Eng. & Tech. Journal, Vol. 27,No.8, 2009

Design of a Neural Networks Linearization for Temperature Measurement System Based on Different Thermocouples Sensors Types

Ahmed Sabah Abdul Ameer Al-Araji

Received on:6/5/2008

Accepted on:5/3/2009

Abstract

This paper describes an experimental method for the estimation of nonlinearity, calibration and testing of the different types of thermocouples (J and K) using modified Elman recurrent neural networks model based Back-Propagation Algorithms (BPA) learning. Thermocouples sensors are nonlinear in behavior nature but require an output that is linear. The linear behavior approximation is accepted, for a given accuracy level, noise and measurement errors are always present. Therefore, neural networks techniques are frequently required to minimize these effects. The problem of estimating the sensor's input–output characteristics is being increasingly tackled using software techniques such as Turbo C++ language. A neural networks and a data acquisition parallel port interface board with designed signal conditioning unit are used for data optimization and to collect experimental data, respectively. After the successful training completion of the neural networks, it is then used as a neural linearizer to calculate the temperature from the thermocouple's output voltage

Keywords: Neural Networks; Temperature Measurement System; Thermocouples Sensors

تصميم شبكة عصبية خطية لنظام قياس درجة الحرارة اعتمادا على أنواع مختلفة من متحسسات المزدوجات الحراري

الخلاصة

يصف هذا البحث طريقة تجريبية لتقدير اللاخطية,التعيير والفحص لأنواع مختلفة من المزدوج الحراري نوع (J & K) باستخدام شبكة عصبية اصطناعية تم تعديلها بنموذج شبكة ايلمن العصبية باستخدام خوارزمية الانتشار العكسي للتعليم. طبيعة تصرف المزدوجات الحرارية هي اللاخطية ولكن المطلوب ان يكون الاخراج خطي.ولكن التصرف الخطي التقريبي المقبول للمزدوج الحراري دائما يحتوي على مستوى معين للدقة, الضوضاء وخطاء القياس, لهذا السبب تم استخدام تقنية الشبكة العصبية لتقليل هذه المؤثرات. مشكلة تخمين خصائص الإدخال والإخراج للمتحسس بدأت تزداد معالجتها باستخدام تقنيات البرمجيات مثل استخدام لغة Turbo C++.الشبكة العصبية ولوح جمع البيانات باستخدام المنفذ المتوازي للحاسبة مع وحدة شرط تكييف الإشارة ألمصممه, استخدمت في أمثلية البيانات و تجميع البيانات التجريبية على التوالي. وبعد نجاح تعلم الشبكة (النموذج العصبي) على التصرف الخطي, تم حساب درجات الحرارة من خلال فولتية الإخراج للمزدوجات الحرارية.

https://doi.org/10.30684/etj.27.8.16

2412-0758/University of Technology-Iraq, Baghdad, Iraq

1- Introduction:

In recent years, application of neural networks has emerged as a promising area of research in field of instrumentation and measurement [1]. It provides a neurocomputing approach for solving complex problems especially in nonlinear system modeling where the neural itself is a nonlinear system. This is extremely useful when the area of interest is absolutely nonlinear including the experimental data that is used for training. One of the most powerful uses of neural networks is in function approximation (curve fitting) [2]. However, many types of sensors are nonlinear in nature from which a linear output is desired. There are many different sensors for temperature measurement and thermocouples are the most commonly used [3]. They are preferred in industrial applications due to their low cost, wide operation range and fast response time. Thermocouples also have nonlinear outputs related to temperature; therefore, sensor modeling and linearization techniques are necessary.

In this work, a linearization of temperature measurement system based on the neural networks approach is proposed.

The organization of this paper is as follows: Section two represents the thermocouples background. Section three describes the use of feedforward neural networks to learn (Modified Elman Recurrent Neural Networks) by using Back propagation Algorithm. Section four and five represents the core of the present paper, Hardware design and Neural Networks Linearization the nonlinearity of the thermocouples responses. The result simulation of the Neural networks linearization algorithm described by section six. Finally, section seven contains the conclusions of the entire work.

2- Thermocouples Background:

A thermocouple is two wires of dissimilar metals that are electrically connected at one end (measurement junction) and thermally connected at the other end (the reference junction) [4].

Calculations determining the temperature corresponding to a given measured voltage of a thermocouple assume that this voltage corresponds to a temperature gradient that is referenced to 0ºC. The reference junction is allowed to follow ambient temperature, where ambient temperature variations of the reference junction would cause significant errors in the temperature calculation from the voltage output of the thermocouple, two methods of reference junction compensation exist [5].

Maintain the reference junction at a constant known temperature such as an ice bath (0ºC). This is where the term 'cold junction' was originally derived [5].

Obviously the second option is far easier to implement and has led to the design of many cold junction compensation circuits. The necessary voltage correction can be carried out with software, hardware, or a combination of both [5].

2-1- Hardware Compensation:

Hardware compensation requires dedicated circuitry to generate a compensation voltage according to the ambient temperature of the isothermal block, and add this voltage to the voltage measured at the measuring junction. As the voltage vs temperature relationship varies between thermocouples, each thermocouple type must have a separate compensation circuit that operates over the required working range of ambient temperatures. This makes hardware compensation circuitry for thermocouples complex and expensive, and by their nature, prone to inherent errors. There are two types of the analog integrated circuit AD594 and AD595 for J & K thermocouples types respectively that analog circuit are frequently used for improving the linearity of the sensor characteristics, which implies additional analog hardware and typical problems particular to analog circuits such as temperature drift, gain and offset error [6,7].

