COMPUTER-BASED INSTRUMENTATION SYSTEM FOR TEMPERATURE MEASUREMENT USING THERMOCOUPLE IN MATLAB APPLICATION

JINA ANAK MULI

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

BORANG PENGESAHAN STATUS TESIS♦

JUDUL: **COMPUTER-BASED INSTRUMENTATION SYSTEM FOR TEMPERATURE MEASUREMENT USING THERMOCOUPLE IN MATLAB APPLICATION**

SESI PENGAJIAN: 2007/2008

Saya **JINA ANAK MULI (850911-13-6303)**
(HURUF BESAR)

mengaku membenarkan tesis (Sarjana Muda/~~Sarjana~~ /~~Doktor Falsafah~~)* ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hakmilik Universiti Malaysia Pahang (UMP).
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (√)

	SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
√	**TIDAK TERHAD**	

Disahkan oleh:

_________________________ _________________________

(TANDATANGAN PENULIS) (TANDATANGAN PENYELIA)

Alamat Tetap:

KM1 JLN SPG ULU SEGAN/ SEBAUH
P.O BOX 3114, 97013 BINTULU
SARAWAK

MOHD ASHRAF BIN AHMAD
(Nama Penyelia)

Tarikh: ________________ Tarikh: :________________

CATATAN:
* Potong yang tidak berkenaan.
** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh tesis ini perlu dikelaskan sebagai atau TERHAD.
♦ Tesis dimaksudkan sebagai tesis bagi Ijazah doktor Falsafah dan Sarjana secara Penyelidikan, atau disertasi bagi pengajian secara kerja kursus dan penyelidikan, atau Laporan Projek Sarjana Muda (PSM).

"I hereby acknowledge that the scope and quality of this thesis is qualified for the award of the Bachelor Degree of Electrical Engineering (Electronics)"

Signature : ____

Name : MR MOHD ASHRAF BIN AHMAD

Date : __________________

COMPUTER-BASED INSTRUMENTATION SYSTEM FOR TEMPERATURE MEASUREMENT USING THERMOCOUPLE IN MATLAB APPLICATION

JINA ANAK MULI

This thesis is submitted as partial fulfillment of the requirements for the award of the Bachelor of Electrical Engineering (Hons.) (Electronics)

Faculty of Electrical & Electronics Engineering
Universiti Malaysia Pahang

NOVEMBER, 2008

Signature : ______________________________

Author : JINA ANAK MULI

Date : __ NOVEMBER 2008

To my beloved Mum and Dad,

(*Mr. Muli Anak Lampun & Mrs. Chua SimBut)*

Specially to my siblings and also other family members

I love you

&

Thanks for Your Infinitely Support

ACKNOWLEDGEMENT

This project could not have been completed without Miss Najidah Hambali who served as my supervisor and Mr Mohd Ashraf bin Ahmad as my co-supervisor. They patiently guided, encouraged and challenged me throughout in this project. They are supportive of my project and work actively to provide me with the extra time to achieve my project objectives especially Mr Mohd Ashraf bin Ahmad that have make me realized that we should never give up until the time is come . Sincerely thanks to him.

I felt grateful to have supportive and joyful friends: Nurliyana and Faiz. We share our point of view for our own project; share our difficulties and the solution to the problems. Thanks for walking along with me on this pathway.

Nobody has been important to me in the pursuit of this project than the members of my family. I would like to thanks to my beloved parent, whose love and guidance are with me in whatever I pursue. They are my role model.

ABSTRACT

A computer- based instrumentation system has been developed for temperature measurement. The measurement is using thermocouple type K. Thermocouple will detect temperature then send the input to temperature transmitter. Temperature transmitter will accept the input from thermocouple or dc milivolt input and convert it to a 4 to 20mA signal for transmission. Then, the temperature transmitter connected to data acquisition (PCI1710HG) that used to read data from the temperature transmitter. SIMULINK model is use to interface DAQ with MATLAB. Analog input from DAQ will transfer to MATLAB Workspace. The system is developed with MATLAB Graphical User Interface (GUI). GUI will capture data from workspace and then calculated average of actual output and output error. The process is continued by plotting five point temperature calibrations and error curve. The system also can do the evaluation of the uncertainty of temperature measurement. Uncertainty of measurement is the doubt that exists about the result of any measurement. The system allowed the user to save the plot and key in the data in excel. By implementing this system in Industrial Instrumentation class, the improvement and continuity of learning process will be achieved.

ABSTRAK

Sistem instrumentasi berasaskan sistem komputer telah dibangunkan untuk digunakan dalam pengukuran suhu. Pengukuran suhu adalah dengan menggunakan thermocouple jenis K. Thermocouple akan mengesan suhu dan seterusnya akan menghantar bacaan tersebut kepada pemancar suhu. Pemancar suhu akan akan menerima bacaan yang dihantar daripada thermocouple dan seterusnya menukar input yang diterima ke dalam bentuk arus iaitu 4 – 20 mA bagi memudahkan penghantaran data.kemudian, kesemua alat-alat yang digunaka dalam pengukuran suhu akan disambungkan dengan PCI1710HG iaitu satu alat yang digunakan untuk menerima data dan membaca data daripada alat pengukuran suhu. Model SIMULINK digunakan untuk menghubugkan PCI1710HG dengan MATLAB. Seterusnya, input analog akan dihantar ke MATLAB workspace. Sistem ini telah dibangunkan dengan menggunakan MATLAB Graphical User Interface (GUI). GUI akan menyalin data dari WORKSPACE kemudian data tersebut akan digunakan bagi pengiraan purata untuk bacaan sebenar dan peratusan ralat. Proses seterusnya adalah melakar graf untuk graf lima titik pengujian alat pengukuran suhu dan graf untuk lengkungan ralat bacaan. System ini juga turut memuatkan atau digunakan untuk evaluasi untuk ketidakpastian bacaan bagi pengukuran suhu. Sistem ini turut membolehkan pengguna menyimpan graf yang sudah dilakar dan kemudian memasukkan kesemua bacaan dan pengiraan ke dalam EXCEL. Ringkasnya pengunaan system ini dalam pembelajaran bagi matapelajaran Industrial Instrumentation akan menyebabkan peningkatan dan kelancaran dalam process pembelajaran dapat dicapai.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

GUI	Graphical User Interface
MSU	Master Standard Unit
UUT	Unit Under Test
GUIDE	Graphical User Interface Design Environment
U1	Uncertainty Due To Repeatability Of the Experiment
U2	Uncertainty Contribution Due To MSU Error
U3	Uncertainty Due To UUT Resolution
Uc	Combined Standard Uncertainty
Ve	Effective Degree of Freedom
STD	Standard
DAQ	Data Acquisition

CHAPTER 1

INTRODUCTION

1.1 Project Overview

Computer-based measurement systems are used in a wide variety of applications. Computer-based instrumentation for temperature is measure system that place the instrument's intelligence and measurement circuitry of temperature inside the computer. Instrument that use are digital thermometer 7563, ISOTECH Jupiter 650B,Yokogawa Temperature Transmitter (PT100),HART 375 Field Communicator and two thermocouples type k (one thermocouple as a reference and other one as a measured value).

Generally, thermocouple is a sensor for measuring temperature. Thermocouple operation is based on the physical principles that if two dissimilar metal wires are joined together and the point of joining is heated (or cooled), a voltage difference appears across the two unheated end. Type K thermocouples are made up of a positive Chromel wire and a negative Alumel wire. They are the most popular thermocouple type and offer a wide measurement range with good temperature precision. Thermocouple type k can read temperature in range -200 °C to +1200°C and develops approximately 0.04 mV/°C.

Data acquisition (DAQ) is the sampling of the real world to generate data that can be manipulated by a computer. In order to take measurements with computer-based DAQ hardware, temperature transmitter, signal conditioning equipment and software such as MATLAB is needed. DAQ typically involves acquisition of signals and waveforms and processing the signal to obtain desired information. In this project, PCI 1710HG will be use. The PCI-1710 Series are multifunction cards for the PCI bus. Their advanced circuit design provides higher quality and more functions, including the five most desired measurement and control functions: 12-bit A/D conversion, D/A conversion, digital input, digital output, and counter/timer [1].

MATLAB GUI is a high-level language and interactive environment that enables you to perform computationally intensive tasks faster than with traditional programming languages such as C, C++, and FORTRAN. MATLAB is a numerical computing environment and programming language. Created by Te Math Work, MATLAB allows, plotting of function and data, implementation of algorithms, creation of user interfaces, and interfacing with programs in other languages [2].

1.2 Problem Statement

The problem statement is to do the improvement or continuity of learning process. Before this student only do the experiment at lab and then insert the data and draw the graph manually. This will cause inaccurate results. So to solve this problem, we need to create a system using MATLAB which will produce high accuracy result. This system can use data for uncertainty evolution and plot five point temperature calibrations. It also uses to compare and measure temperature value and actual value and automatically calculate the output error and graphing for output error curve. So that, in future this system can be use for learning process.

1.3 Objectives

This project has three objectives. The first objective is to understand the basic measurement principles of temperature transmitter using thermocouple type k. This is very important to make more understand the basic of temperature measurement so that the project can do smoothly.

Second objective is to develop a hardware that can use to integrate signal from instrument to software. DAQ model PCI1710HG Advantech is used in this project. The basic operation of DAQ card is also need to be studied.

The third objective is to develop the system using MATLAB GUI for student to use in the future. The application software such as MATLAB GUI is the brain of DAQ system. The application of MATLAB GUI controls the DAQ hardware for acquiring data. Once the data is acquired, the MATLAB GUI can use to analyze and present the data.

1.4 Scopes

There are several scopes that need to be proposing in this project. First scope is to do the basic temperature measurement such as five point calibration of temperature transmitter. Calibrations are the process of determining the relation between the output of measuring instrument and the value of the input quantity or attribute a measurement standard.

The second scope is to integrate the signal from instrument to software, so the hardware is needed to be developed. Hardware that will be use is DAQ card which is a basic A/D converter coupled with an interface that allows a personal computer to control the actions of the A/D, as well as to capture the digital output information from the converter. A DAQ card is designed to plug directly into a personal computer's bus.

All the power required for the A/D converter and associated interface components is obtained directly from the PC bus.

Then the third scope is to create a program using MATLAB that can interface with the hardware component. The PC will read the data (temperature reading) from the Thermocouple reading using real time application. Then, the program that creates using MATLAB will do the evaluation of uncertainty of the measurement.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The literature on fundamentals and applications of data acquisition, instrumentation, and control to engineering and technology is very extensive. Today in any type of computer aided manufacturing project work and laboratory tests, precision and reliability of instrumentation and data acquisition techniques may cause major impacts on results and outcomes. Therefore, students in technology programs must gain knowledge and skills pertinent to their curriculum and job requirements when they need to obtain any type of physical or virtual data on manufacturing, testing, measurement, and protection areas [1].

2.2 MATLAB GUI Based Instrumentation System

At the simplest level, data acquisition can be accomplished manually using paper and pencil, recording readings from a multimeter or any other instrument. For some applications this form of data acquisition may be adequate. However, data recording applications that require large number of data readings where very frequent recordings are necessary must include instruments or microcontrollers to acquire and record data precisely [2].

After more than 20 years of development, MATLAB has evolved from a powerful matrix calculation application into a universal programming tool used extensively within scientific and engineering communities both commercial and academic. MATLAB versions 6.0 and 7.0 include functionality for developing advanced graphical user interfaces, GUIs, and real-time animation and graphics. GUI applications offer many advantages for users who wish to solve complex problems by providing interactivity and visual feedback. GUI development can apply in instrumentation and data Acquisition interfaces [3].

2.3 Graphical User Interface

The graphical user interface (GUI) is intended to give a view of the status of the data acquisition system and its sub-systems (e.g. Event Transfer, Event filter, Event builder, Back End, etc.) and to allow the user to control its operation. The GUI was developed not only for general users, such as shift operators, but also to provide DAQ experts the ability to control and debug the DAQ system. The run control system can have many GUIs associated with a particular experiment. However, only one GUI can be a master, capable of controlling the DAQ system. The rest of the GUIs will visualize the monitored information. In the run control environment the GUI is considered to be a software component and will have an associated agent in the platform to interact with the DAQ/control component agents [4].

2.4 Temperature calibration and measurement

2.41 Fundamental of Temperature Calibration

Temperature is one of the most frequently measured parameters in industrial processes. Wide varieties of mechanical and electrical thermometers are used to sense and control process temperatures. Regular calibration of these thermometers is critical to ensuring consistent quality of product manufactured, as well as providing regulatory compliance for some industries [5].

Most simply stated, temperature calibration consists of placing a thermometer under test into a known, stable temperature environment. A comparison is made between the actual temperature and the reading indicated by the thermometer under test and the difference is noted.

Adjustments can then be made either directly to the thermometer or to its readout. Electrical thermometers are adjusted by mathematically re-creating the coefficients used by SMART transmitters or other readout devices to translate their electrical output to temperature.

In industrial applications, the temperature environment is usually provided by a drywell, or "dry-block" calibrator, or a micro-bath. Both offer portability and a wide range of temperatures. Drywells use high stability metal blocks with drilled wells to accept the reference and UUT. Drywells typically cover ranges from -45°C to 1200°C and micro-baths cover ranges from -25°C to 200°C. Micro-baths are similar in size to drywells but use a small tank of stirred fluid instead of a metal block. Micro-baths offer significant advantages when calibrating short or odd shaped probes.

2.42 External and Internal Reference

Micro-baths and dry-wells have a built-in sensor to provide a feedback loop to the unit's controller and to provide a temperature reading to the user. The manufacturer of the heat source can calibrate this sensor so the unit displays a traceable temperature within a stated uncertainty.

The reference system, however, should be more accurate than the process system being calibrated. As a general rule, temperature uncertainties are larger at higher temperature. Using an internal reference is sometimes preferred because it requires fewer instruments and enhances portability for field applications. This block diagram is illustrated in Figure 2.1.

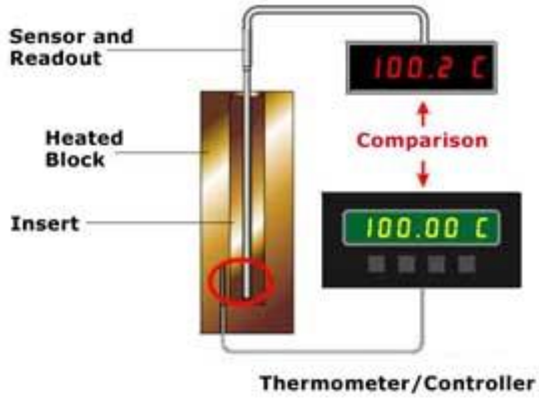

Figure 2.1: Heat source as reference standard

Uncertainty requirements are more rigorous, external references thermometers help improve system uncertainty (see Figure 2.2). Because external thermometers are more accurate, they increase the relative significance of other components of calibrations uncertainty, such as uniformity and stability. It is, of course, critical in any calibration to account for all sources of uncertainty in the process.

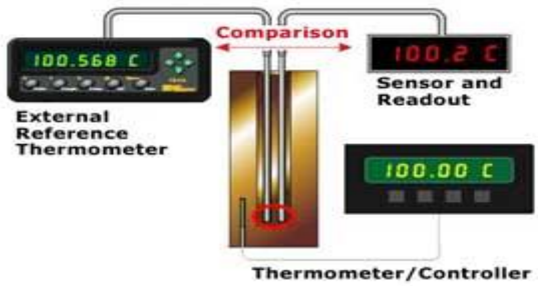

Figure 2.2: External preference standard

2.43 Component Calibrations

Most temperature sensors used in processes are read by transmitters, which send a 4 to 20 mA signal to a control panel, which then displays the temperature for process monitoring. Several calibration methodologies are used in the process plant and the most representative method being to calibrate the complete measurement system from sensor through transmitter to indicator or controller; alternatively each component of the measurement system can be individually calibrated.

The temperature sensor can be individually calibrated using a drywell or micro-bath heat source to simulate the process temperature. If the temperature sensor is electrical, a readout device measures its output.

The transmitter is calibrated using a precision simulator to generate the resistance or voltage output from the temperature sensor and input to the transmitter. The simulator also measures the resulting transmitter current or voltage output. The transmitter is adjusted to ensure that the output follows the input, e.g. for a 4 to 20 mA transmitter with a range of 50°C to 200°C, 4 mA corresponds to 50°C and 20 mA corresponds to 200°C. The simulator provides a wide range of input and output ranges to cover all resistance thermometer and thermocouple types.

The indicator or controller is also calibrated using a precision simulator to generate simulate the resistance or current input from the transmitter. The indicator or controller is adjusted so that the display variable matches the simulated input.

2.5 Software Interfacing

Data acquisition is the sampling of the real world to generate data that can be manipulated by a computer. Sometimes abbreviated DAQ or DAS, data acquisition typically involves acquisition of signals and waveforms and processing the signals to obtain desired information. The components of data acquisition systems include appropriate sensors that convert any measurement parameter to an electrical signal, which is acquired by data acquisition hardware.

Acquired data is displayed, analyzed, and stored on a computer, either using vendor supplied software, or custom displays and control can be developed using various text-based programming languages such as BASIC, C, Fortran, Java, Lisp, Pascal. It is a standard programming method to access data acquisition hardware. MATLAB provides a programming language but also built-in graphical tools and libraries for data acquisition and analysis.

2.51 Data Acquisition

Data acquisition begins with the physical phenomenon or physical property of an object (under investigation) to be measured. This physical property or phenomenon could be the temperature or temperature change of a room, the intensity or intensity change of a light source, the pressure inside a chamber, the force applied to an object, or many other things. An effective data acquisition system can measure all of these different properties or phenomena.

A transducer is a device that converts a physical property or phenomenon into a corresponding measurable electrical signal, such as voltage or current. The ability of a data acquisition system to measure different phenomena depends on the transducers to convert the physical phenomena into signals measurable by the data acquisition hardware.

2.52 DAQ Hardware

DAQ hardware is what usually interfaces between the signal and a PC. It could be in the form of modules that can be connected to the computer's ports (parallel, serial, USB, etc...) or cards connected to slots (PCI, ISA) in the mother board. Usually the space on the back of a PCI card is too small for all the connections needed, so an external breakout box is required. The cable between this Box and the PC is expensive due to the many wires and the required shielding and because it is exotic. DAQ-cards often contain multiple components (multiplexer, ADC, DAC, TTL-IO, high speed timers, RAM).

CHAPTER 3

HARDWARE DESIGN

3.1 Instrument

The process of create computer-based instrumentation for temperature can be divided in to three main parts. There are instruments, hardware and software. The instruments are digital thermometer 7563, ISOTECH Jupiter 650B, Yokogawa Temperature Transmitter (PT100), Yokogawa digital manometer MT 220, HART 375 Field Communicator and two thermocouples type k (one thermocouple as a reference and other one as a measured value).

3.1.1 Digital Thermometer 7563

The 7563 Digital Thermometer has 16 ranges of temperature sensors and DC, V, and Ohm measuring functions. Yokogawa-original A/D converter (feedback pulse width modulation method) features superior noise immunity, stability and high-speed sampling. In addition, versatile functions are suitable for system use and cover a wide variety of applications from test to R&D. Figure 3.1 shows the digital thermometer 7563.

