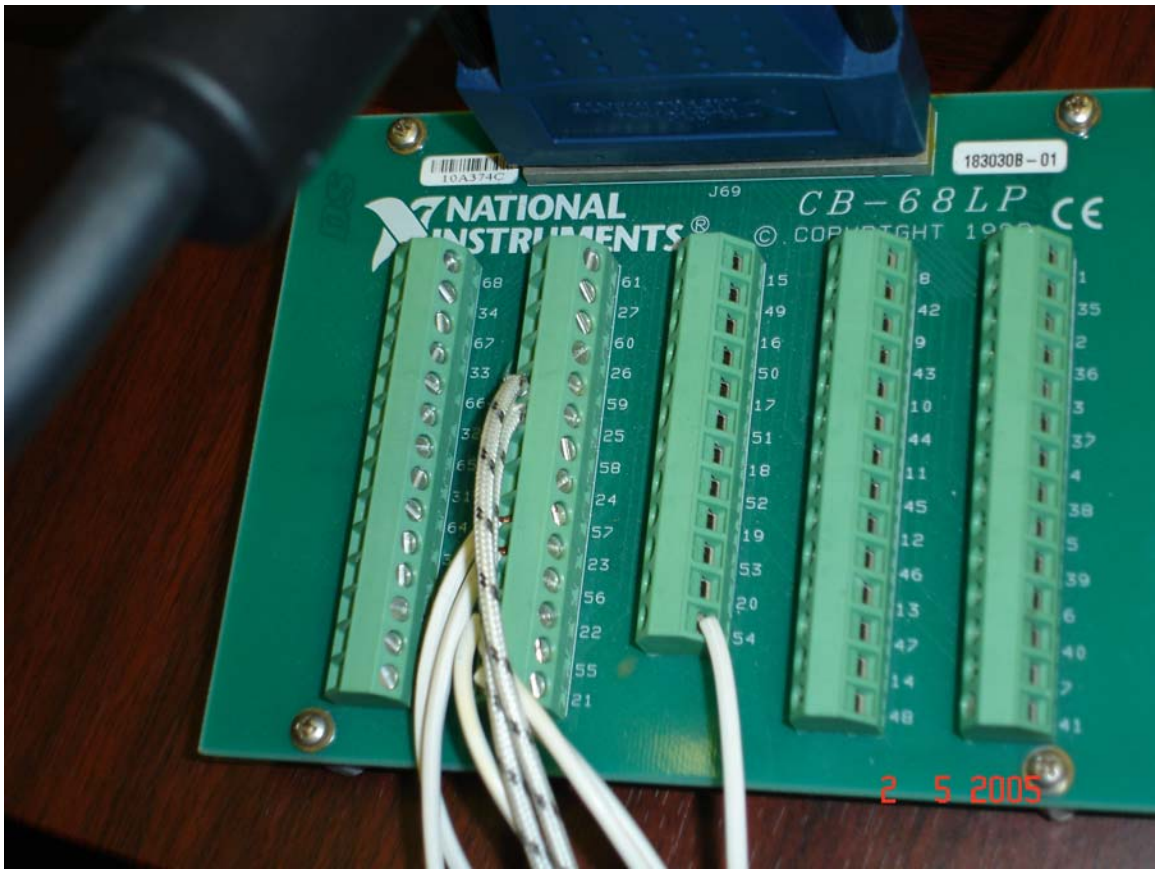
Picture C-3: Probes connection. Water tail can be seen in the instrumental pipe.



Picture C-4: Terminal block for the Probes wires.



Picture C-5 Probes connections to the Analog Input interface card.



Picture C-6 Output interface Card it sends 0~10VDC

Picture C-7: Signal Converter, it converts 0~10VDC to 4~20mADC for the MOV signal.

Picture C-8 : Water Separator Tank receives water from the Extraction Port (Clear- No oil)

Picture C- : Water Separator Tank having some oil because of MOV isolation Valve is opened more.

BIBLIOGRAPHY

[1] Karl J Astorm, Bjorn Wittenmark, ADAPTIVE CONTROL, Addison-Wesley Longman Publishing Co, 2ND Edition, 1994

[2] Emile Fiesler, Russel Beale, Handbook of Neural Computation , Institute of Physics, 1997

[3] Benard Widrow, and Michael A. Lehr, "30 Years of Adaptive Neural Networks: Preceptron, Madaline, and Backpropagation", PROCEEDINGS OF THE IEEE, VOL.78, NO. 9, September 1990.

[4] Ibrahim Babelli, "The In-Line Surface Oil/Water Separator", Schlumberger Confidential Report, 10/05/2004.

[5] Si-Zahao Qin, H. T. Su, and T. J.McAvoy, "Comparison of Four Neural Net Learning Methods for Dynamic System Identification", IEEE Trans on Neural Network, Vol#3 pp.122:130, 1992

[6] Chen Xiahong, G. Geng, Q. Jixin and S. Youxian, "A Non-linear Adaptive Controller Based on RBF Networks", IEEE, pp. 661:666, 1996

[7] Anil K. Jain, Jiancchang Mao, "Artificial Neural Networks: A Tutorial", Computers, March 1996.

[8] Ben Krose, Patrick V. der Smagt, *An introduction to Neural Networks*, University of Amsterdam, 8th Edition, 1996.

[9] A. Cichocki, and R. Unbehauen, *Neural Networks for Optimization and Signal Processing*, John Wiley & Sons, 1994

[10] Simon Haykin, *Neural Networks: A Comprehensive Foundation*, Macmillan College Publising Company, 1994

[11] James A. Feeman, David M. Skaputa, *Neural Networks: Algorithms, Applications, and Programming Techniques*, Addison-Wesley Publishing Company, Inc. 1991.

[12] Haykin, S. , ADAPTIVE FILTER THEORY, Prentice Hall 3rd Edition, 1995

[13] Haykin, S., *KALMAN FILTERING AND NEURAL NETWORKS*, John Wiley & Sons, Inc. 2001

[14] F. C. De Castro, M. C. De Castro, and D. S. Arantes, "A Supervised Neural Constant Bit Rate Video Controller for MPEG-2 Encoders", IEEE pp.504:509,1998

[15] Ohr, J., "ANTI-WINDUP AND CONTROL OF SYSTEMS WITH MULTIPLE INPUT SATURATIONS Tools, Solutions and Case Studies", Dissertation for the degree of Doctor of Philosophy in Electrical Engineering with specialization in Automatic Control at Uppsala University, 2003

[16] Micael Sternad, Stefan Ronnback, "A FREQUENCY DOMAIN APPROACH TO ANTI-WINDUP COMPENSATOR DESIGN", Report UPTEC 93024R, Department of Technology, Uppsala, University, April 1993

[17] Armando B. Corripio, *DESIGN AND APPLICATION OF PROCESS CONTROL SYSTEM*, Instrument Society of America 1998

[18] J. A. Stegemann and N. R. Buenfeld, "A Glossary of Basic Neural Network Terminology for Regression Problems", Neural Comput & Applic (1999)8:290–296

VITA

Name: Wesam Abdullah Al-Shammasi

Date of Birth: 25/05/1974

Nationality: Saudi

Education & Qualification:

B.Sc Degree in Electrical Engineering 1996

King Fahd University of Petroleum and Minerals

Dhahran, 31261, Saudi Arabia

MS Degree in Electrical Engineering, June 2005

King Fahd University of Petroleum and Minerals

Dhahran, 31261, Saudi Arabia