2-2- Software Compensation:

Software compensation requires only that an additional direct reading temperature sensor, such as a thermistor or silicon sensor, be used to measure the isothermal block temperature of the reference junction. Software is then used to calculate the equivalent reference junction voltage, either by polynomial equations, or look-up tables, for the thermocouple type being used. Once calculated, this value is added to the measured output voltage from the thermocouple. The resulting voltage is converted back to a temperature, representing the true thermocouple temperature [5].

$$T = a_0 + a_1 v + a_2 v^2 + ... + a_n v^n \quad ...(1)$$

Where v is the thermocouple voltage in *volts*, T is the temperature in degrees Celsius, and a_0 through a_n are coefficients that are specific to each thermocouple type table (1) lists NIST ***National Institute of Standards and Testing*** polynomial coefficients for several popular thermocouple types over a selected range of temperature and over a selected range of temperature [5].

3- Recurrent Neural Network and Learning Algorithm:

Recurrent neural networks have been an important focus of research and development during the 1990's. They are designed to learn sequential or time varying patterns. A recurrent net is a neural network with feedback (closed loop) connections [8]. Recurrent neural network techniques have been applied to a wide variety of problems such as learning strings of characters and addressed problems involving dynamical systems with time sequences of events [8].

Recurrent neural networks (RNN) have one or more feed-back connections, where each artificial neuron is connected to the others The RNN structures are suitable to channel equalization and multi-user detection applications, since they are able to cope with channel transfer functions that exhibit deep spectral nulls, forming optimal decision boundaries and are less computationally demanding than MLP (Multi-Layer Preceptron) networks for these applications.

The structure of MENN (Modified Elman Neural Network) shown in figure (1) represents one of the simplest types that can be trained using dynamic BPA (Back Propagation Algorithm), and it is used to minimize the oscillation or even instabilities to the training linearizer network. The output of the context unit in the modified Elman network is given by:

$$h_c^o(k) = a h_c^o(k-1) + b h_c(k-1) \quad \ldots.(2)$$

where $h_c^o(k)$ and $h_c(k)$ are the output of the context unit and hidden unit respectively and a is the feedback gain of the self-connections and b is the connection weight from the hidden units (c'th) to the context units (c'th) at the context layer. The value of a and b are selected randomly between (0 and 1). From figure (1) it can be seen that:

$$h(k) = F\{V1U(k), V2h^o(k)\} \quad \ldots..(3)$$

$$O(k) = Wh(k) \quad \ldots\ldots (4)$$

where V1,V2 and W are weight matrices and F is a non-linear vector function and $U(k)$ is input patterns to the neural networks. The multi-layered modified Elman neural network shown in figure (1) is composed of many interconnected processing units called neurons or nodes. where:

$V1$: Weight matrix of the hidden layers.

$V2$: Weight matrix of the context layers.

W: Weight matrix of the output layer.

L: Denotes linear node.

H: Denotes nonlinear node with sigmoid function.

$U(k)$: Input patterns to the neural networks.

To explain these calculations, consider the general j'th neuron in the hidden layer shown in figure (2). The inputs to this neuron consist of an ni – dimensional vector (ni is the number of the input nodes). Each of the inputs has a weight V1 and V2 associated with it. The first calculation within the neuron consists of calculating the weighted sum net_j of the inputs as [9]:

$$net_j = \sum_{i=1}^{nh} V1_{j,i} \times U_i + \sum_{c=1}^{C} V2_{j,c} \times h_c^o \quad \ldots(5)$$

C and nh number of the context nodes and hidden nodes. For the standard design recurrent neural networks, the number of the context nodes is equal to hidden nodes nh=C, then c=j.

Next the output of the neuron h_j is calculated as the continuous sigmoid function of the net_j as:

$$h_j = \mathrm{H}(net_j) \quad \ldots.(6)$$

$$\mathrm{H}(net_j) = \frac{2}{1+e^{-net_j}} - 1 \quad \ldots..(7)$$

Once the outputs of the hidden layer are calculated, they are passed to the output layer. In the output layer, a single linear neuron is used to calculate the weighted sum (neto) of its inputs (the output of the hidden layer as in equation (8)).

$$\mathrm{neto}_k = \sum_{j=1}^{nh} W_{kj} \times h_j \quad \ldots.(8)$$

Where W_{kj} is the weight between the hidden neuron h_j and the output neuron. The single linear neuron, then, passes the sum (neto_k) through a linear function of slope 1 (another slope can be used to scale the output) as:

$$O_k = L(neto_k) \quad \ldots.(9)$$

The learning (training) algorithm is usually based on the minimization (with respect to the network weights) of the following objective (cost) function mean square error (MSE) as given in equation (10) by using dynamical Back Propagation Algorithm learning.

$$MSE = \frac{1}{Np} \sum_{k=1}^{Np} [(Temp_m(k) - Temp_t(k))^2] \quad \ldots..(10)$$

where: $Temp_m(k)$ is the temperature of the linearization neural networks model at sample k.

$Temp_t(k)$ is the true temperature measurement at sample k.

Np is the number of the training patterns.

4- Hardware Design:

A thermocouple generates a voltage proportional to the measurement junction temperature at mV levels while the cold junction temperature is constant. In order to make an accurate measurement the cold junction temperature must be known. Figure (3) shows the block diagram of the temperature measurement system designed via neural networks in the data reading operation. It consists of four thermocouples (type J & K).

The first two thermocouples (J & K) exposed to a desired temperature for (learning & testing) are connected to (AD594 and AD595) respectively monolithic thermocouple amplifier with cold junction compensation that is configured as a standalone Celsius thermometer.

The second two thermocouples, including signal conditioning circuit (5B37-J and 5B37-K) module to measure the thermocouple output voltage in (mV) at the temperature being measured with the equation (11) [10]:

$$V_{out} = (V_{TC} - V_{Zero}) \times Gain \quad ....(11)$$

Where:

V_{out} is the 5B37 type J & K module output from (0 to 5Volt).