Figure 3.1: Digital Thermometer 7563

3.1.2 ISOTECH Jupiter 650B

The Isotech Jupiter 650 B offers industry-leading performance in an easy-to-use portable package. With its wide temperature range, the Isotech Jupiter 650 dry block calibrator will reach that important 1200°F mark making it ideal for the calibration of thermocouples as well as platinum resistance thermometers. The Isotech Jupiter 650 dry block calibrator has been designed for fast heating and cooling for convenient field use. For flexibility, surface sensor and infrared thermometer accessories can be added.

Linear Process Inputs including 4-20mA current transmitters to be displayed on the built-in indicator. The Isotech Jupiter 650 dry block calibrator's SITE indicator is commonly used to display an external standard thermometer giving greater accuracy by eliminating temperature gradient and loading errors. Figure 3.2 shows the ISOTECH Jupiter 650B.

Figure 3.2: ISOTECH Jupiter 650B

3.1.3 Yokogawa Temperature Transmitter (PT100)

Temperature Transmitter use to convert thermocouple input to analog signals for direct interface with computer- based system. Temperature transmitter will accept the input from thermocouple and convert it to a 4 to 20mA signal for transmission. Signal from the thermocouple are very weak and can be easily affected by RFI and EMI interferences. These transmitters are robust and immune to RFI and EMI interferences. It also allows upscale and downscale temperature setting. User can set error correction and sensor failure detection if desire with ease. Figure 3.3 shows the Yokogawa Temperature Transmitter (PT100).

Figure 3.3: Yokogawa Temperature Transmitter (PT100)

3.1.4 Yokogawa digital manometer MT 220

The MT220 can measure temperature with outstanding accuracy, high resolution and excellent stability. It offers a wealth of functions for field calibration, including transmitter output measurement (DCV/DCA functions), 24-V DC output, percent error readout, measurement data memory, and Ni-Cd battery operation. Figure 3.4 shows the Yokogawa digital manometer MT220.

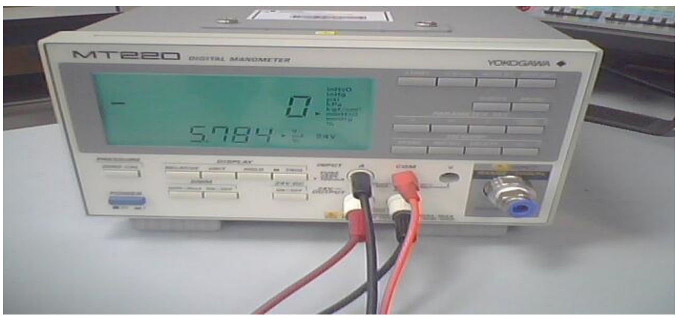

Figure 3.4: Yokogawa digital manometer MT 220

3.1.5 HART 375 Field Communicator

375 Field Communicator is the new standard in handheld communicators. It was universal HART® and Foundation TM field bus, intrinsically safe, rugged and reliable. Figure 3.5 shows HART 375 Field Communicator.

-

Figure 3.5: HART 375 Field Communicator

3.1.6 2793 Decade Resistance Box

Model 2793 is high-accuracy, stable DC variable resistor with 6 dials and is available in two styles: 279301 for medium resistance from 0.1 to 1,111.210 Ω in 1 mΩ steps (best suited for calibration of resistance thermometers or bridges); 279303 for high resistance from 0 to 111.1110 MΩ in 100 Ω steps (suitable for calibration of insulation resistance testers or bridges).Figure 3.6 shows 2793 Decade Resistance Box.

Figure 3.6: 2793 Decade Resistance Box

3.2 Thermocouple Type K

Thermocouple is a temperature sensor that measure temperature by generating a small voltage signal proportional to the temperature difference between the junctions of two dissimilar metals. One junction (the hot junction) is typically encased in a sensor probe at the point of measurement; the other junction (the cold junction) is connected to the measuring instrument. The measurement instrument measures the voltage signal and the cold junction temperature then computes the temperature.

Type K is made from Chromel (Nickel-Chromium Alloy) for positive lead and Alumel (Nickel-Aluminium Alloy) for negative lead. This is the most commonly used "general purpose" thermocouples. They are available in the ~ -200 °C to +1200 °C range. Thermocouple type K shows by Figure 3.7.

Figure 3.7: Thermocouple Type K

3.3 Data Acquisition

Data acquisition cards are those cards that are used as interfaces between an instrument and a computer in order to capture data for temperature instrumentation. The 1710HG features 16 channels of analog input, two channels of analog output, a 68-pin connector and eight lines of digital I/O.

The PCI-1710 Series are multifunction cards for the PCI bus. Their advanced circuit design provides higher quality and more functions, including the five most desired measurement and control functions: 12-bit A/D conversion, D/A conversion, digital input, digital output, and counter/timer. Figure 3.8 shows PCI-1710HG.

Figure 3.8: PCI-1710 HG

3.4 Equipment Connection

24V power supply is connected to 4-20mA ammeter with series connection. Then negative side of power supply is connected to 2793 decade resistance box (250 ohm). Ammeter also connected to positive temperature transmitter while 2793 decade resistance box is connected to negative temperature transmitter. Next, HART 375 field communicator is connected parallel with temperature transmitter.

Meanwhile, Thermocouple type K is connected to temperature transmitter where positive thermocouple (red wire) is connected to pin number 2 of temperature transmitter, while white wire (-) of thermocouple is connected to pin number 1. The other one of thermocouple is functioned as reference. It connected to pin number 6 (for positive wire) and pin 5(for negative wire).

Temperature will detected by thermocouple and then thermocouple will send the input to temperature transmitter. The input from thermocouple will accepted by temperature transmitter and converted to a 4 to 20mA signal for transmission.

Then, the instrument is connected with the data acquisition (DAQ). DAQ that used in this project is PIC1710 HG which is the sampling of the real world to generate data that can be manipulated by a computer. The connection is show in figure 3.9.

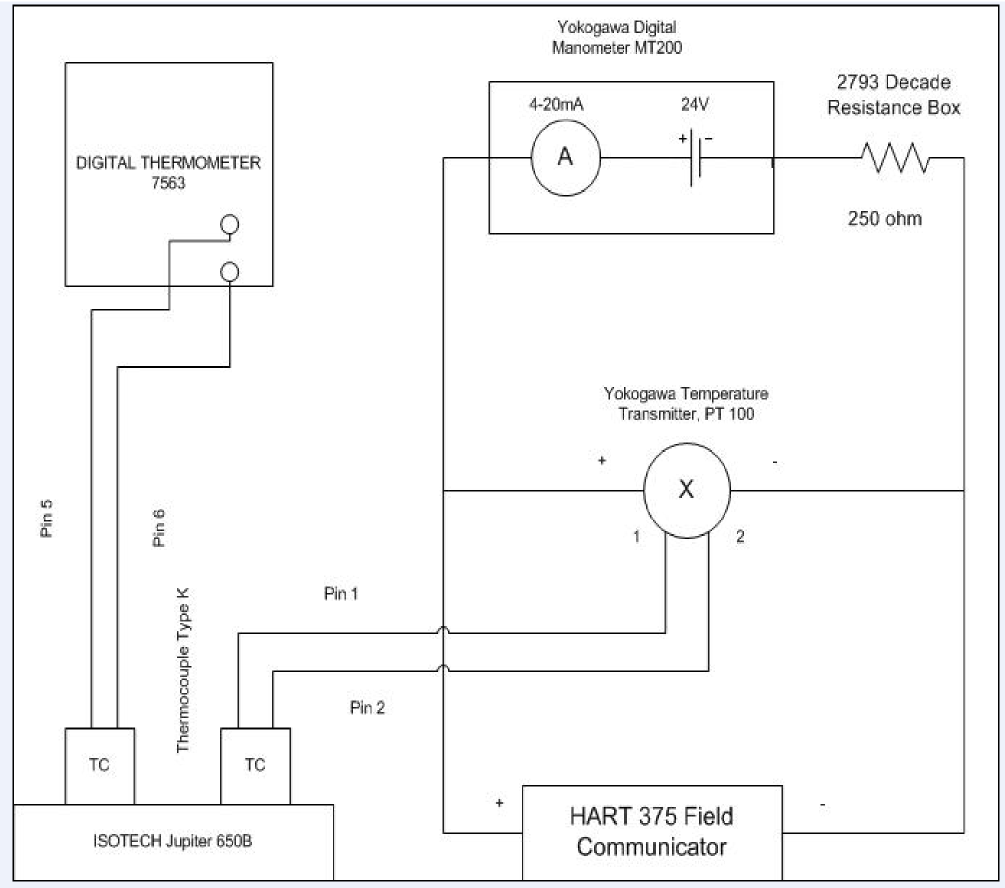

Figure 3.9**:** Equipment Connection

3.5 Block Diagram of the System

Refer to figure 3.10; DAQ (PCI1710HG) is used to interface the instrument with the system that has developed using MATLAB. SIMULINK model is used to run the DAQ and then data from instrument was transferred to MATLAB WORKSPACE. Data that transferred into workspace is captured and displayed at MATLAB GUI. Data that captures is used for calculation of average of actual output and output error. Meanwhile graph for five point calibration of temperature transmitter and output error versus MSU applied value are plotted and saved. Afterward, the data also used for uncertainty evaluation.

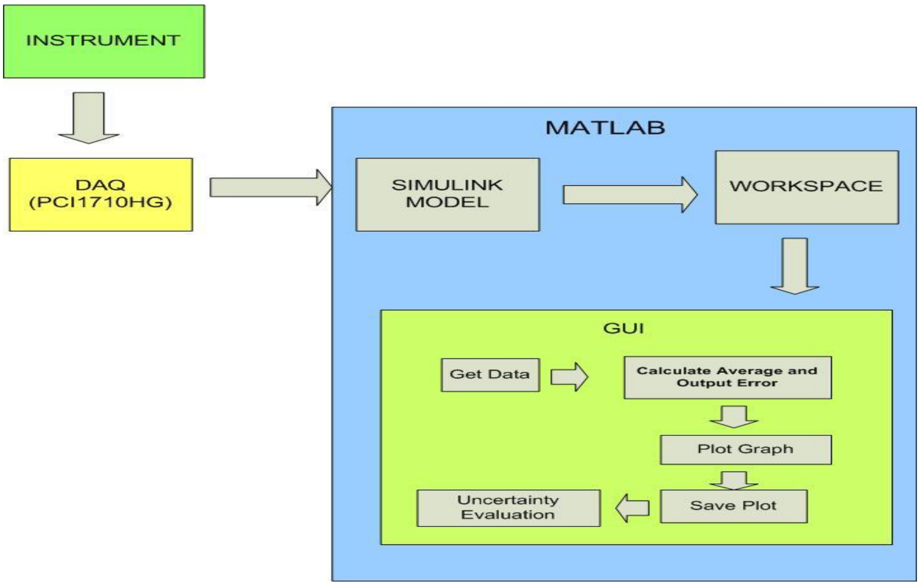

Figure 3.10: Block diagram of the system

3.6 Experiment procedures

Figure 3.11 shows the procedures of temperature instrumentation;

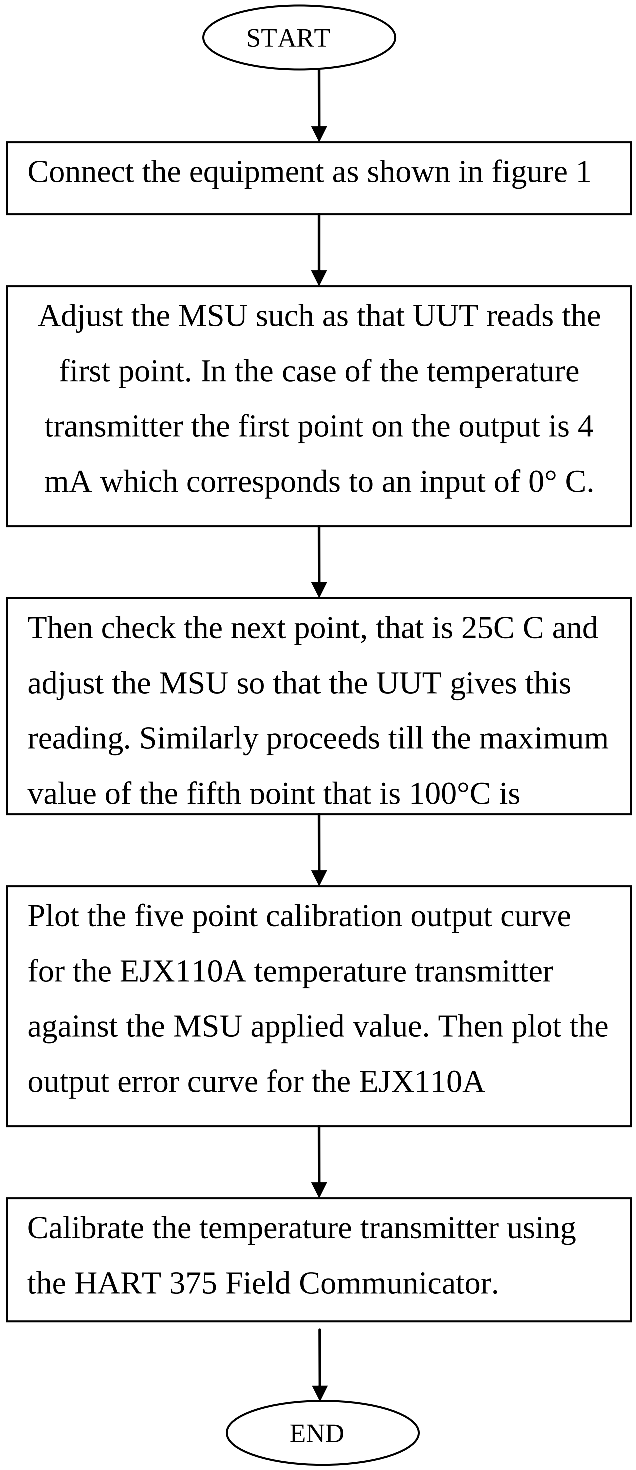

Figure 3.11: Flow diagram of experiment procedures

3.7 DAQ Configuration

3.7.1 Real Time Window Target Setup

Real time window Target enables to run Simulink and Stateflow models in real time on desktop or laptop PC for rapid prototyping or hardware-in-the-loop simulation of control system and signal processing algorithms. A real-time execution can be created and controlled entirely through Simulink. Using Real-Time Workshop, C code can be generated, compiled, and started real-time execution on Windows PC while interfacing to real hardware using PC I/O board (PCI-1710HG). I/O device drivers are included to support an extensive selection of I/O boards, enabling to interface to sensors, actuators, and other devices for experimentation, development, and testing real-time systems. Simulink block diagram can be edited and Real-Time Workshop can be used to create a new real-time binary file. This integrated environment would implement any designs quickly without lengthy hand coding and debugging. Figure 3.12 shows the required product of Real Time Window Target.

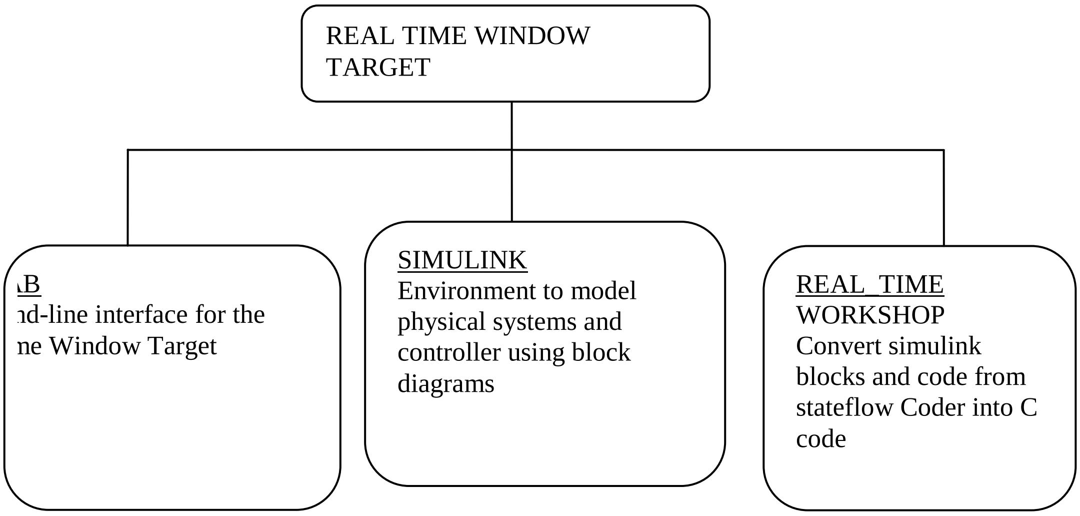

Figure 3.12: Required Products of Real Time Window Target

Real –Time Window Target includes a set of I/O blocks that provide connection between the physical I/O board and real –time model. Hardware –in-the-loop simulations can be ran quickly observed how simulink model responds to real-world behavior. I/O signals can be connected using the block library for operation with numerous I/O boards.

The following types of blocks are included:

- Digital input blocks: connect digital input signal to simulink block diagram to provide logical inputs.
- Digital output blocks: connect logical signals from Simulink block diagram to control external hardware.
- Analog input blocks: enable to use A/D converters that digitalize analog signals for us as inputs to simulinks block diagrams.
- Analog output blocks: enable simulinks block diagram to use D/A converters to output analog signals from simulink model using I/O board(s).
- Counter Input Blocks: enable to count pulses or measure frequency using hardware counters on I/O board(s).
- Encoder Input blocks. Enable to include feedback from optical encoders.

3.7.2 Installation and Configuration

The Real-Time Window Target is a self- targeting system where the host and the targeting computer are the same computer. It can be installed on a PC-compatible computer running Window NT 4.0, Windows 2000, or Windows XP.

3.7.2.1 C Compiler

The Real-Time Windows Target requires one of the following C compilers which are not included in with the Real-Time Windows Target:

- Microsoft Visual C/C++ compiler- Version 5.0, 6.0, 7.0.
- Watcom C/C++ compiler- version 10.6, and 11.0. During installation of watcom C/C++ compiler, a DOS target is specified in addition to windows target to have the necessary libraries available for linking.

After installation, the MEX utility is run to select compiler as the default compiler for building real-time applications.

Real-Time Workshop uses the default C compiler to generate executable code, and the MEX utility uses this compiler to create MEX-files. This procedure is executed in order to select either a Microsoft Visual C/C++ compiler before build an application. Note, the LCC compiler is not supported:

1. mex- setup is typed in the MATLAB window
MATLAB will display the following message:
Please choose your compiler for building external interface
(MEX) files. Would you like mex to locate installed compilers? ([y]/n):
Then a letter "y" is typed

MATLAB will display the following message:

Select compiler:
[1]: WATCOM compiler in c:\watcom
[2]: Microsoft compiler in c:\visual
[0]: None
Compiler:

Next, a number is typed. For example, number 2 is typed to select the Microsoft compiler.

MATLAB will display the following message:

Please verify your choices:

Compiler: Microsoft 5.0

Location: c:\visual

Are these correct? ([y]/n)

Finally, a letter "y" is typed

MATLAB will reset the default compiler and display the message:

The default options file:

"c:\WINNT\Profiles\username\Application Data\MathWorks\MATLAB\mexopts.bat" is being updated.

3.7.2.2 Installing the kernel

During installation, all software for the Real-Time Window target is copied onto hard drive. The kernel is not automatically installed. Installing the kernel sets up the kernel to start running in the background each time when computer is started. The kernel can be installed just after the Real-Time Windows Target has been installed. The installation of the kernel is necessary before a Real-Time Windows Target can be executed:

1. rtwintgt-install is typed in the MATLAB window.

MATLAB will display the following message

```
You are going to install the Real-Time Windows Target kernel.
Do you want to proceed? [y]:
```

2. The kernel installation is continued by typing a letter "y".

MATLAB will install the kernel and display the following message

```
The Real-Time Windows Target kernel has been successfully installed.
```

The computer has to be restart if a "restart" message being displayed.

3. The kernel should be checked whether it was correctly installed. Then, rtwho is typed.

```
MATLAB would display a message similar to
Real-Time Windows Target version 2.5.0 (c) The
Mathworks, inc.1994-2003
MATLAB performance = 100.0%
Kernel timeslice period = 1ms
```

After the kernel being installed, it remains idle, which allows windows to control the execution of any standard Windows application. Standard Windows applications include internet browsers, word processors, MATLAB, and so on. It only during real-time execution of model that the kernel intervenes to ensure that the model is given priority to use the CPU to execute each model updating at the prescribed sample intervals. Once the model update at a particular sample interval completed, the kernel releases the CPU to run any other Windows application might need servicing.

3.7.2.3 Testing the installation

The installation can be tested by running the model rtvdp.mdl. This model does not have any I/O blocks, so that this model can be run regardless of the I/O boards in computer. Running this model would test the installation by executing Real-Time Workshop, the Real –Time Window Target, and the Real-Time Windows Target kernel. After the Real-Time Windows Target kernel being installed, the entire installation can be tested by building and running a Real-Time application. The Real-Time Window Target includes the model rtvdp.mdl, which already has the correct Real-Time Workshop options selected for users:

1. rtvdp is typed in the MATLAB window

The simulink model rtvdp.mdl window will be opened as shown in Figure 3.13.

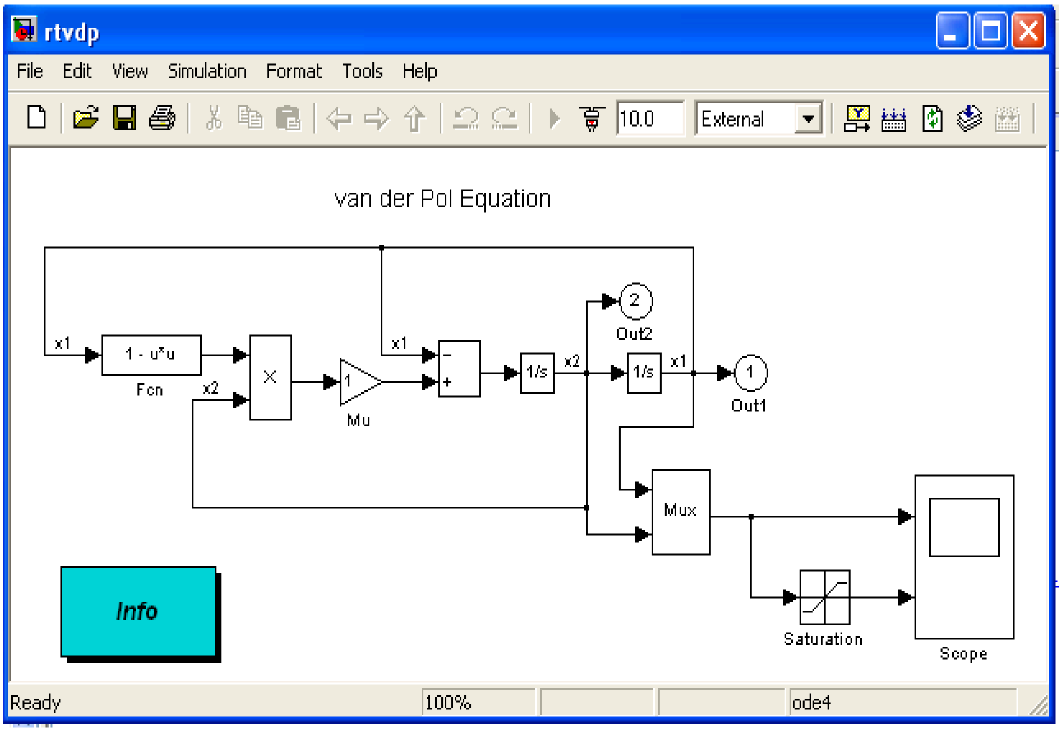

Figure 3.13: Simulink Model rtvdp.mdl

2. From the Tools menu, it should be pointed to Real-Time Workshop and then clicked Build Model. The MATLAB window display the following messages:

```
# # # starting Real-Time Workshop build for model: rtvdp
# # # Invoking Target Language Compiler on rtvdp.rtw
. . .
# # # Compiling rtvdp.c
. . .
# # # Create Real-Time Windows Target module rtvdp.rwd.
# # # Successful completion of Real-Time Workshop build procedure for model:rtvdp
```

3. From the **Simulation** menu, **External** should be clicked and followed by clicking **Connect to target.**

The MATLAB window displayed the following message:

```
Model rtvdp loaded
```

4. Start Real-Time Code is clicked from Simulation menu.

The scope window will display the output signals. After the Real-Time application has been run, Scope Window should indicate such a figure as shown in figure 3.23.

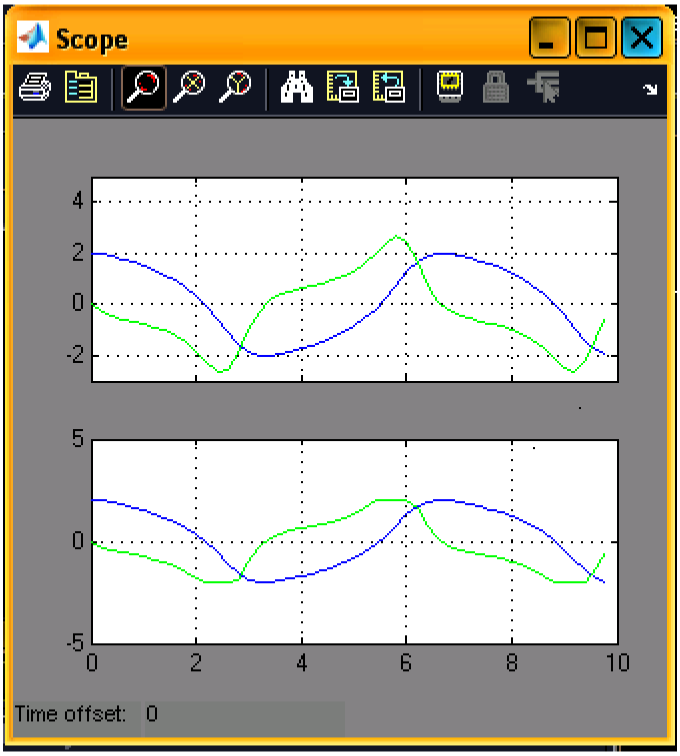

Figure 3.14: Output Signals of rtvdp.mdl

5. From the Simulation menu, after the Stop Real time Code is clicked, the real-time application will stop running and then the Scope window will stop displaying the output signals.

3.7.3 Procedures of Creating Real Time Applications

3.7.3.1 Creating a Simulink Model

This procedure explains how to create a simple Simulink model. This model is used as an example to learn other procedures in the Real-Time windows Target. A Simulink model has to be created before it can be run as simulation or create a real-time application:

1. Simulink is typed in the MATLAB Command Window.

The simulink Library browser window is opened as shown in Figure 3.13.

2. From the toolbar, the Create a new model button is clicked.

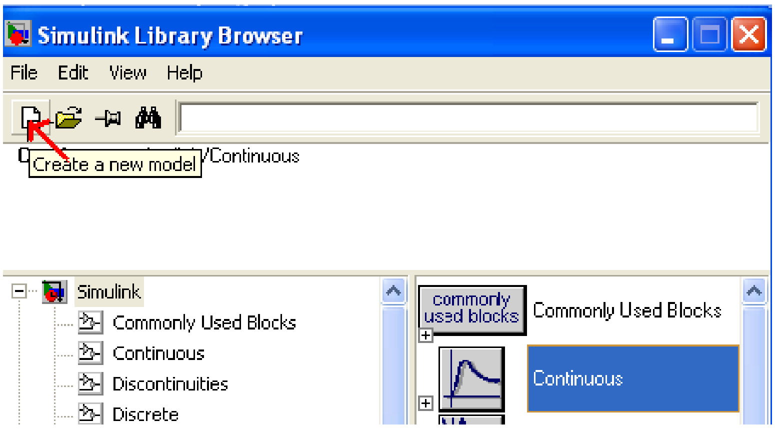

Figure 3.15: create a new model

An empty Simulink window is opened. With the toolbar and status bar disabled, the window looks like the following figure (Figure 3.16).

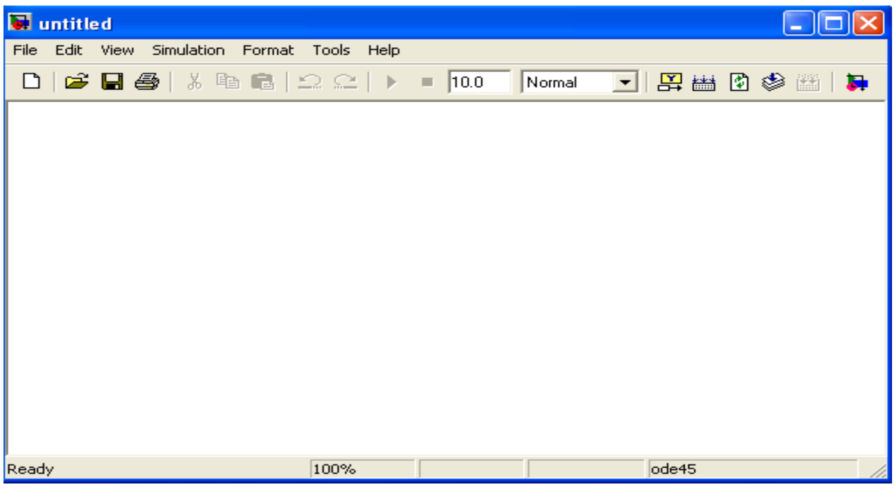

Figure 3.16: Empty simulink model

3. In the simulink Library Browser window, **Simulink** is double-clicked, and then **Sources** is also double clicked. Next, signal Genarator is clicked and dragged to the Simulink window.

Sinks is clicked. **Scope** is clicked and dragged to the Simulink window. Real-Time window Target is clicked. **Analog input** is clicked and dragged to the Simulink window.

4. The **Signal Generator** output is connected to the scope input by clicking-and –dragging a line between the blocks. Likewise, the **Analog Input** input is connected to the connection between Scope and Signal Generator.

5. The Signal Generator block is double clicked. The **Block Parameters** dialog box opens. From the **Wave form** list, square is selected.

In the **Amplitude** text box , 0.25 is entered.

In the frequency text box. 2.5 are entered.

From the **Units** list, hertz is selected.

The Block Parameters dialog box is shown in the Figure 3.17.

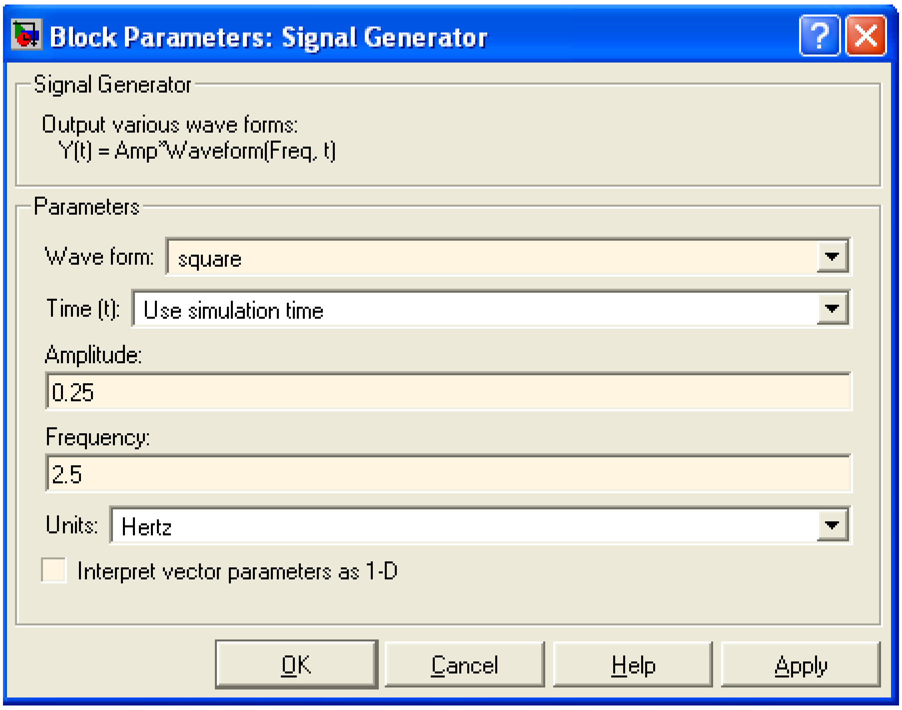

Figure 3.17: Block Parameters of Signal Generator

6. **OK** is clicked.

7. The analog input block is double clicked

   The Block Parameters dialog box will open

8. The Install new baoard button is clicked a board name.

   For example, it should be pointed to Advantech and then clicked PCL1710HG.

9. one of the following is selected:

➢ For an ISA bus board, a base address is entered. This value must match the base address switches or jumpers set on the physical board. For example, to enter a base address of 0x300 in the address box, 300 is typed. The base address also could be selected by selecting check boxes A9 through A3.

➢ For a PCI bus board, the PCI slot is entered or the Auto-detect check box is selected.

10. The Test button is clicked.

    The Real-Time window Target tries to connect to the selected board and the following message would display if successful. It can be seen in Figure 3.18.

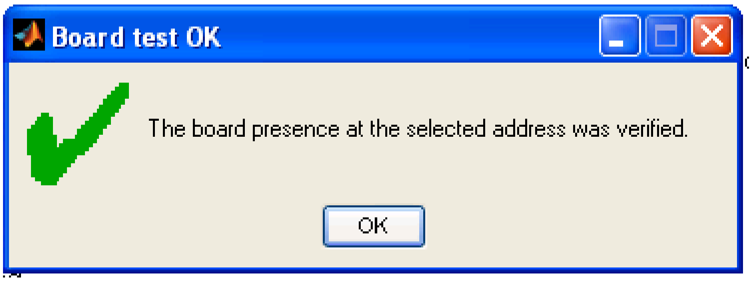

Figure 3.18: Board Test OK

11. On the message box, **OK** is clicked.

12. The same value as entered in the **Fixed step size** box from the **Configuration Parameters** dialog box is entered in the Sample time box. For example, 0.001 is entered.

13. A channel vector that select the analog input channels that are using on this board is entered in the **output Channels** box. the vector can be any valid MATLAB vector form. For example, to select analog output channel on PCI1710HG board.

14. The input range for all the analog input channels that have been entered in the input channels box is chosen from the **Output range** list. For example, with the PCI1710HG, 0 to 5 V is chosen.

15. From the block input signal list, the following options is chosen:

- Volt - expects a value equal to the analog output voltage.

- Normalized unipolar – expects a value between 0 to +1 that is converted to the full range of the output voltage regardless of the output range. For example, an analog output range of 0 to +5 volts and -5 volt would both be converted from values between 0 to +1.

- Normalized bipolar - expects a value between -1 and +1 that is converted to the full range of the output voltage regardless of the output voltage range.

- Raw – Expects a value of 0 to 2n-1. For example, a bit A/D converter would expect a value between 0 to 212-1(0 to 4095).

- The advantage of this method is expected value is always an integer with no round errors.

16. The initial value is entered for each analog output channel that has been entered in the Output channels box. For example, if 1 is entered in the **Output Channels** Box, and an initial value of 0 volts is needed, 0 in entered.

17. The final value is entered for each analog channel that has been entered in the Output channels box. For example, if 1 is entered in the **Output Channels** box, and a final value is needed, 0 is entered. The dialog box would look similar to the figure 3.19 if Volts is chosen.

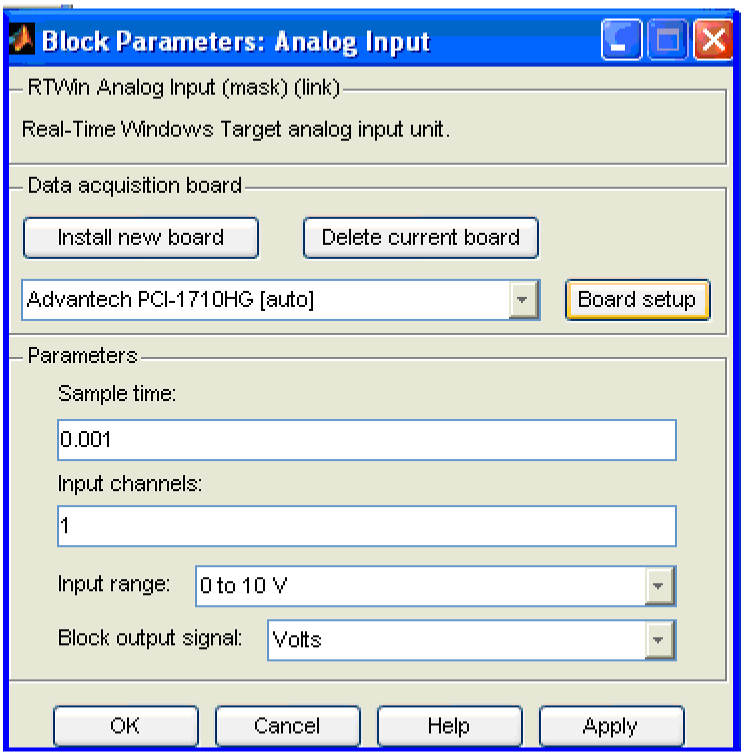

Figure 3.19: Block Parameters of Analog Output

18. One of the following is executed:

- ➢ Apply is clicked to apply the changes to the model and the dialog box is left open
- ➢ OK is clicked to apply the changes to the model and the Block Parameters: Analog Output dialog box will close.

19. Parameters dialog box is closed, and the parameter values are saved with the Simulink model.

20. In the Simulink window, the Scope block is double clicked.
A Scope window will open

21. The **Parameters** button is clicked.

A Scope parameters dialog box will open

22. The **General** tab is clicked. The number of graphs that is needed in one Scope window is entered in the **Number of axes** box. For example, 1 is entered for a single graph. Do not select the **floating scope** check box. In the **Time range** box, the upper value for the time range is entered. For example, 1 second is entered. From the **Tick labels** list, bottom axis only is chosen.