V_{TC} is the thermocouple output voltage in (mV) at the temperature being measured.

Vzero is the thermocouple output voltage in (mV) at the minimum temperature span specified for the 5B37-J is equal to -4.632 mV and for 5B37-K is equal to -3.553 mV.

The gain of the 5B37-J type J is equal to 0.105143(V/mV) and for 5B37-K type-K is equal to 0.086688(V/mV).

The four devices output are connected to the analog CMOS multiplexer (4051), which select the desired thermocouple, or the output of Celsius thermometer. With 12-bit analog to digital converter (ADC 574) and designed of Input/Output interfacing circuit that attached to standard parallel port of the personal computer with SPP (Standard Parallel Port) protocol mode.

The designed signal conditioning circuit has amplifier (TL064) with suitable variable gain output and a 12-bit ADC (AD574) is a high speed, low power 12-bit A/D converter [11,12].

This part contains a successive approximation, and a high-speed parallel interface. Accuracy of the system depends directly on step size of analog to digital converter (ADC). From a 0 to 10 V inputs, LSB of ADC574 is 2.44mV. When AD594 & AD595 are used as a Celsius thermometer, the thermocouple is omitted, and the differential inputs are shunted together to common. In this mode, AD594 and AD595 generate a voltage with a scale factor of 10mV/C^o and its output is used for cold junction temperature data that the written software is used. Output signal of TL064 is digitized by ADC574, its output is connected to the I/O interface card and transferred to parallel port of personal computer where data reduction and optimization are implemented as shown in figure (4).

To establish the modified Elman recurrent neural networks weights and biases, during the learning operation (training phase), the International Temperature System – 1990 (ITS-90) calibration unit, with the accuracy of ±0.05% in the range of (0 to 100C^o) for each thermocouple type is used as the maximum error allow. These voltages and thermocouple type are used as inputs of the neural networks, and thermocouple temperature without cold junction compensation is the output of the neural networks. The data used for learning and testing operation is taken from Celsius thermometer output.

5- Neural Networks Linearization:

In addition to requiring cold-junction compensation, thermocouples are also highly nonlinear, and thus require linearization. For most purposes, some form of software-based linearization is used. Two techniques of linearization are common:

1-) Look-up tables.

2-) Polynomial compensation.

The modified Elman recurrent neural networks architecture is used as a neural linearizer as shown figure (5). The proposed technique involves a neural network to evaluate the thermocouple temperature for two thermocouples types (J & K) that thermocouple output voltage (mV) is given as input patterns to the neural networks.

Training the neural networks by using Back-Propagation Algorithm learning to calculate the temperature involves presenting it with different sets of input values and corresponding measured values which range between (0 to 100C^{o}) and a voltage range between (0 to 10) V.

Differences between the target output taken from (AD594 or AD595) and the neural model output of the neural network are evaluated by the learning algorithm (Back propagation Algorithm) to adapt the weights depend on (MSE) Mean Square Error as Eqs. (10).

The neural linearizer model used for calculating the temperature is shown in figure (5) that is trained with different sets of 100 thermocouple temperatures for each thermocouple that is uniformly distributed between 0 and 100C^{o} as shown in figures (6 and 7), which is obtained in the learning operation.

The input and output data are normalized between (-1.0 and 1.0) before training [9].

After several trials with Back-Propagation learning algorithms it is found that the most suitable network configuration is (2 - 10 - 1). This means that the number of neurons is five for the hidden layer and five for the context neurons as shown in figure (5). After training the weights of the hidden layer, context layer and output layer can be shown in appendix.

The input and output layers have the linear activation function and the hidden layers have the hyperbolic tangent sigmoid activation function.

6- Simulation Result:

Turbo C++ program is used to learn and test the modified Elman neural networks linearizer mode for temperature measurement system with the use of Back-Propagation learning Algorithm to obtain better performance and faster convergence with simpler structure.

The inputs learning patterns of the temperature are from 0 to 100C^{o} and the output voltage of the thermocouple module type J is equal to (0.487V to 1.045V) and for K type thermocouple module, the output voltage is equal to (0.308 to 0.665V) for the same input patterns, and then digitize the output voltage of the thermocouples to the personal computer through the parallel port interface by using ADC574 with 12 bit conversion.

The desired patterns of the temperature are from 0 to 100C^{o}. The output voltage of the thermocouple type J with AD594 analog compensation is equal to (0V to 1V) and for K type thermocouple with AD595 analog compensation, the output voltage is equal to (0 to 1V) for the same input patterns. After 5500 epoch for training, the (MSE) is equal to 1.03×10^{-6}.

As a result, it is observed that the neural model output is the same the actual output of the (AD594 or AD595) as shown in figure (6) for type J thermocouple and type K thermocouple in figure (7).

The error between the actual temperature and the neural temperature for learning set for the type (J and K) thermocouples is shown in figure (8) which shows that the range of error is between ± 0.005 for type (J) thermocouple and ± 0.0065 for type (K) thermocouple.

For the testing set, it is observed that the neural model temperature measurement is close to the desired actual temperature measurement and the neural model output is the same the actual output of the (AD594 or AD595) a can be seen in figure (9) for type J thermocouple and for type K thermocouple in figure (10).

The error between the actual temperature and the neural model temperature for testing set, for the type (J and K) thermocouples, as shown in figure (11), range between ± 0.0075 for type (J) thermocouple and ± 0.01 for type (K) thermocouple. The objective function ***MSE*** for the neural network linearization system is shown in figure (12).