From the Sampling list, decimation is chosen and 1 is entered in the text box.

The Scope parameters dialog box would look like such a figure 3.20 as shown.

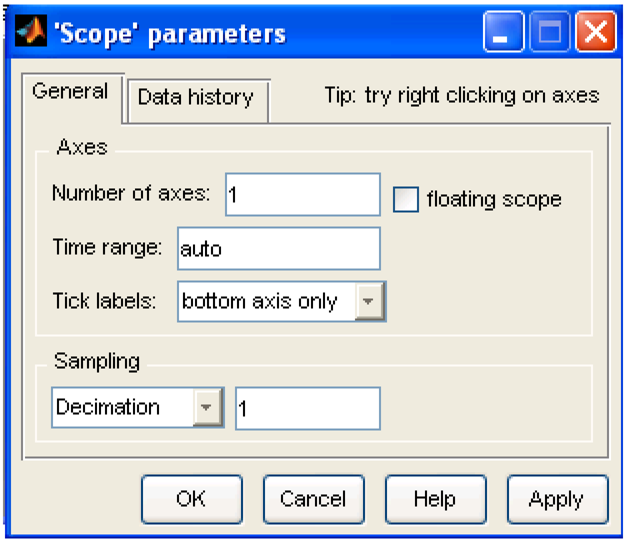

Figure 3.20: Scope parameters Dialog Box

23. One of the following done:

- **Apply** is clicked to apply the changes to the model and the dialog box is left open.
- **OK** is clicked to apply the changes to the model and the Scope parameters: dialog box is close.

24. In the Scope window, it should be pointed to the y-axis as shown in the Figure 3.21, and then right-clicked.

25. **Axes Properties** is clicked from the pop-up menu.

26. The **Scope Properties**: axis 1 dialog box is opened. In **the Y-min** and **Y-max** text boxes, the range for the y-axis is entered in the Scope window. For example, -2 and 2 are entered as shown in the Figure 3.21.

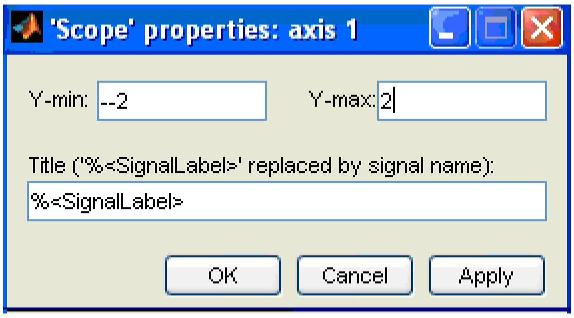

Figure 3.21: Scope Properties: axis 1

27. One of the following is done:

- **Apply** is clicked to apply the changes to the model and the dialog box is left open.
- **OK** is clicked to apply the changes to the model and the Axes Parameters: dialog box is closed.

The completed Simulink block diagram is shown in figure 3.22.

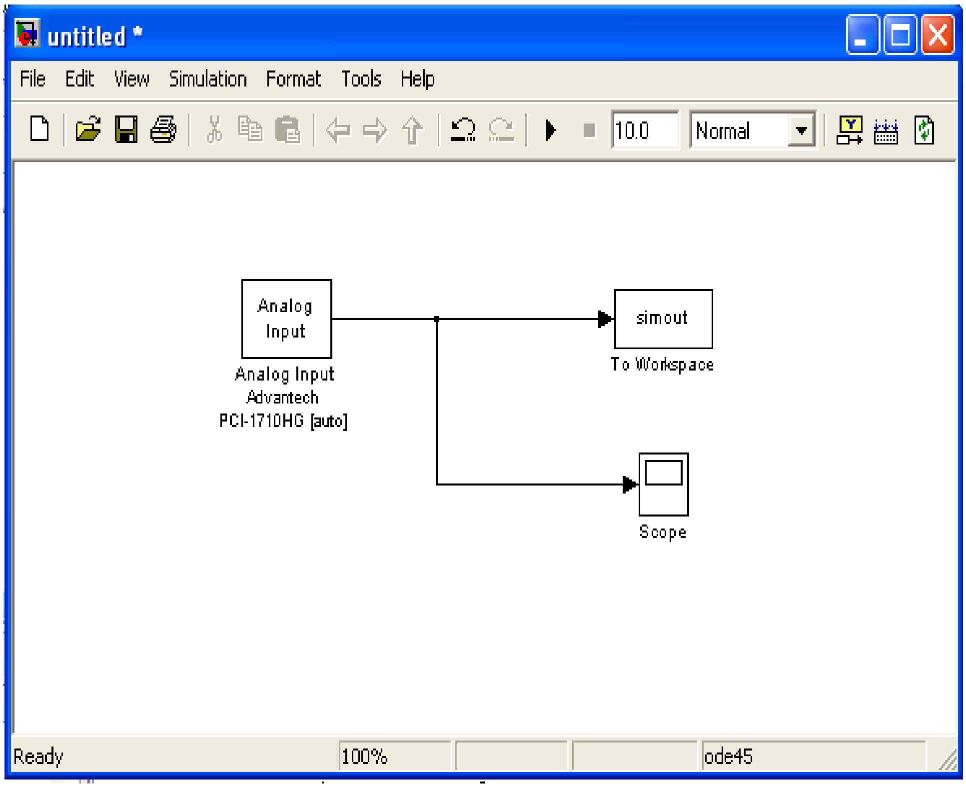

Figure 3.22: Completed Simulink Block Diagram

Save As is clicked from the **File** menu. The **Save As** dialog box is opened. In the **file name** text box, a filename for the Simulink model is entered and **Save** is clicked. For example, rtwin_model is typed
Simulink saved the model in the file , rtwin_model.mdl.

3.7.3.2 Entering Configuration Parameters for Simulink

The configuration parameters give information to Simulink for running a simulation. After created a Simulink model, the configuration parameters could be entered for Simulink.

1. In the Simulink window, **Configuration Parameters** is clicked from the **Simulation** menu. In the **Configuration Parameters** dialog box, the Solver tab is clicked.

   The **Solver** pane will open.

2. In the Start Time box, 0.0 is entered. In the Stop time box, the amount of time that model needs to run is entered. For example, 99999 seconds is entered.

3. From the **Type** list, Fixed-step is chosen. Real-Time Workshop does not support variable step solvers.

4. From the **Solver** list, a solver is chosen. For example, the general purpose solver ode5 (Dormand-Prince) is chosen.

5. In the **Fixed step size** box, a sample time is entered. For example, 0.001 seconds is entered for a sample rate of 1000 samples/second.

6. From the **Tasking Mode** list, singleTasking is chosen. Multitasking is chosen for models with blocks that have different sample times.

   The **Solver** pane would look similar to the Figure 3.23.

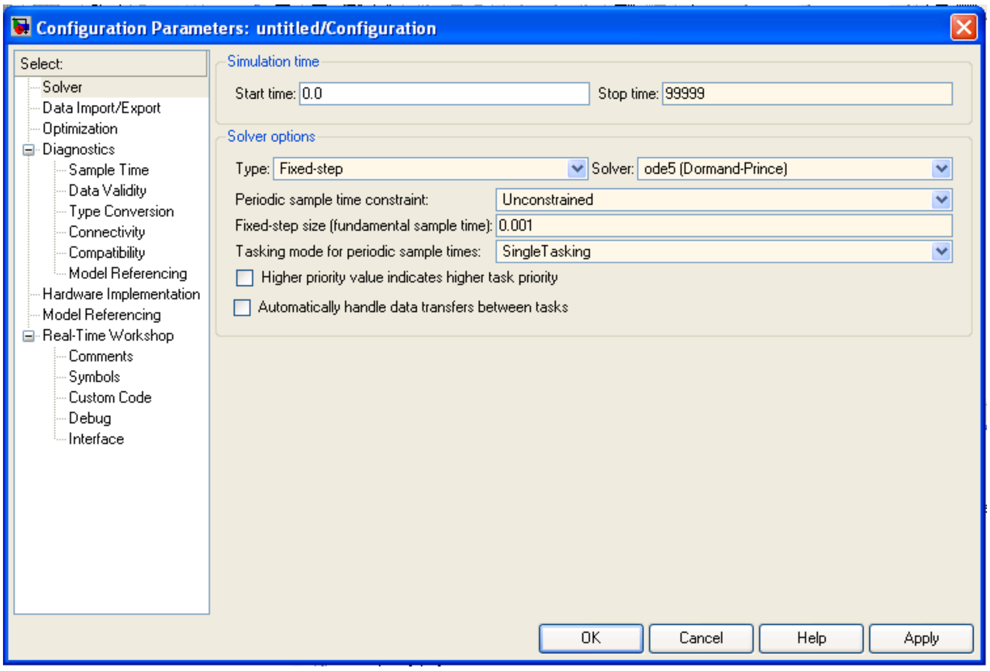

Figure 3.23: Configuration Parameter-Solver

7. One of the following is done:

➢ **Apply** is clicked to apply the changes to the model and the dialog box is left open.

➢ **OK** is clicked to apply the changes to the model and the Configuration Parameters dialog box is closed.

3.7.3.3 Entering Simulation Parameters for Real-Time Workshop

The simulation parameters are used by Real-Time Workshop for generating C code and building a real-time application.

1. In the Simulink window, **Configuration Parameters** is clicked from the **Simulation** menu as shown in Figure 3.24.
2. The **Hardware Implementation** node is clicked.
3. From the **Device type** list, 32-bit Real-Time Windows Target is chosen.

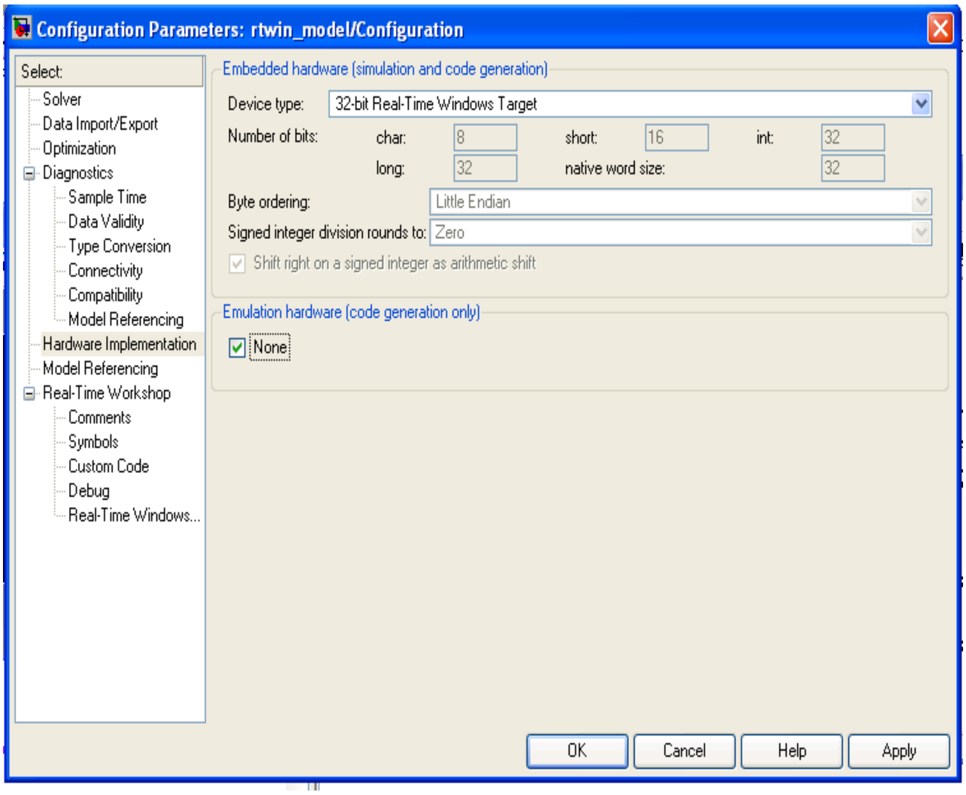

Figure 3.24: Configuration Parameters – Hardware Implementation

4. The **Real-Time Workshop** node is clicked.
   The **Real-Time Workshop** pane will open.

5. In the **Target selection section**, the Browse button is clicked at the RTW system target file list. The **System Target File Browser** will open as shown in Figure 3.25.

6. The system target file is selected for the Real-Time Window Target and **OK** is clicked.

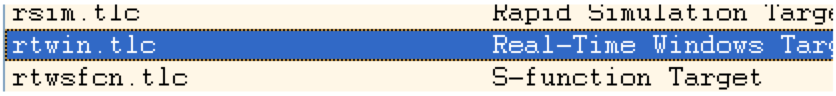

Figure 3.25: System Target File Browser

The system target file rtwin.tlc, the template makefile rtwin.tmf, and the make command make_rtw are automatically entered into the **Real-Time Workshop** pane.

Although not visible in the **Real-Time Workshop pane**, the external target interface MEX file rtwinext is also configured after **OK** is clicked. This allows external mode to pass new parameters to the real-time application and to return signal data from the real-time application. The data is displayed in Scope blocks or saved with signal logging.
The **Real-Time Workshop** pane would look similar to the Figure 3.26

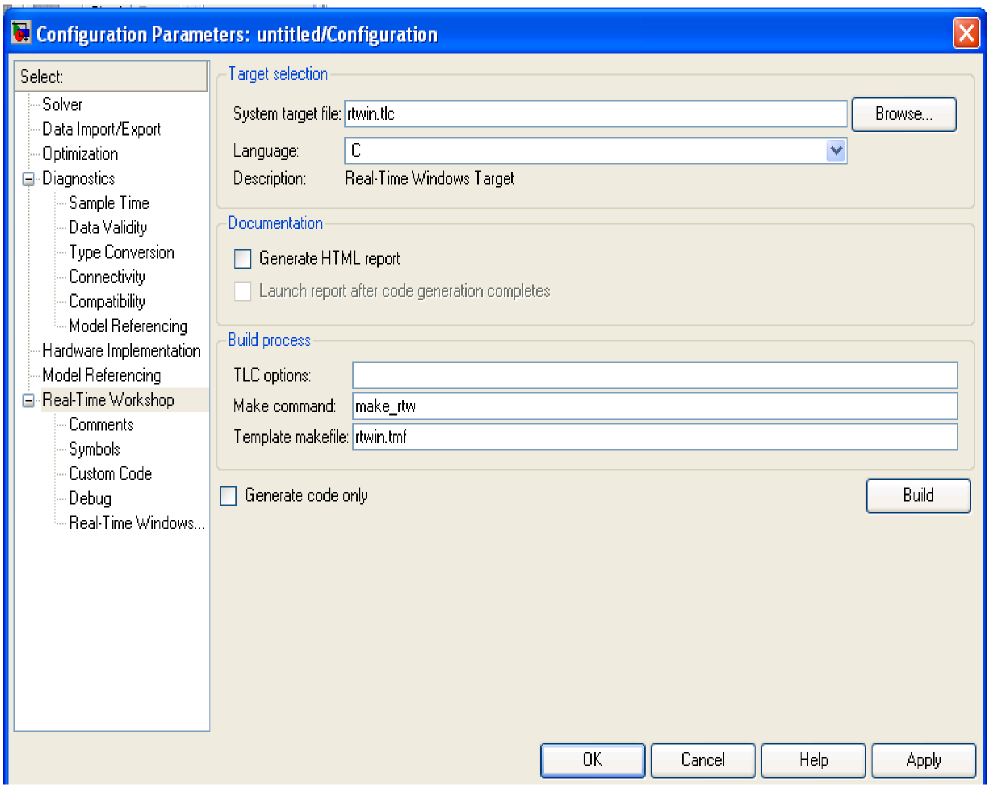

Figure 3.26: Configuration Parameters – Real-Time Workshop

7. After one of the following is done:

- **Apply** is clicked to apply the changes to the model and the dialog box is left open.
- **OK** is clicked to apply the changes to the model and the Configuration Parameters dialog box is closed.

3.7.3.4 Creating a Real-Time Application

Real-Time Workshop generates C code from the Simulink model, and then the Microsoft **Visual Basic C++** compiler and links that C code into a real-time application. After parameters are entered into the **Configuration Parameters** dialog box for Real-Time Workshop, a real-time application could be built.

1. In the Simulink window, and from the Tools menu, it should be pointed to Real-Time Workshop, and then clicked Build Model. The build process does the following:

- Real-Time Workshop creates the C code files

rtwin_model.c and rtwin_model.h

- The make utility make_rtw.exe creates the makefile rtwin_model.mk from the template makefile rtwin.tmf.
- The make utility make_rtw.exe builds the real-time application rtwin_model.rwd using the makefile rtwin_model.mk created above. The file rtwin_model.rwd is binary files that refer to as the real-time application. The real-time application could be run with the Real-Time Windows Target kernel.

2. The Simulink model is connected to the real-time application

   After the real-time application is created, MATLAB could be closed and started again later, and then executable is connected and run without having to rebuild.

3.7.3.5 Running a Real-Time Application

The real-time application is run to observe the behavior of the model in real time with the generated code.

The process of connecting consists of

- Establishing a connection between your Simulink model and the kernel to allow exchange of commands, parameters, and logged data.
- Running the application in real time.

After the real-time application is built, the model could be run in real time.

1. From the **Simulation** menu, **External** is clicked, and then **Connect To Target** is connected from the **Simulation** menu. Also, it could be

connected to the target from the toolbar by clicking . It can be seen in figure 3.27.

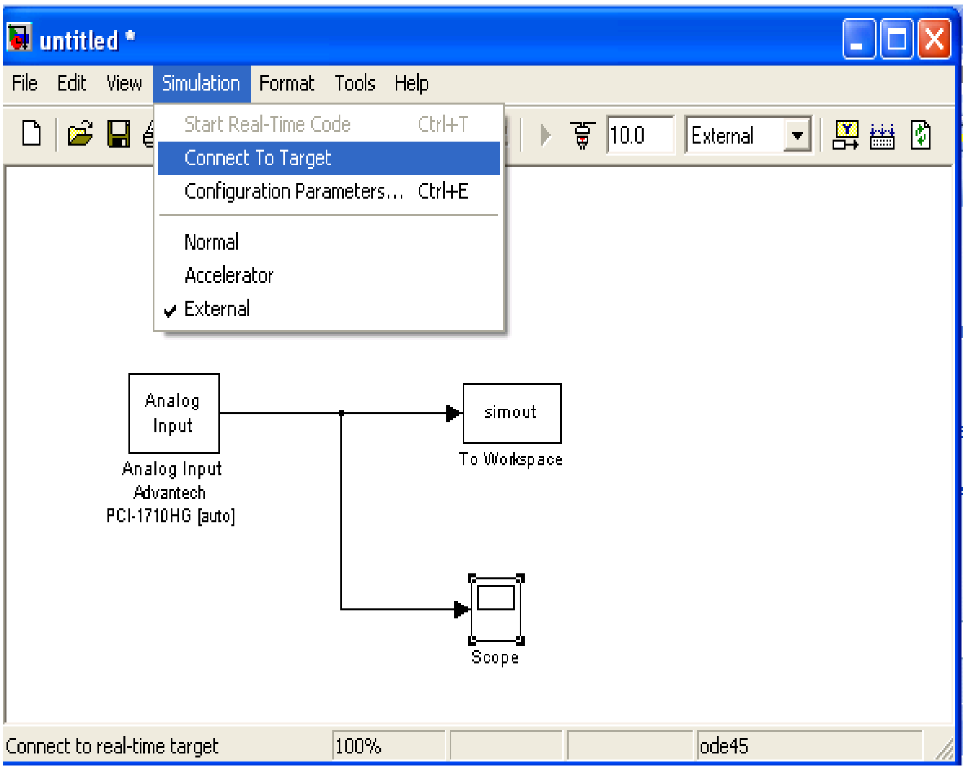

Figure 3.27: Connect to Target from the Simulation menu

MATLAB will display the message

```
Model rtwin_model loaded
```

2. In the Simulation window, and from the Simulation menu, Start Real-Time Code is clicked. The execution also could be started from the toolbar by clicking . It can be seen in Figure 3.28.

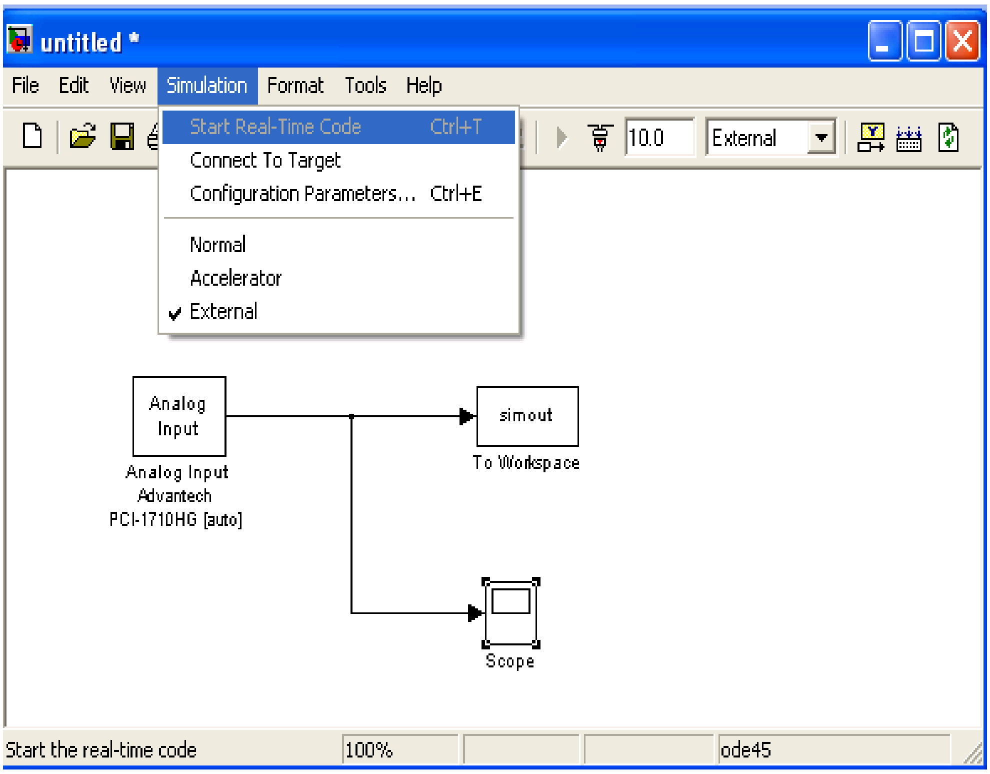

Figure 3.28: Start Real-Time Code from Simulation menu

Simulink runs the execution and plots the signal data in the Scope window. In the model, the Scope window displays 1000 samples in 1 second, increases the time offset, and then displays the samples for the next 1 second.

Note:

Transfer of data is less critical than calculating the signal outputs at the selected sample interval. Therefore, data transfer runs at lower priority in the remaining CPU time after real-time application computations are perform while waiting for another interrupt to trigger the next real-time application update. The result may be a loss of data points displayed in the Scope window.

3. One of the following is done:

- ➢ The executable is let to be run until it reaches the stop time.
- ➢ **Stop Real-time Code** is clicked from the Simulation menu.

The real-time application is stopped.

4. In the Simulation window, **Disconnect From Target** is clicked from the Simulation menu.

5. From the Simulation menu, external is clicked

MATLAB will display the message

```
Model rtwin_model unloaded
```

CHAPTER 4

TEMPERATURE MEASUREMENT SOFTWARE

4.1 **Overview**

The software that use is MATLAB GUI (Graphical User Interface). MATLAB® is a software environment for data acquisition, data analysis, and application development. MATLAB supports the data acquisition and analysis process along with interfacing with data acquisition devices and instruments, analyzing and visualizing the data. It also produces presentation-quality reports to share results with others.

The system that can interface with the component hardware is created. The program can do the evaluation of the uncertainty of temperature measurement. Uncertainty of measurement is the doubt that exists about the result of any measurement. By quantifying the possible spread of measurements, we can say how confident we are about the result. The uncertainty derives from measuring device and from the skill of the person doing measuring.

The system also will able to plot the five point temperature calibrations. Calibration is the process of determining the relation between the output of measuring instrument and the value of the input quantity or attribute, a measurement standard.

Beside that, the system also uses to compare and measure temperature value and actual value and automatically calculate the output error and graphing for output error curve.

4.2 Five Point Calibration Of Temperature Transmitter

For five point calibration, the program is used to calculate the average value of actual UUT output. To get the average, the experiment run for 3 times but the 1^{st} and 2^{nd} run are already set and 3^{rd} run get the data online from the PC. When all the reading is already set in the data, the program will calculate average and plot the five point temperature calibrations versus MSU applied value. By using the average value, output error percentage is calculated automatically. Then by clicking the output error button in MATLAB GUI, the output error curve against MSU applied value is plotted. Below are formula to calculate the average value and output error percentage;

$$\text{i) Average value} = \frac{\text{Tem } 1^{st} \text{ run} + \text{Tem } 2^{nd} \text{ run} + \text{Tem } 3^{rd} \text{ run}}{3} \tag{4.1}$$

$$\text{ii) Output error} = \frac{\text{desired UUT output} - \text{actual UUT output}}{\text{Desired UUT output}} \times 100\% \tag{4.2}$$

4.3 Uncertainty Evaluation

The uncertainty of a measurement is stated by giving a range of values which are likely to enclose the true value. The uncertainty of a measurement is found by repeating the measurement enough times to get a good estimate of the standard deviation of the values. Then, any single value has an uncertainty equal to the standard deviation. However, if the values are averaged, then the mean measurement value has a much smaller uncertainty, equal to the standard error of the mean, which is the standard deviation divided by the square root of the number of measurements.

Standard deviation can be calculated by using the formula;

Std deviation = sqrt(sum((value - mean) ²) / N) or

$$\sigma = \sqrt{\sum_{i-1}^{n}(xi - \bar{x})}^2 \qquad (4.3)$$

Mean = sum (value) / N or $\bar{x} = \frac{1}{N}\sum_{i-1}^{N} xi$

i. Uncertainty Due to Repeatability of The Experiment

For determining the uncertainty contribution due to repeatability experiment, the worst case standard deviation is chosen. The uncertainty is equal with the experimental standard deviation of mean s ($\bar{x}$). This s ($\bar{x}$) is the estimation of the spread of the distribution of the means. For a sample size n=3 the formula for the standard deviation of the mean is given by;

$$S(\bar{x}) = S(X_k) \div \sqrt{n} \qquad (4.4)$$

$$= S(X_k) \div \sqrt{3}$$

Thus, uncertainty, **U1** equal to S ($\bar{x}$) with a degree of freedom ***y1*** = 3-1=2.

ii. Uncertainty Contribution Due to MSU Error

The MSU used in this calibration is the Model: MT220, digital manometer standard. For the 700 kPa range, the accuracy specification for this instrument provided by the manufacturer is the following:

±(0.01% + 0.005% range)

For a maximum reading of 200 kPa and range of 700kPa. Hence the error in MSU = ±((0.0001 x 200) + (0.00005 x 700) V. Therefore the maximum error, a = 0.035 kPa.

The uncertainty contribution due to MSU error, **U2** is given by;

U2 = a/√3 = 0.035/√3 = 0.020207 kPa with a degree of freedom ***y****2* = ∞

iii. Uncertainty Due to UUT Resolution / MSU resolution.

Considering the worst case scenario the maximum resolution is 0.06 kPa. The uncertainty **U3** is calculated as;

U3= $0.06\sqrt{3}$ =0.034641 kPa with $y3 = \infty$

iv. Combined Standard Uncertainty, Uc

The combination standard uncertainty, Uc is determined from the individual uncertainties **U1, U2, U3,** by the following formula;

$$Uc = \sqrt{U1^2 + U2^2 + U3^2} \tag{4.5}$$

The effective degrees of freedom, Ve is given by;

$$Ve = \frac{(Uc^2)^2}{\frac{(U1^2)^2}{V1}} \tag{4.6}$$

The total uncertainty, U at any confidence level is determined using student's t distribution. The coverage factor K is determined from the student table.

The confidence limits are obtain by formula,

$$U = Uc \times K \tag{4.7}$$

4.4 Procedures to Run MATLAB GUI for Plotting Graph

i) The first step to open the MATLAB GUI is to type 'guide' in the MATLAB command window as shown in figure 4.1

.

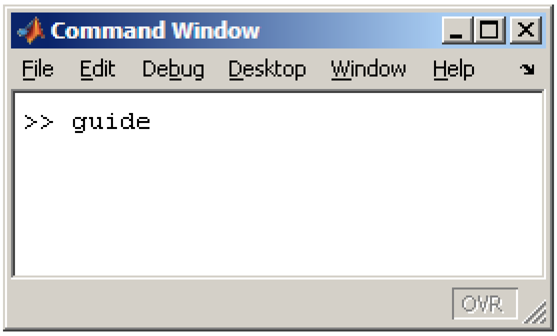

Figure 4.1: Command window (guide)

ii) Choose to open the sample GUI by clicking on “Open Existing GUI”. Click on “Browse” to locate where you saved the GUI files.(Figure 4.2)

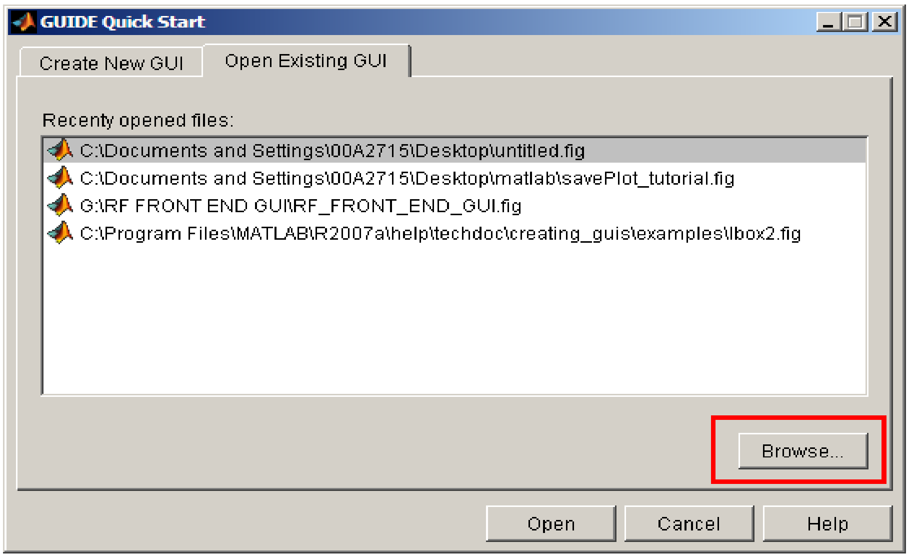

Figure 4.2: Guide Quick Start

iii) To execute the GUI, open the file for five point Calibration of Temperature Transmitter. The figure below will appear. (Figure 4.3)

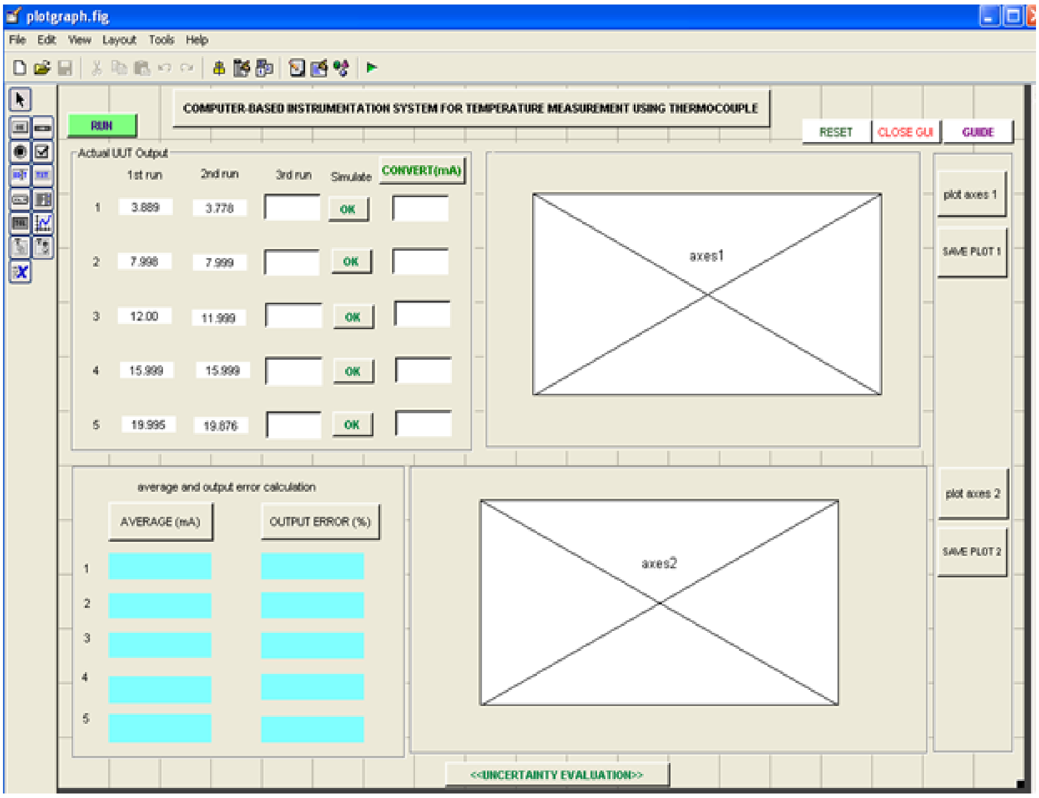

Figure 4.3: GUI for Plot Graph

iv) Then click the green button  at the top of GUI design to run the program. (Figure 4.4)

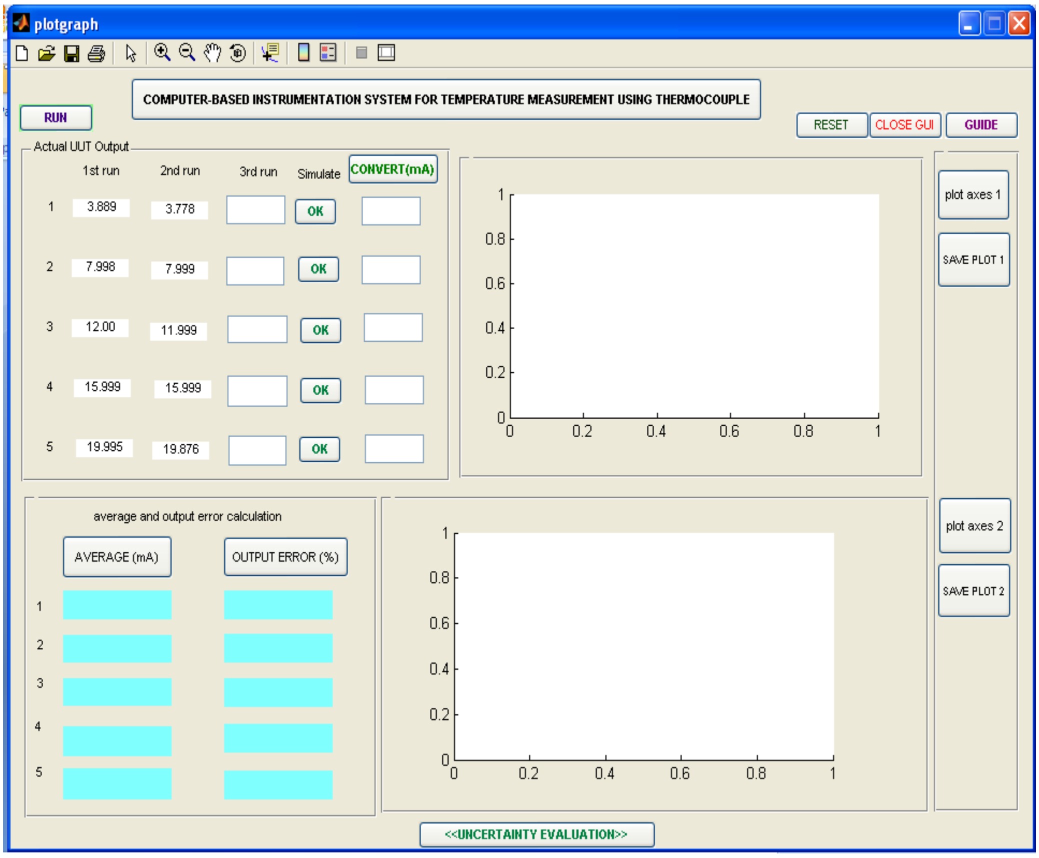

Figure 4.4 Plot Graph GUI

v) There is Guide for a new user about the procedures or step that needs to know to use the system. Click the  button to read the procedures before starting. The dialog box would look similar to the figure 4.5.