7- Conclusions

In this paper, the proposed technique has a potential future in the field of instrumentation and measurement, this technique for temperature linearization measurement based on a modified Elman neural networks model is proposed. The training process for recurrent neural networks is performed successfully in this study with the use of back propagation algorithm, which gives the following results. Maximum range of error is between ± 0.0075 for type (J) thermocouple and ± 0.01 for type (K) thermocouple. Objective function ***MSE*** for the neural network linearization system, the maximum value is equal to 1.03×10^{-6} at epoch 5500. Gain and offset errors of the signal conditioning circuit are automatically cancelled as a consequence of the usage of the neural networks technique.

8- Appendix:

The weights of the *temperature linearization measurement system* model based on Modified Elman neural networks for two types of the thermocouples sensors.

$$V1_{j,i} = \begin{matrix} -0.33 & 0.745 & 1.34 & 0.297 & -0.34 \\ 0.981 & -2.37 & 1.569 & -3.76 & 0.911 \end{matrix}$$

$$V2_{j,c} = \begin{matrix} -1.127 & 2.345 & -0.32 & -1.09 & 0.22 \\ 0.22 & 0.57 & -0.01 & 0.51 & -2.302 \\ -0.19 & -2.121 & 1.23 & 0.571 & 1.17 \\ 1.145 & -0.823 & 0.034 & -0.25 & -1.145 \\ -1.131 & 1.101 & 1.23 & -1.154 & 0.417 \end{matrix}$$

$$W_{k,j} = -0.998 \quad 1.33 \quad -1.97 \quad 0.255 \quad 0.763$$

9-References

[1]- J.M.D. Pereira, O. Postolache, P.M.B.S. Girao, M. Cretu, "Minimizing Temperature Drift Errors Conditioning Circuits Using Artificial Neural Networks", IEEE Trans. Instrum. Meas. 49 (2000), pp 1122- 1127.

[2]- R. Rojas, "Neural Networks, A Systematic Introduction" Springer Verlag, New York, 1996.

[3]- L. Michalski, K. Eckersdorf, "Temperature Measurement" 2nd edition , Wiley Interscience, 2001.

[4]- John Park and Steve Mackay "Practical Data Acquisition for Instrumentation and Control System", IDC Technologies, British Library Cataloguing in Publication Data, First published 2003, ISBN 07506 57960.

[5]- Matthias Nau "Electrical Temperature Measurement with Thermocouples and Thermometers", 10th expanded edition, Fulda, August 2002, ISBN 3-935742-07-X.

[6] LeFort, Bob, Taking the Uncertainty Out of Thermocouple Temperature Measurement (With the AD594/595), Analog Devices.

[7]-Internet http://www.analog.com/UploadedFiles/Datasheets/421725987AD594_5_c.pdf

[8]- L.R. Medsker and L.C. Jain "Recurrent Neural Networks Design and Applications", by CRC press LLC, 1st edition, 2001.

[9]- Ahmed S. Al-Araji "Genetic Algorithm and Elman Neural Network Used for Tuning the Parameters of the PID Neural Controller Based Model Reference" Iraqi journal of computers, communication, control and systems engineering, University of Technology Vol. 5, No.1, pp. 111-126, 2005.

[10]- Dogan Ibrahim" Microcontroller Based Temperature Monitoring and Control", Elsevier Science and Technology Books, Pub. Date: September 2002, ISBN: 0750655569.

[11]- Steven J. Faulkenberry "Introduction to Operational Amplifiers with Linear Circuit Applications"John-Wiley Inc. 1986.

[12]- Willis J. Tompkins and John G. Webster "Interfacing Sensors to the IBM PC" Prentice – Hall. Inc 1988.

[13]- Ahmed S. Al-Araji" A Neural Controller with a Pre-assigned Performance Index" M.Sc. Thesis, University of Technology, November 2000.

Table 1. NIST Polynomial Coefficients for Voltage-to-Temperature Conversion ($T = a_0 + a_1 v + a_2 v^2 + ... + a_n v^n$)

	Thermocouple Type					
	E	J	K	R	S	T
Range	0° to 1,000 °C	0° to 760 °C	0° to 500 °C	-50° to 250 °C	-50° to 250 °C	0° to 400 °C
a_0	0.0	0.0	0.0	0.0	0.0	0.0
a_1	1.7057035E -2	1.978425E -2	2.508355E -2	1.8891380 E-1	1.84949460E -1	2.592800E -2
a_2	-2.3301759E -7	-2.001204E -7	7.860106E -8	-9.3835290E -5	-8.00504062E -5	-7.602961E -7
a_3	6.5435585E -12	1.036969E -11	-2.503131E -10	1.3068619E -7	1.02237430E -7	4.637791E -11
a_4	-7.3562749E -17	-2.549687E -16	8.315270E -14	-2.2703580E -10	-1.52248592E -10	-2.165394E -15
a_5	-1.7896001E -21	3.585153E -21	-1.228034E -17	3.5145659E -13	1.88821343E -13	6.048144E -20
a_6	8.4036165E -26	-5.344285E -26	9.804036E -22	-3.8953900E -16	-1.59085941E -16	-7.293422E -25
a_7	-1.3735879E -30	5.099890E -31	-4.413030E -26	2.8239471E -19	8.23027880E -20	
a_8	1.0629823E -35		1.057734E -30	-1.2607281E -22	-2.34181944E -23	
a_9	-3.2447087E -41		-1.052755E -35	3.1353611E -26	2.79786260E -27	
a_{10}				-3.3187769E -30		
Error	±0.02° C	±0.05° C	±0.05° C	±0.02° C	±0.02° C	±0.03° C