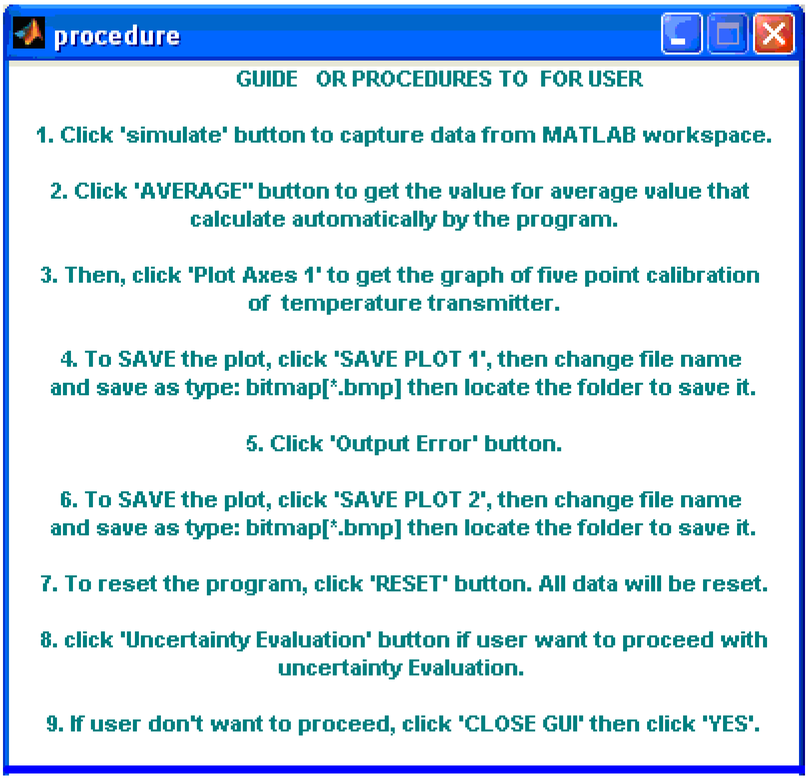

Figure 4.5: Procedures to Use Program

vi) After run the GUI, click  button to get the data from SIMULINK. The value of actual UUT output that taken from experiment will display at the edit text.(Figure 4.6)

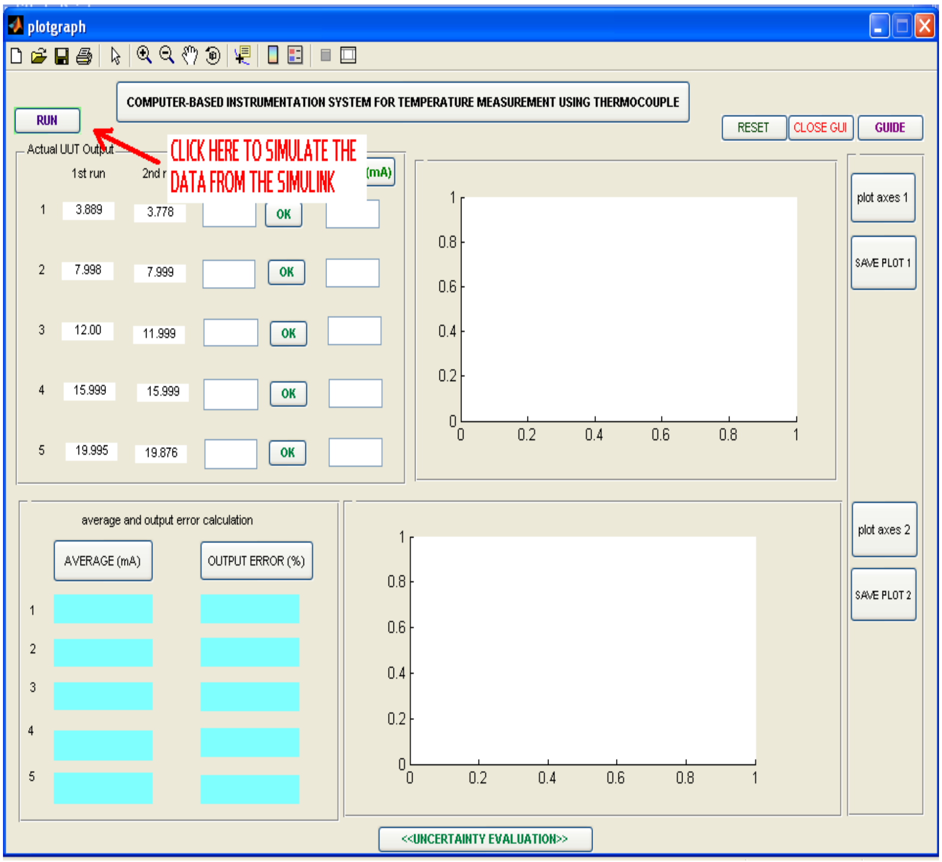

Figure 4.6: Simulate Button

vii) The reading that simulate from DAQ is in voltage, so click the CONVERT(mA) button to change the reading to current.

viii) After finished simulate the actual value, click the AVERAGE (mA) button to get the value for the average value that calculated automatically by the program. Follow by clicking the OUTPUT ERROR (%) button to get the value for the output error that calculated automatically by the program (Figure 4.7)

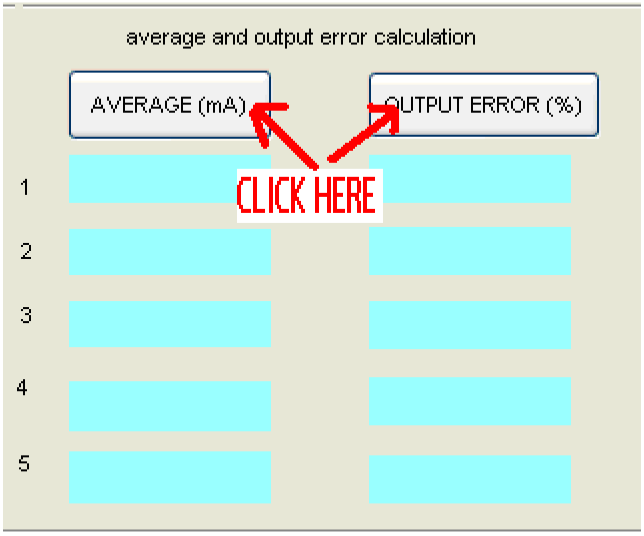

Figure 4.7: Average Button

ix) Next, click plot axes 1 button to get the graph of five point calibration of temperature transmitter vs. MSU applied Value.(Figure 4.8)

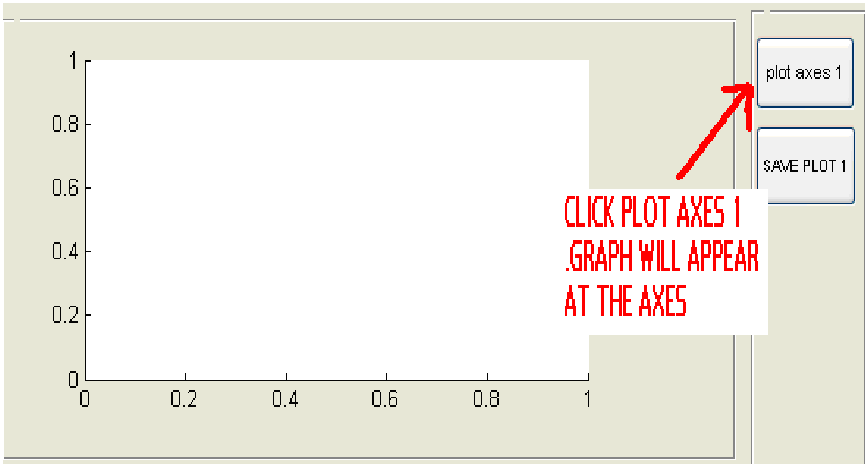

Figure 4.8: Plot axes 1 Button

x) After the graph appear, click SAVE PLOT 1 button to save the graph. The dialog box will look similar to the figure 4.9. Change the file name and save as bitmap (*.bmp) then save at any folder by click “save”.

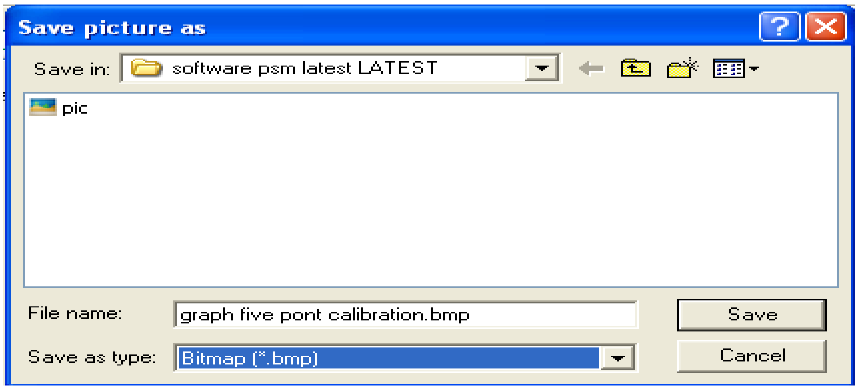

Figure 4.9: Save Picture as

xi) Next, click the plot axes 2 button to get the graph of output error curve against MSU applied value. (figure 4.10)

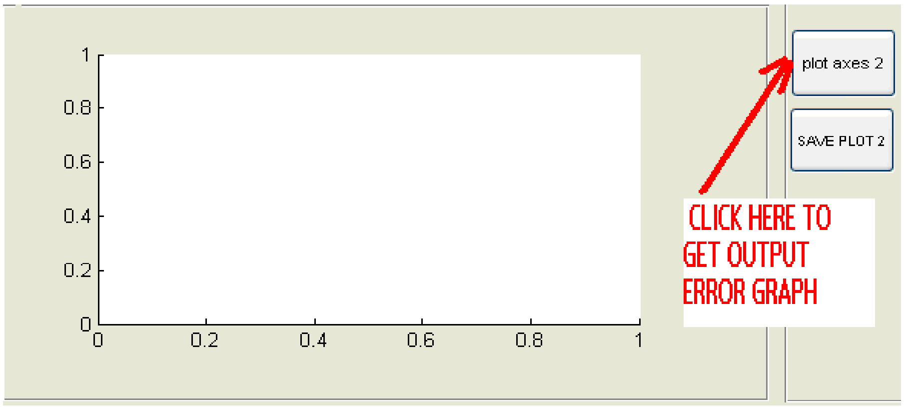

Figure 4.10 Plot axes 2 Button

xii) Then save the graph by clicking SAVE PLOT 2. The step to save is similar with step (viii).

xiii) If the users want to repeat back the experiment, click the RESET button. This will reset or clear all the data and graph.

xiv) If the user wants to leave the system, click CLOSE GUI, the dialog box as figure 4.11 will appear. Click 'YES' to close. But if the user wants to continue with the uncertainty evaluation, click <<UNCERTAINTY EVALUATION>> at the bottom of the GUI.

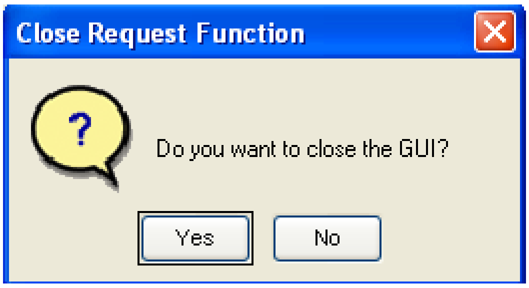

Figure 4.11: Close Request Function

4.5 Procedures to Run MATLAB GUI for Uncertainty

i) To run the GUI for uncertainty evaluation clicked the <<UNCERTAINTY EVALUATION>> button at the buttom of the figure 4.12 to open the separate gui file. This allowed the user sharing data between to separate GUI.

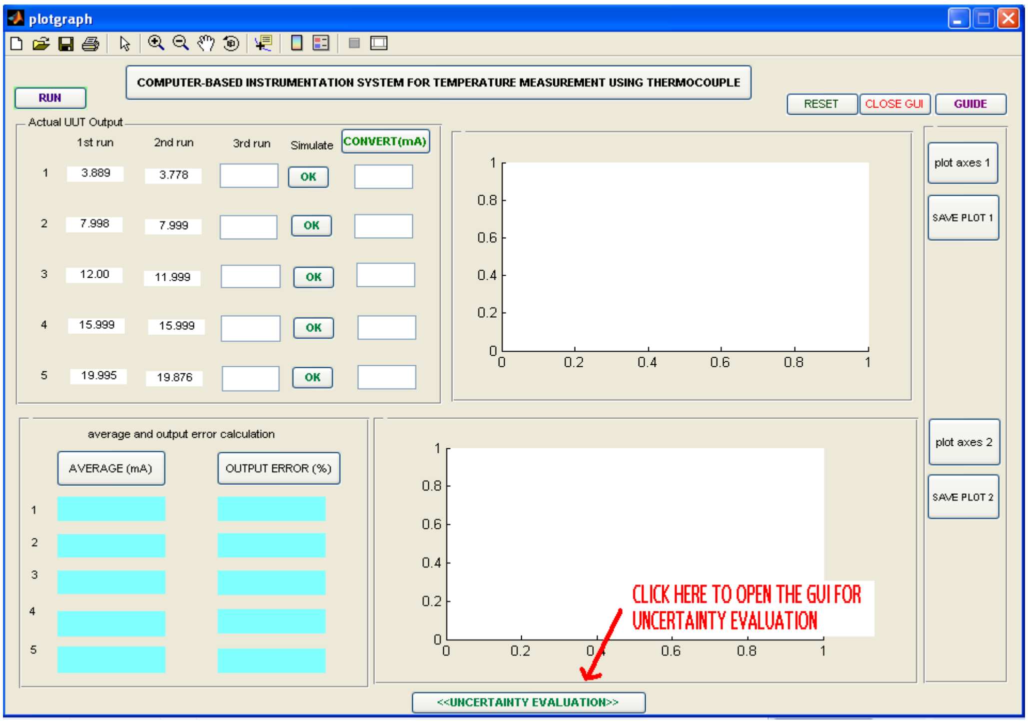

Figure 4.12 Uncertainty Evaluation Button

ii) Figure as below appeared (refer Figure 4.13).

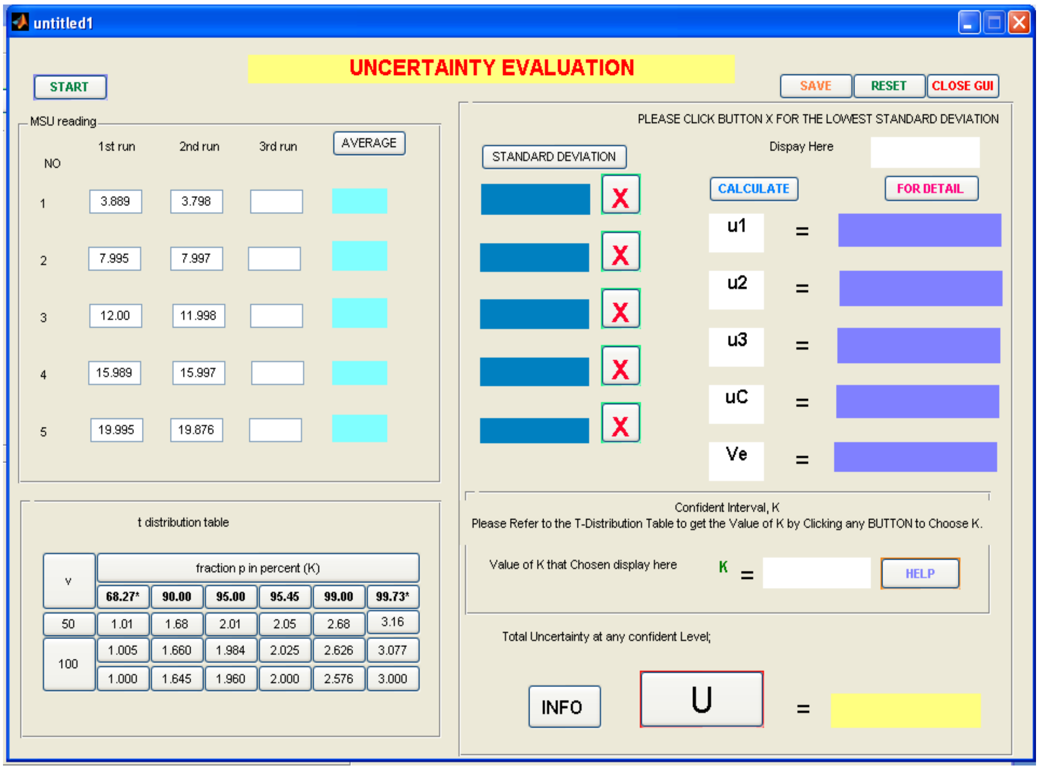

Figure 4.13: Uncertainty Evaluation GUI

iii) Click START to capture data from the Plot Graph GUI. Data for MSU reading will automatically transfer to this GUI. This is use application of sharing data between two separate GUIs. After finished capturing data, click the AVERAGE Then click STANDARD DEVIATION (Figure 4.14).

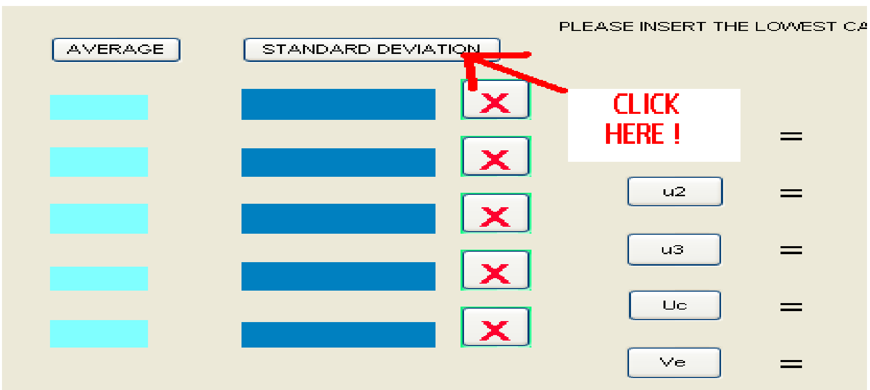

Figure 4.14: Average and Standard Deviation Button

iv) Then choose the lowest case standard deviation by clicking the  button. (Figure 4.15)

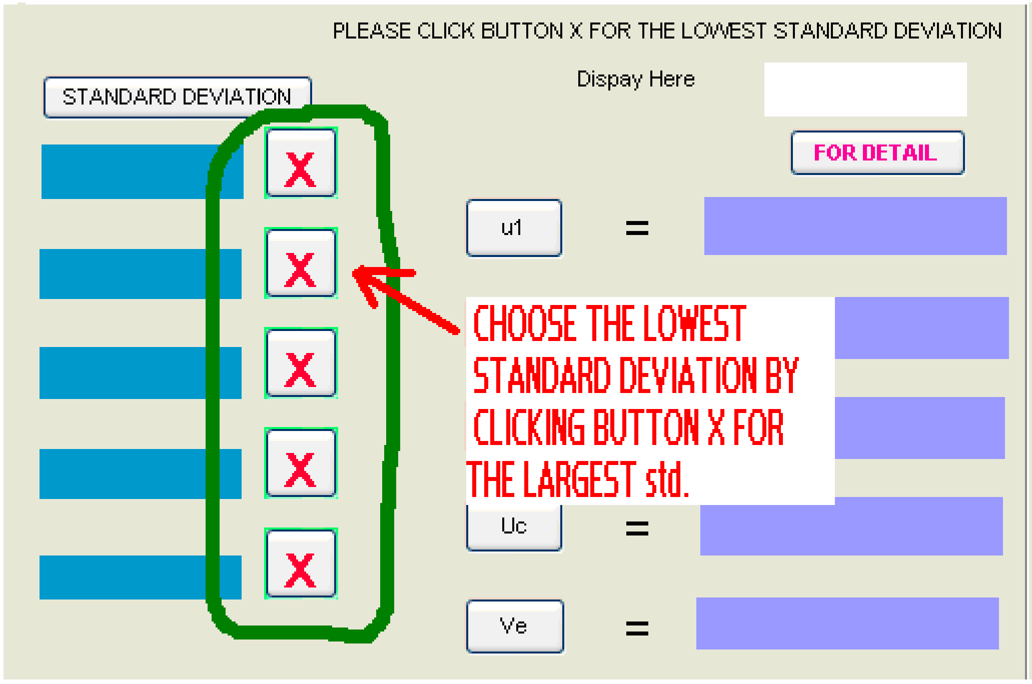

Figure 4.15: Lowest Standard Deviation

v) Then user can see the value that have select will appear at the empty white button as in figure below. (Figure 4.16)

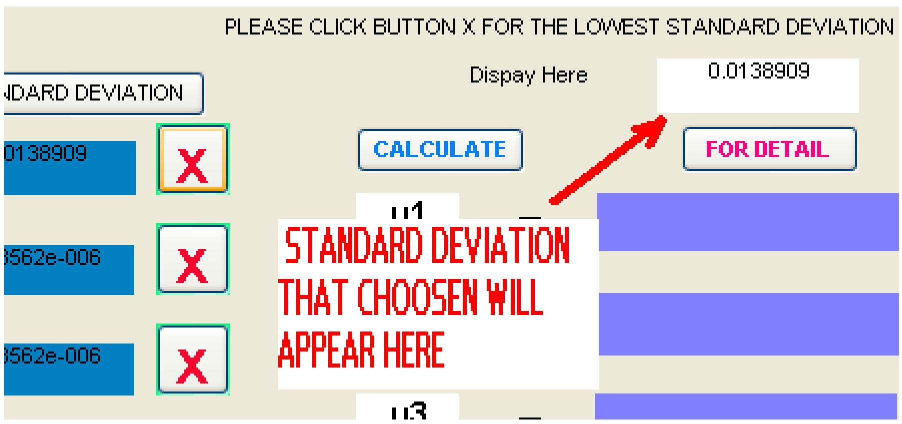

Figure 4.16: Lowest standard Deviation

vi) Now, click the  button to get the value of Uncertainty Due to Repeatability of The Experiment (u1), U2 (Uncertainty Contribution Due to MSU Error), U3 (Uncertainty Due to UUT Resolution / MSU resolution), Uc (Combined Standard Uncertainty) and Ve (The effective degrees of freedom). (Figure 4.17)

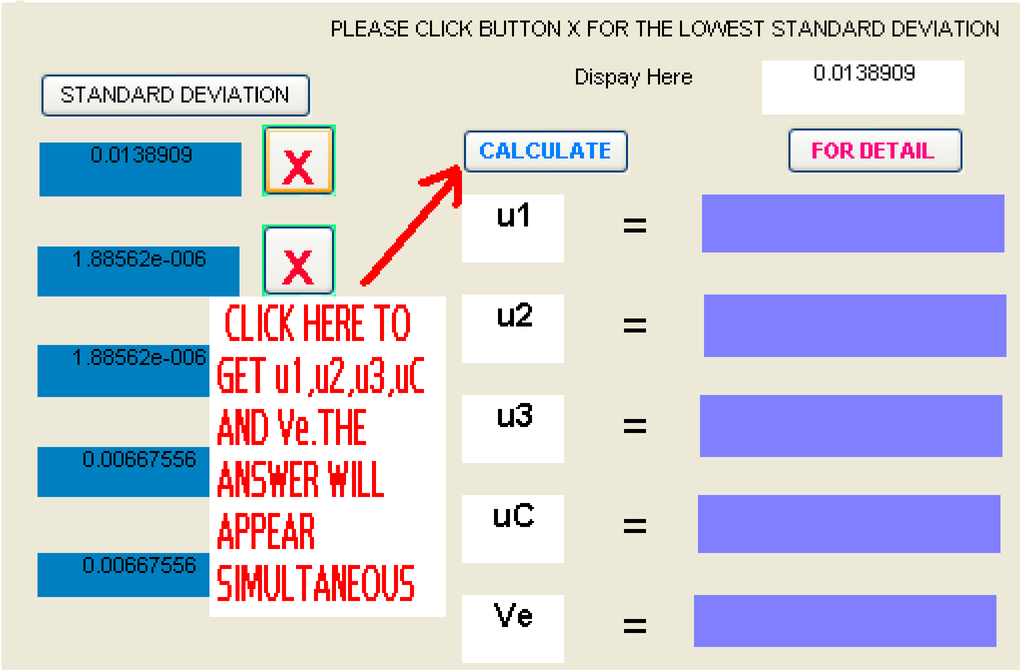

Figure 4.17: Calculate Button

vii) The total uncertainty at any confidence level is determined using t-Distribution table. Thus to choose the value of K, user must choose the value that lower than Ve value (k< Ve). For Ve than greater than 100, choose K for v=100.User can choose by click any K value that suitable at t-Distribution table. Then it will appear as below. (Figure 4.18)

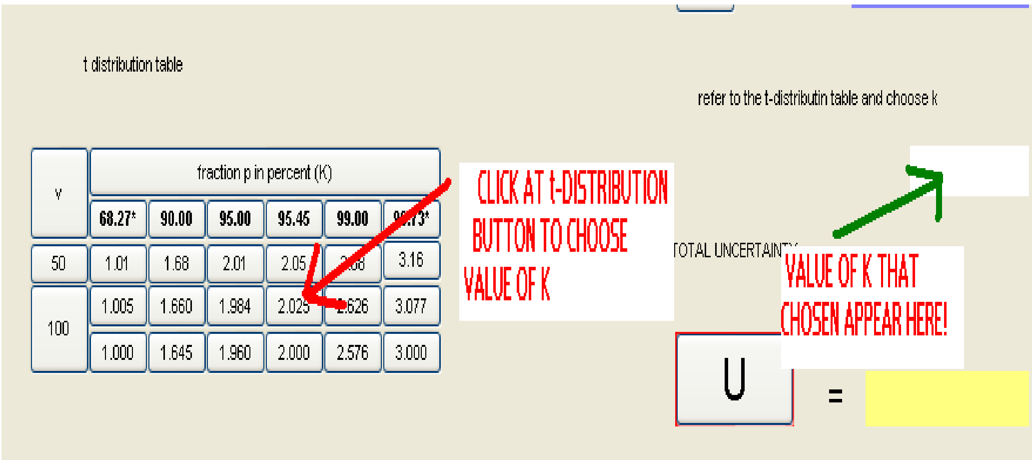

Figure 4.18: T-Distribution Button

viii) Lastly, click U button to get total uncertainty. (Figure 4.19)

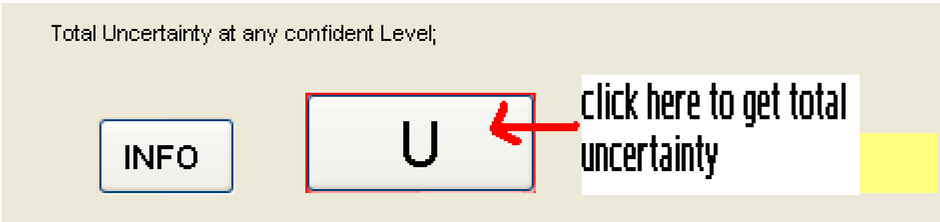

Figure 4.19: Total uncertainty

ix) After complete all of the calculation, user can save the result in EXCEL .click SAVE button and MATLAB GUI will link to excel. Excel booklet will appear as in Figure 4.20. This will allow user to key in all the data.

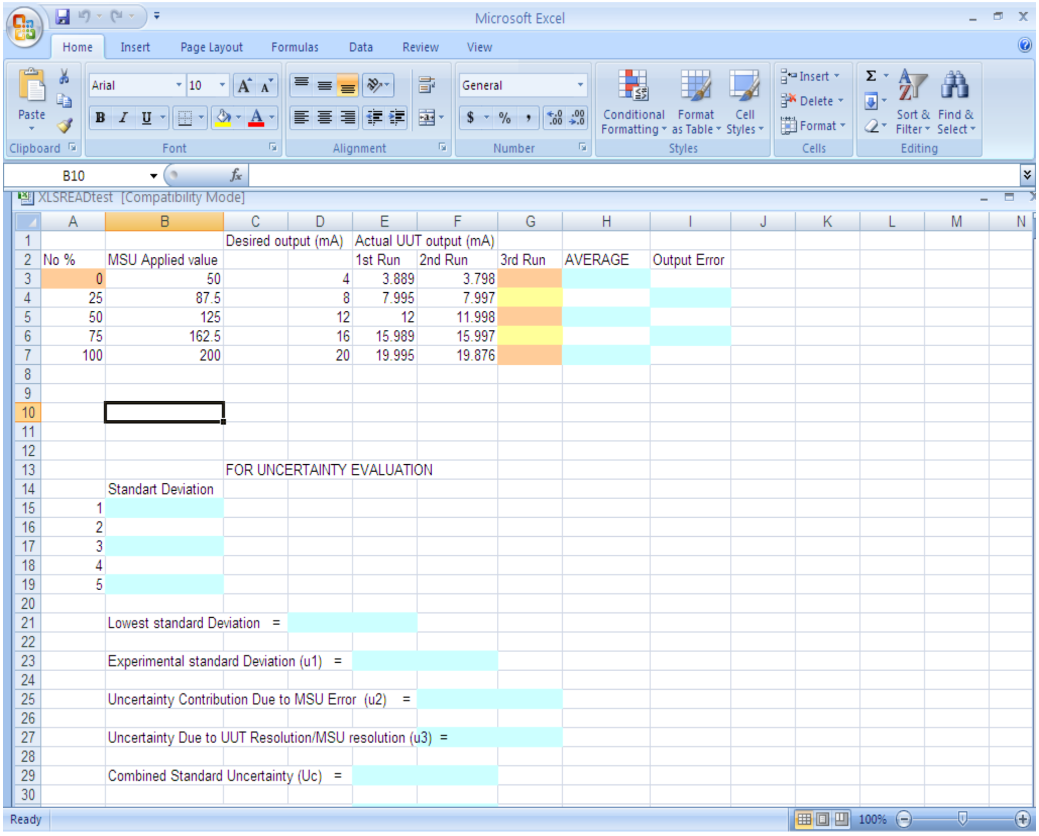

Figure 4.20: Excel Link

CHAPTER 5

RESULT AND ANALYSIS

5.1 Result For Five – Point Calibration Of Temperature Transmitter

Table 5.1 shows the average of actual UUT output and output error from the manual calculation. As the reading of 1st run and 2nd run is fixed, that mean the value of average and output error was depend on 3rd run reading. As the actual UUT outputs reading get closer to the desired UUT output, the output error decreased.

Table 5.1: Five – Point Calibration of Temperature Transmitter

N0 %	MSU applied Value (°C)	Desired UUT output (mA)	Actual UUT output (mA)				Output error (%)
			1st run	2nd run	3rd run	average	
0	50	4	3.889	3.798	3.996	3.894	2.6500
25	87.5	8	7.998	7.999	7.994	7.997	0.0375
50	125	12	12.000	11.999	11.997	11.998	0.0167
75	162.5	16	15.999	15.999	15.998	15.999	0.0313
100	100	20	19.995	19.876	19.857	19.909	0.4550

5.2 Calcu

5.2 Calculation For Five – Point Calibration Of Temperature Transmitter

i. The desired output for 4-20 mA range is calculated based on the 50-100°C range using below equation:

$$Desired\ ouput = \frac{x}{100}(URV - LRV) + LRV$$

Where; X= ith point

URV= Upper range value

LRV= Lower range value

Thus;

For $x = 0$,

Desire output = x/100(URV – LRV) + LRV

= 0/100(20mA – 4mA) + 4mA

= 4mA

For $x = 25$;

Desire output = x /100(URV – LRV) + LRV

= 25/100(20mA – 4mA) + 4mA

= 8mA

For $x = 50$

Desire output = x /100(URV – LRV) + LRV

= 50/100(20mA – 4mA) + 4mA

= 12mA

For $x = 75$

Desire output = x /100(URV – LRV) + LRV

=75/100(20mA – 4mA) + 4mA

= 16mA

For $x = 100$

Desire output = x /100(URV – LRV) + LRV

=100/100(20mA – 4mA) + 4mA

= 20mA

ii. The average actual UTT output can calculated based on the equation below;

$$\text{Average value} = \frac{\text{tem } 1^{st} \text{ run} + \text{tem } 2^{nd} \text{ run} + \text{temp } 3^{rd} \text{ run}}{3}$$

For $x = 0$;

$$\text{Average value} = \frac{3.889 + 3.798 + 3.996}{3}$$

= 3.894 mA

For $x = 25$;

$$\text{Average value} = \frac{\mathbf{7}.998 + 7.999 + 7.997}{3}$$

= 7.997 mA

For $x = 50$;

$$\text{Average value} = \frac{12.000 + 11.998 + 11.997}{3}$$

= 11.998 mA

For $x = 75$;

$$\text{Average value} = \frac{15.999 + 15.999 + 15.998}{3}$$

= 15.999 mA

For $x = 100;$

Average value = $\frac{19.995+ 19.876+ 19.857}{3}$

= 19.909 mA

iii. The output error can calculated based on the equation below:

$$\text{Output Error} = \frac{\text{desired UUT output - average UUT output}}{\text{Desired UUT output}} \times 100\%$$

Thus for 50°C;

Output Error = (4-3.894)/$\sqrt{4} \times$ 100%

= 2.65%

For 87.5°C;

Output Error = (8 - 7.995) $\sqrt{8} \times$100%

=0.0625%

For 125.0°C;

Output Error = (12 - 11.998)/ $\sqrt{12} \times$100%

=0.0167%

For 162.5°C;

Output Error = (16-15.995)/ $\sqrt{16} \times$100%

= 0.0313%

For 200.0°C;

Output Error = (20-19.909)/ $\sqrt{20} \times$100%

=0.455%

5.3 Result using Software

5.31 The average actual UTT output can show below. (Figure 5.1)

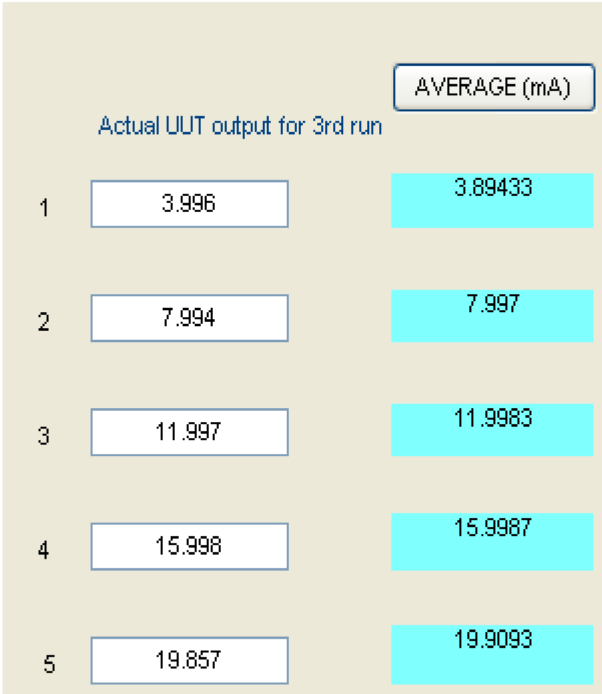

Figure 5.1: Average actual UUT Output

5.3.2 The graph for five point calibration of temperature against MSU applied value from GUI.

Figure 5.2 shown the graph for five point calibration of temperature transmitter or average output versus MSU applied Value. Average output is proportionally with MSU applied value. As the MSU value is constant, when MSU applied value increase the value of average output also increased.

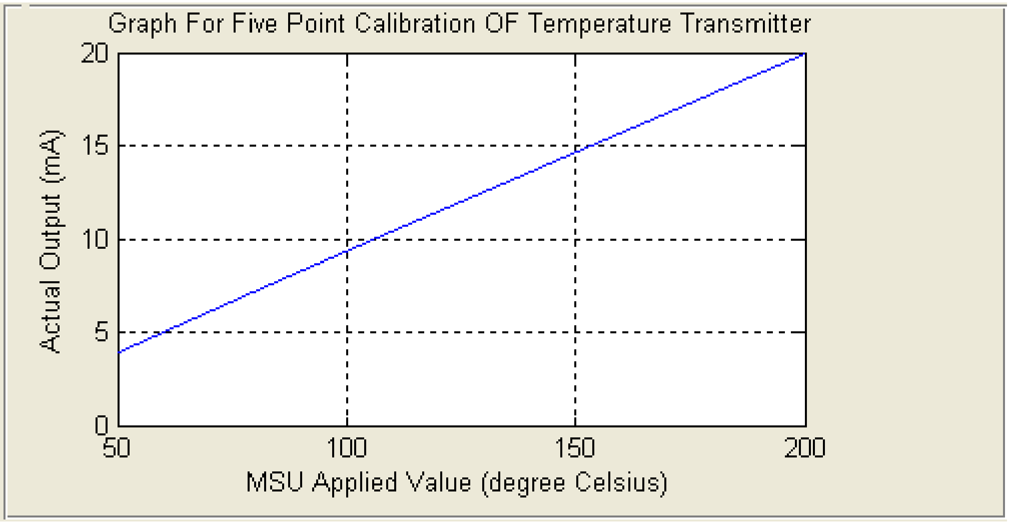

Figure 5.2: Graph for Five Point Calibration of Temperature Transmitter

5.3.3 The output error calculated using GUI (Figure 5.3)

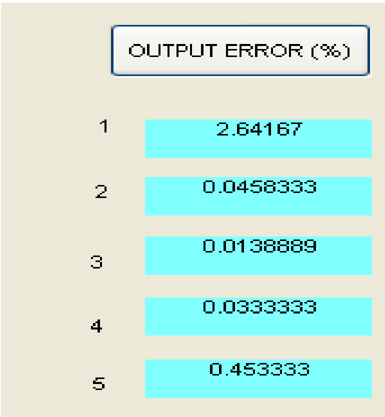

Figure 5.3: Output Error

5.3.4 The graph for output error curve against MSU applied value from GUI

Figure 5.4 shows the graph for output error versus MSU applied value. The shape of the graph is curved because the output error is depending on the reading from the experiment. Less error of the reading means more precision the experiment.

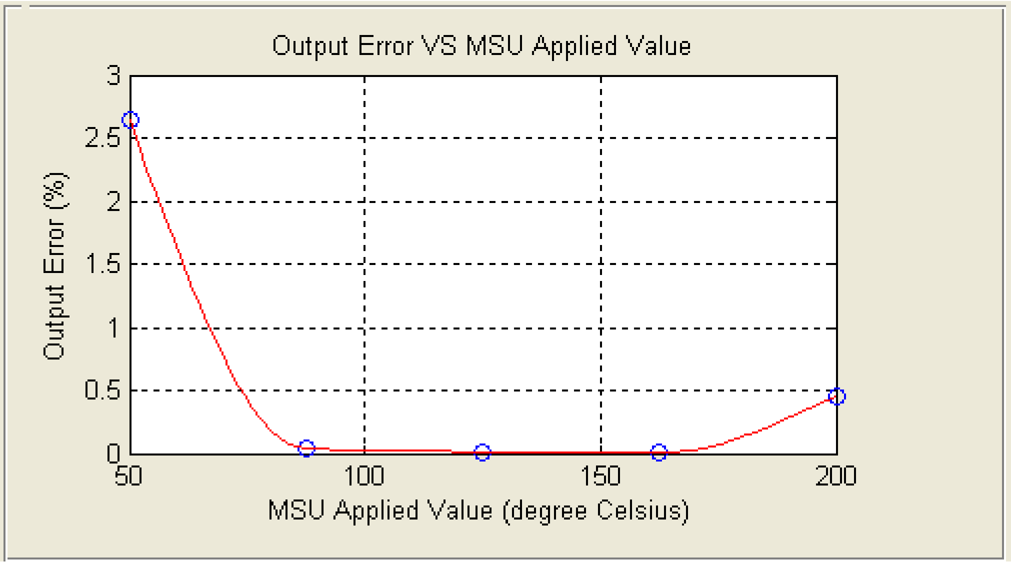

Figure 5.4: Graph Output Error Versus MSU applied Value

5.4 Comparison between manual calculation and by using software

Table 5.2 shown the comparison between manual calculation and calculation using the system that have been developed. The value of actual and output error from the manual calculation is almost same with the calculation using system. So that, it proved that the system are done the correct calculation.