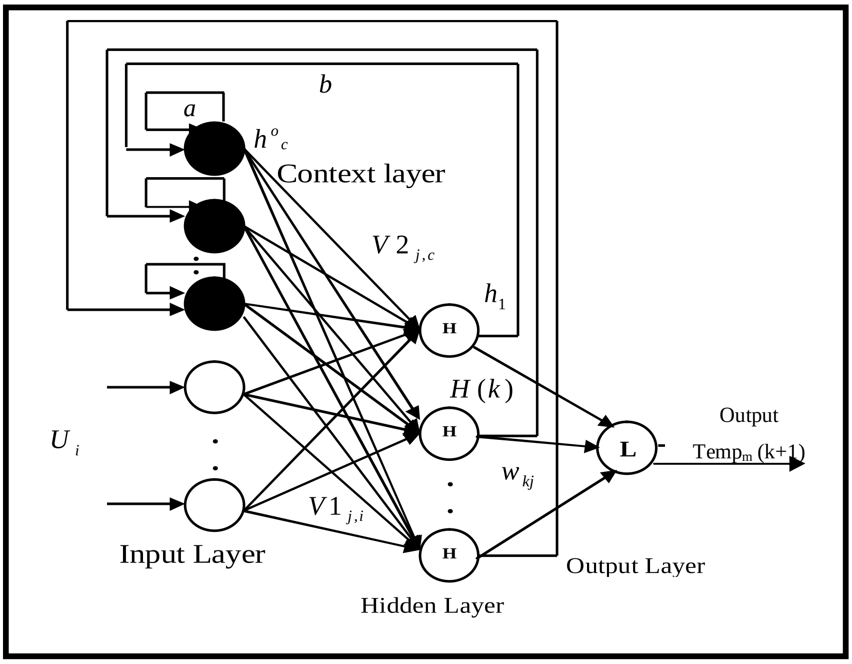


Figure (1): Modified Elman Recurrent Neural Networks.

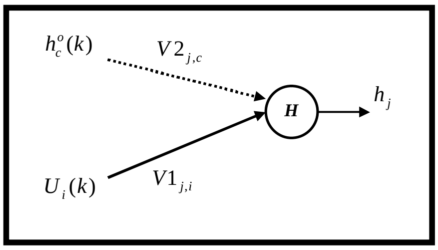


Figure (2): Neuron j in the Hidden Layer.

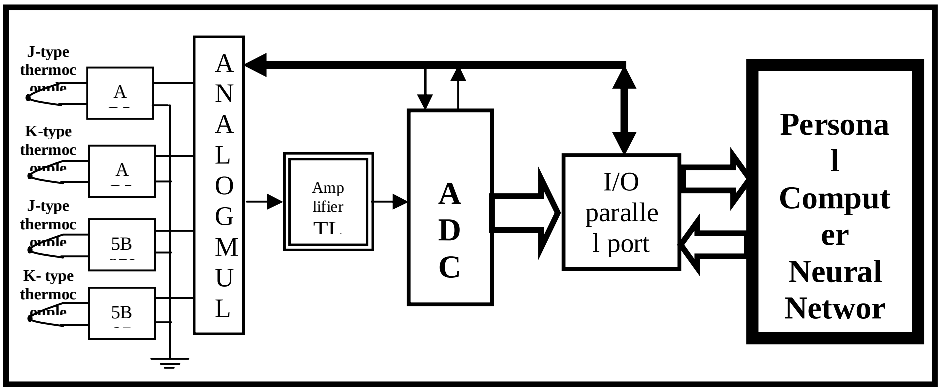


Figure (3): Block Diagram of

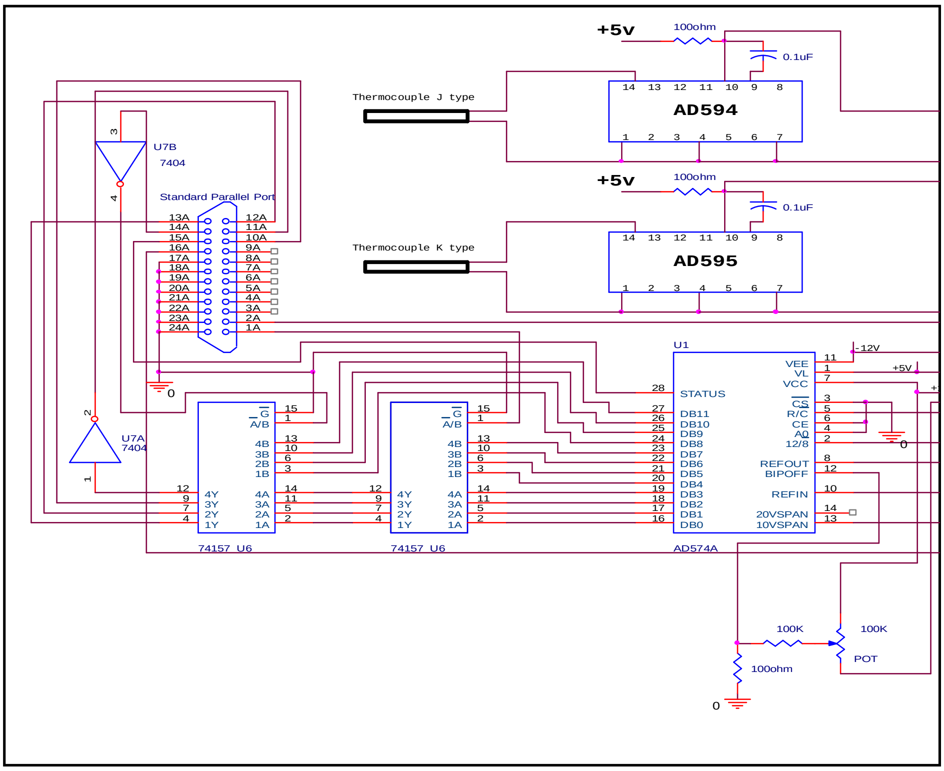


Figure (4): The over all schematic diagram of the interfacing

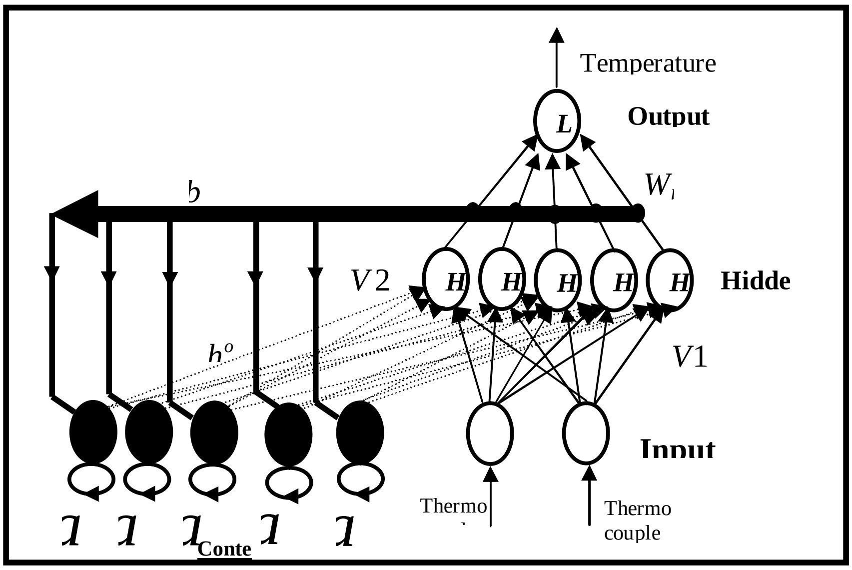
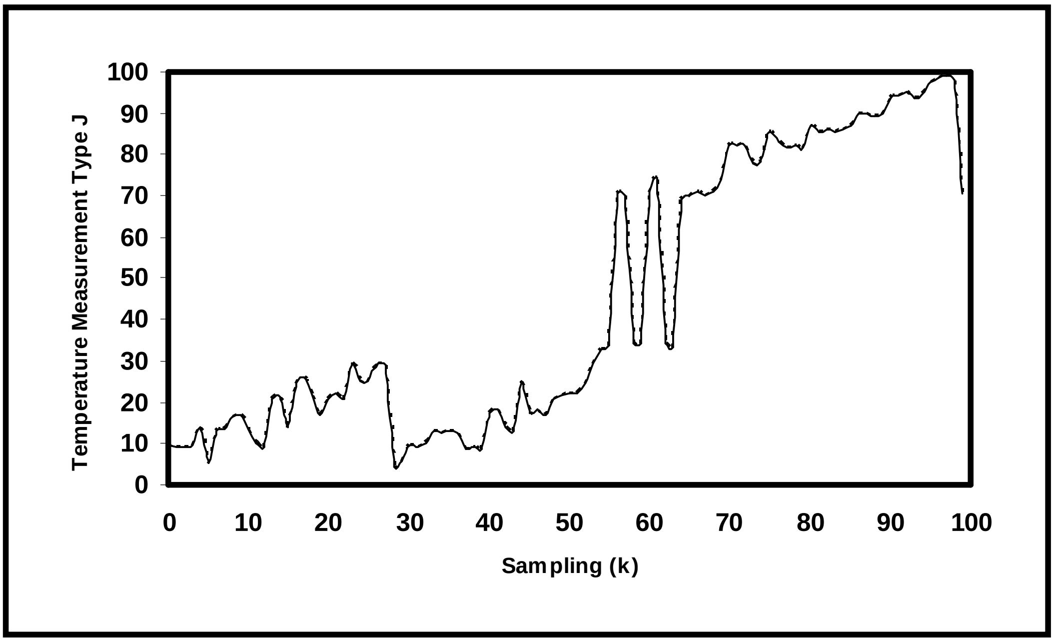
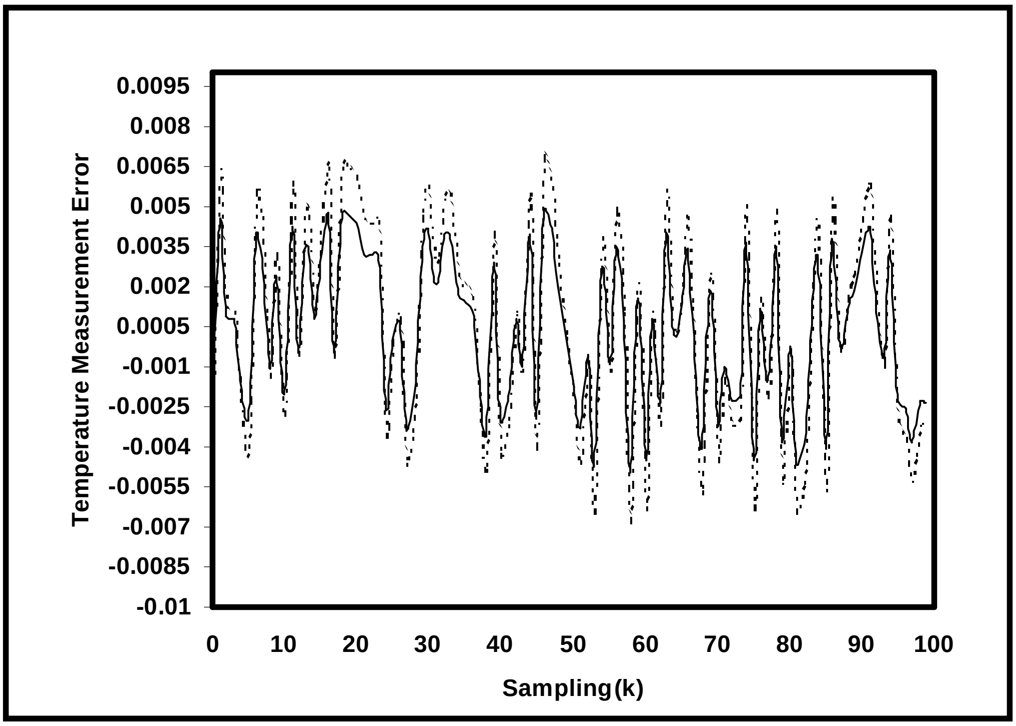


Figure (5): The Weights of Suitable Modified Elman



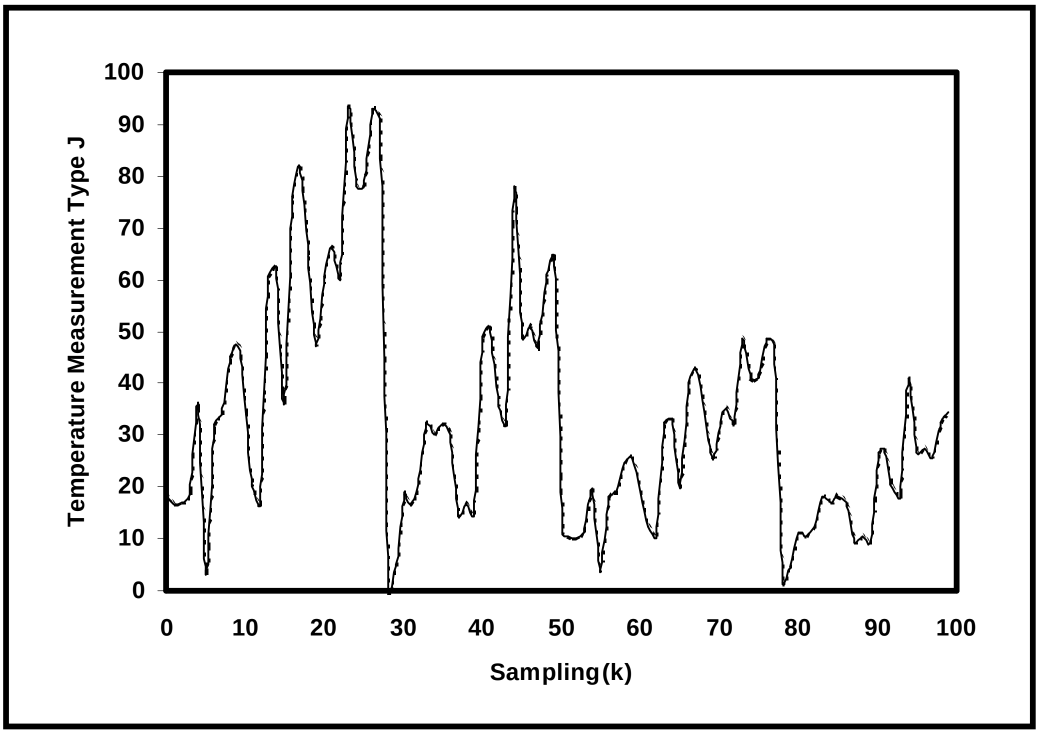
——— The actual temperature for type J thermocouple
------ The neural temperature for type J thermocouple

Figure (6): Temperature Measurement of Thermocouple type (J) for the Learning Set



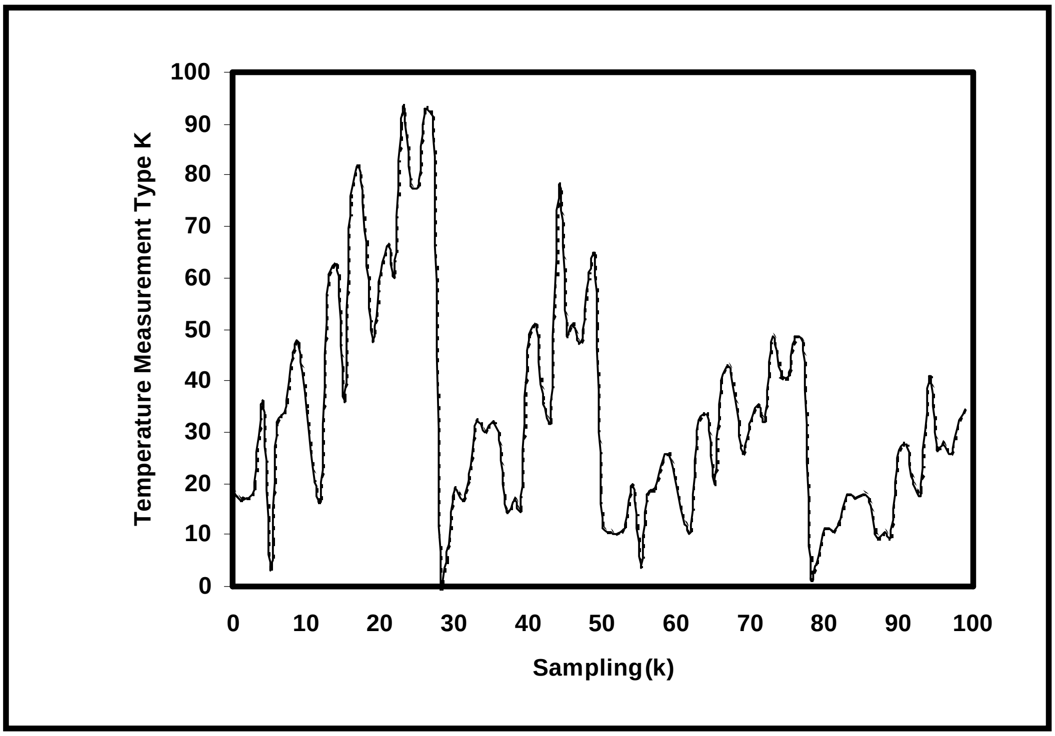
——— The error between actual temperature and neural temperature for type J thermocouple
- - - - - The error between actual temperature and neural temperature for type K thermocouple

Figure (8): Temperature Measurement Error of Thermocouple type (J and K) for the Learning Set.



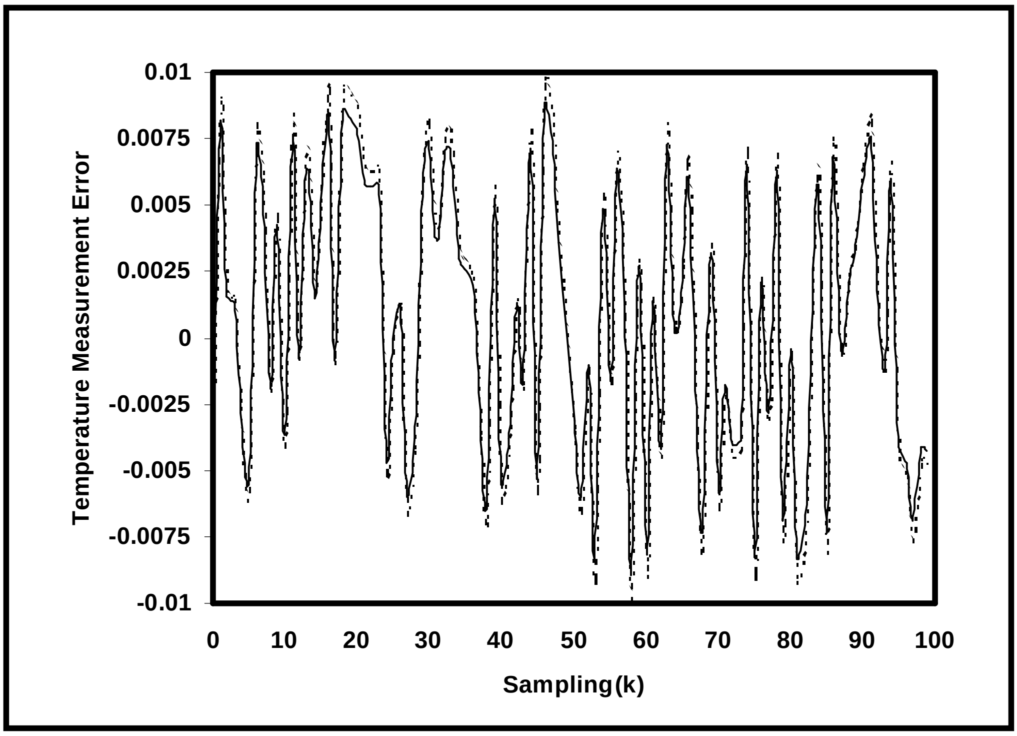
——— The actual temperature for type J thermocouple
- - - - - The neural temperature for type J thermocouple

Figure (9): Temperature Measurement of Thermocouple type (J) for the Testing Set.



——— The actual temperature for type K thermocouple
- - - - - The neural temperature for type K thermocouple

Figure (10): Temperature Measurement of Thermocouple type (K) for the Testing Set.



——— The error between actual temperature and neural temperature for type J thermocouple
- - - - - - - - - The error between actual temperature and neural temperature for type K thermocouple

Figure (11): Temperature Measurement Error of Thermocouple type (J and K) for the Testing Set.

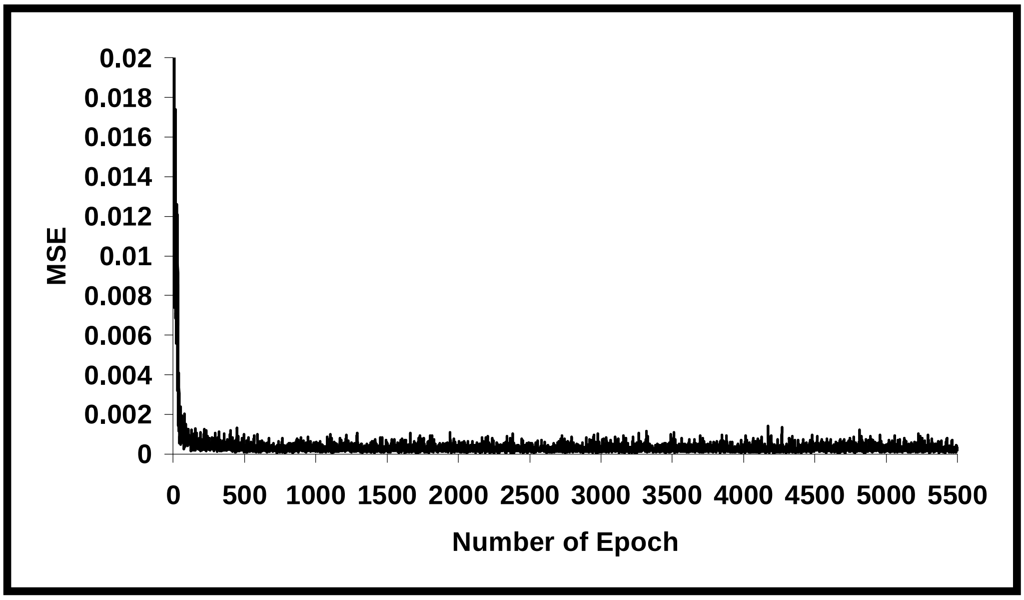


Figure (12): The Objective Function ***MSE***.