Table 5.2: Comparison between manual and Software

No	Manual		software	
	Actual UUT output average(mA)	Output Error (%)	Actual UUT output average (mA)	Output Error (%)
1	3.894	2.65	3.89433	2.64167
2	7.997	0.0375	7.997	0.045833
3	11.998	0.0167	11.9983	0.013889
4	15.999	0.0313	15.9983	0.033333
5	19.909	0.4550	19.9093	0.4550

5.5 Result For Uncertainty Evaluation

Table 5.3 shows the result for uncertainty from the manual calculation. Using average output, standard deviation is calculated for each run. The highest value of standard deviation will use to calculate the uncertainty. Refer to the Table 5.3; the highest value of standard deviation is 1.3890×10^{-2}.

Table 5.3: uncertainty due to repeatability of the experiment

No	Desired UUT output (mA)	Actual UUT output (mA)				Standard deviation
		1st run	2nd run	3rd run	average	
1	4	3.889	3.798	3.996	3.894	1.3890e-2
2	8	7.995	7.997	7.994	7.995	3.5355e-6
3	12	12.000	11.998	11.997	11.998	3.5355e-6
4	16	15.989	15.997	15.999	15.995	3.9598e-5
5	20	19.995	19.876	19.857	19.909	6.7663e-3

5.6 Calcul

5.6 Calculation for Uncertainty Evaluation

i. Calculation for standard deviation Refer to equation (4.3)

For desired UUT = 4mA;

S ($\bar{x}$) =$\sqrt{0.5}$ × (3.889-3.894)^2 + (3.798-3.895)^2 + (3.996-3.895)^2]

=0.01389 =1.3890e-2

For desired UUT = 8mA;

S ($\bar{x}$) = $\sqrt{0.5}$ × [(7.995- 7.997)^2 + (7.998-7.995)^2 + (7.994-7.995)^2]

=3.5355e-6

For desired UUT = 12mA;

S ($\bar{x}$) = $\sqrt{0.5}$ × [(12.00-11.998)^2 + (11.998-11.998)^2 + (11.997-11.998)^2]

=3.5355e-6

For desired UUT = 16mA;

S ($\bar{x}$) = $\sqrt{0.5}$ × [(15.989-15.995)^2 + (15.997-15.995)^2 + (15.999-15.995)^2]

=3.9598e-5

For desired UUT = 20mA;

S ($\bar{x}$) = $\sqrt{0.5}$ × [(19.995-19.909)^2 + (19.876-19.909)^2 + (19.857-19.909)^2]

=6.7663e-3

ii. Uncertainty Due to Repeatability of The Experiment

Choose the lowest standard deviation $s(x_k)$ or the highest value of standard deviation.

From the table the, choose 1.3890e-2 as lowest case. (Refer to equation (4.4)).

Where S $(\bar{x})$ = $s(x_k) \div \sqrt{n}$

$= s(x_k) \div \sqrt{3}$

$= 0.01389 \div \sqrt{3}$

$= 8.0194 \times 10^{-3}$

Thus, uncertainty, **U1** equal to S $(\bar{x})$ =**8.0194e-3** kPa with a degree of freedom ***y1*** = 3-1=**2**.

iii. Uncertainty Contribution Due to MSU Error

U2 = a/$\sqrt{3}$ = 0.035/$\sqrt{3}$ = 0.020207 kPa with a degree of freedom ***y****2* = ∞

iv. Uncertainty Due to UUT Resolution / MSU resolution.

U3= 0.06$\sqrt{3}$ =0.034641 kPa with ***y****3*= ∞

v. Combined Standard Uncertainty, Uc

When U1=8.0194e-3 kPa, U2= 0.020207 kPa, U3= 0.034641 kPa. Refer to equation (4.5).

$$Uc = \sqrt{(8.0194e-3)^2 + (0.02027)^2 + (0.034641)^2}$$

= **0.040898 kPa**

The effective degrees of freedom, Ve is given by;

$$Ve = \frac{(Uc^2)^2}{\frac{(U1^2)^2}{V1}}$$

$$Ve = \frac{(0.040898^2)^2}{\frac{(8.0194e-3)^2}{2}}$$

$= 1352.9158 \approx 1353$

The confidence limits are obtain by formula,

U = Uc $\times$ K

Referring to the Table of student's t distribution. #(note: for the Ve than greater than 100, use v=100)

Let k = 2.025

Thus; U = Uc × K

= 0.040898□2.025

= 0.08281 kPa

5.7 Result using Software for uncertainty evaluation

5.7.1 The average of MSU reading and standard deviation using GUI is as below. (Figure 5.5)

Figure 5.5 shows the value of standard deviation that calculated using the system. The value are 0.138909, 3.29983×10^{-6}, 3.29983×10^{-6}, 3.9598×10^{-5}, and 0.00711 which almost same with manual calculation. It was proved the system is calculated the value correctly.

NO	MSU Reading 1st run	2nd run	3rd run	AVERAGE	STANDARD DEVIATION
1	3.889	3.798	3.996	3.89433	0.0138909
2	7.995	7.997	7.994	7.99533	3.29983e-006
3	12.00	11.998	11.997	11.9983	3.29983e-006
4	15.989	15.997	15.999	15.995	3.9598e-005
5	19.995	19.876	19.857	19.9093	0.00791158

Figure 5.5: Result for Average and Standard Deviation

5.7.2 Uncertainty Due to Repeatability of The Experiment (U1), Uncertainty Contribution Due to MSU Error (U2), Uncertainty Due to UUT Resolution / MSU resolution (U3), Combined Standard Uncertainty (Uc) and The effective degrees of freedom (Ve) (Figure 5.6)

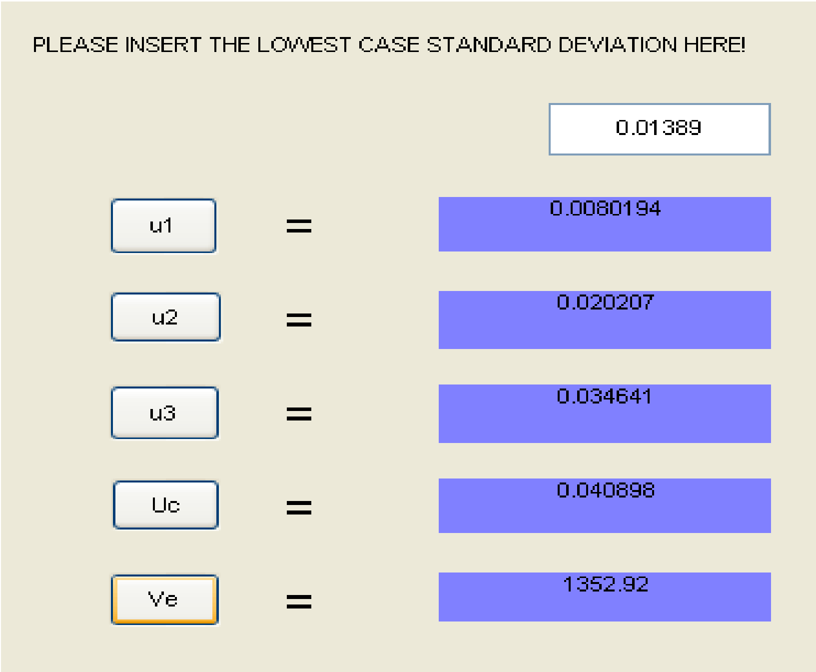

Figure 5.6 U1, U2, U3, Uc, and Ve value

5.7.3 The Total Uncertainty (U) calculated using GUI (figure 5.7)

The total uncertainty is the reading that represented the doubt that exists about the result of any measurement. Therefore 0.0828184 is the uncertainty of this system. So, lower value of uncertainty more precise the reading. Figure 5.7 shows the total uncertainty using the system.

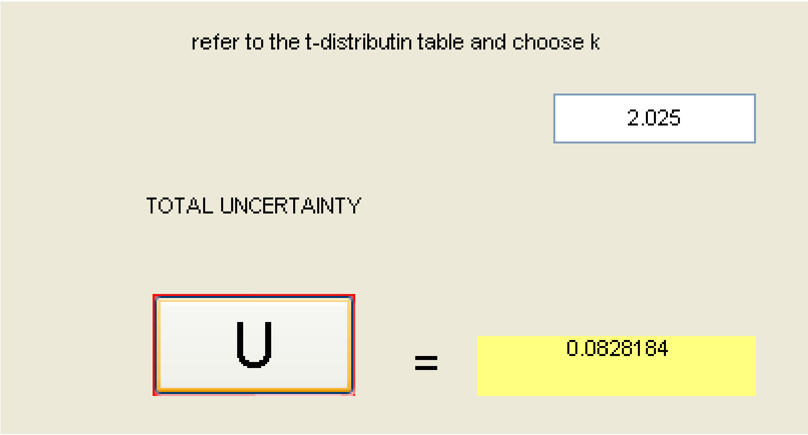

Figure 5.7 Total uncertainty

5.8 Costing and Commercialization

This project do not have any cost because it more to the system development using software. All the instruments and DAQ card is provided at Lab. As this system is use for the learning process for Industrial Instrumentation subject, so it not suit to commercialize.

CHAPTER 6

CONCLUSION AND RECOMMENDATION

6.1 Conclusion

The objectives of the project are successfully achieved. After finished this project, the basic of temperature instrumentation were able to be understand such as the theory of temperature measurement, thermocouple connection and skills of handle experimental instruments.

Then, the PCI 1710HG is used to integrate the instrument and software for user easily interfaces with the system. The basic operation of DAQ card is also has been studied included DAQ configuration using real time window target. Beside that SIMULINK model are also studied for running DAQ and transferring data to MATLAB WORKSPACE. Method for displayed data to MATLAB GUI from WORKSPACE also has been success implemented to the system.

The system was able to calculate the average and output error of actual reading, plot five point calibration and error curve. Besides, uncertainty evaluation also developed. This system can be implementing in Industrial Instrumentation class.

6.2 Recommendation

There are some future research to be considered in improving and searching for the details of this project. First is by using Excel link. Although, excel link was success implemented in the system but the user still need to key in the data manually. Therefore in the future this system can be improved by implement excel link that can automatically save all the graph and result of all calculation so that user no need to key in it manually. This excel link would make user easily do their lab report.

Besides that, to make the user easily simulate the data using DAQ, in future use one pushbutton and program it to be able to capture data for each temperature that required automatically.

REFERENCES

[1] US Department of energy office of fossil energy, national energy Technology laboratory

[2] http://en.wikipedia.org/wiki/MATLAB

[3] Dr Recayi Pecen, Dr. MD Salim, & Mr. Ayhan Zora, A LabView Based Instrumentation System for a Wind-Solar Hybrid Power Station, *Journal of Industrial* Technology, June 2004 through August 2004, Volume 20, Number

[4] Scott T. Smith, MATLAB Advanced GUI Development, http://www.mat-gui.com

[5] V. Gyurjyan, C. Timmer, D. Abbott, G. Heyes, E. Jastrzembski, D. Lawrence, E. Wolin, Jeferson LAB DATA acquisition run control system, TJNAF, Newport News, VA 23606, USA

[6] Bernard Morris, Vice-President of Sales, Hart Scientific, Inc, Temperature Calibration in Industrial Processes

[7] Stanley Wolf and Richard F.M Smith, student reference Manual for Electronic Instrumentation Laboratories, 2nd Ed, Pearson Prentice Hall, 2004.

[8] Bartelt, Industrial Control Electronics devices, System & Application, 2nd Ed, Thomson Learning, 2002.

[9]http://www.yokogawa.com/fld/TEMPERATURE/fld-temperature-01en.htm

[10] http://www.ueidaq.com/daq.html

[11] Gerson Beauchamp-baez, Luis V. Melendez-Gonzalez. Department of Electrical and Computer Engineering. University of Puerto Rico – Mayaguez Campus
Available at: **fie.engrng.pitt.edu**/fie98/papers/1226.pdf

[12] http://zone.ni.com/devzone/cda/tut/p/id/4084

[13] US Department of energy office of fossil energy, national energy Technology laboratory

[14] Fred Schraff, P.E., IOtech, Inc. The Principles and Methods of Using Thermocouples
Available at: http://www.iotech.com/mcjun96.html

[15] http://edis.ifas.ufl.edu/AE077

[16] http://en.wikipedia.org/wiki/MATLAB

[17] Tom Green and M.E Parten, PC-based Medical Data Acquisition and Analysis, Texas Tech University.

[18] http:// www.blinkdagger.com

APPENDIX A

Choice of Master Standard Units

The units under test (UUT) will be a the Temperature Recorder , Model UR1000. Master Standard Units for thermocouple will be the Digital Thermometer Md: 7563 with the 4 wire PT100 into the Isotech Jupiter. Master Standard Units for RTD calibration will be the Digital Thermometer Md: 7563 with the 4 wire PT100 into the Isotech Jupiter Model: CA100 or Decade Resistance Box, Model: 279301. The specified accuracies of these instruments are as follows.

Table 3. Accuracy Specifications

UUT SPECIFICATIONS

Instrument	Range	Span	Accuracy
TC Wire	Type K 0-1000 °C Type T 0-200 °C	Type K 0-200 °C Type T 0-200 °C ~~Internal RJC~~	± 2.2 °C ± 2.2 °C ± 0.5 °C
Universal Recorder UR1000	TC Type K -200.0 to 1370.0 °C RT PT100 -200.0 to 550.0 °C DC Volt 0 to 200m V	TC Type K 0 –400.0 °C Internal RJC RTD PT100 0-100.0 °C DC Volt 0.0- 200.0m V	Refer to Appendix A2
RTD probe	PT100 0-400 °C	PT100 0-200 °C	Class B- ±0.012% Class A: ±0.15 °C

Table 3. Cont

MSU SPECIFICATIONS

Instrument	Range	Span	Accuracy
Digital Thermometer 756301	TC Type K -200 to 1370.0 °C RTD PT100 -200 to 650.0 °C DC Volt 0 to 200mV Ohms Range 0-200 Ω	TC Type K 0 –200.0 °C RTD PT100 0-100.0 °C DC Volt 0.0– 199.9999 mV Ohms Range 0- 199.9999Ω	Refer to Appendix
Decade Box 279301	0.100 Ω– 1,111.210Ω		0.01% + 2mΩ @ 23 ± 2° C
Dry Bath Jupiter 650	30 -650°C	30-250°C	± 0.02° C
Ice Point	0°C	0°C	± 0.03° C
6.5 digit DC Source 7651	0-10mV 0-100mV	0-10mV 0-100mV	± (0.025% of setting + 5 μV) ± (0.025% of setting + 10 μV)

From the data given above the proper MSU can be chosen on the basis of the Test Uncertainty Ratio (TUR).

Recognized rules-of-thumb are greater than or equal to 4:1 (US Mil-Std 45662A and latterly ANSI/NCSL-Z540) or 3:1 (ISO10012).

APPENDIX B

Thermocouple Reference

TABLE A3-1 Type K Thermocouple Thermal E.M.F. Table

This is the reference thermal e.m.f. table defined in JIS C1602-1995.

Unit: µV

°C	0	−1	−2	−3	−4	−5	−6	−7	−8	−9	°C
−270	−6 458										−270
−260	−6 441	−6 444	−6 446	−6 448	−6 450	−6 452	−6 453	−6 455	−6 456	−6 457	−260
−250	−6 404	−6 408	−6 413	−6 417	−6 421	−6 425	−6 429	−6 432	−6 435	−6 438	−250
−240	−6 344	−6 351	−6 358	−6 364	−6 370	−6 377	−6 382	−6 388	−6 393	−6 399	−240
−230	−6 262	−6 271	−6 280	−6 289	−6 297	−6 306	−6 314	−6 322	−6 329	−6 337	−230
−220	−6 158	−6 170	−6 181	−6 192	−6 202	−6 213	−6 223	−6 233	−6 243	−6 252	−220
−210	−6 035	−6 048	−6 061	−6 074	−6 087	−6 099	−6 111	−6 123	−6 135	−6 147	−210
−200	−5 891	−5 907	−5 922	−5 936	−5 951	−5 965	−5 980	−5 994	−6 007	−6 021	−200
−190	−5 730	−5 747	−5 763	−5 780	−5 797	−5 813	−5 829	−5 845	−5 861	−5 876	−190
−180	−5 550	−5 569	−5 588	−5 606	−5 624	−5 642	−5 660	−5 678	−5 695	−5 713	−180
−170	−5 354	−5 374	−5 395	−5 415	−5 435	−5 454	−5 474	−5 493	−5 512	−5 531	−170
−160	−5 141	−5 163	−5 185	−5 207	−5 228	−5 250	−5 271	−5 292	−5 313	−5 333	−160
−150	−4 913	−4,936	−4 960	−4 983	−5 006	−5 029	−5 052	−5 074	−5 097	−5 119	−150
−140	−4 669	−4 694	−4 719	−4 744	−4 768	−4 793	−4 817	−4 841	−4 865	−4 889	−140
−130	−4 411	−4 437	−4 463	−4 490	−4 516	−4 542	−4 567	−4 593	−4 618	−4 644	−130
−120	−4 138	−4 166	−4 194	−4 221	−4 249	−4 276	−4 303	−4 330	−4 357	−4 384	−120
−110	−3 852	−3 882	−3 911	−3 939	−3 968	−3 997	−4 025	−4 054	−4 082	−4 110	−110
−100	−3 554	−3 584	−3 614	−3 645	−3 675	−3 705	−3 734	−3 764	−3 794	−3 823	−100
−90	−3 243	−3 274	−3 306	−3 337	−3 368	−3 400	−3 431	−3 462	−3 492	−3 523	−90
−80	−2 920	−2 953	−2 986	−3 018	−3 050	−3 083	−3 115	−3 147	−3 179	−3 211	−80
−70	−2 587	−2 620	−2 654	−2 688	−2 721	−2 755	−2 788	−2 821	−2 854	−2 887	−70
−60	−2 243	−2 278	−2 312	−2 347	−2 382	−2 416	−2 450	−2 485	−2 519	−2 553	−60
−50	−1 889	−1 925	−1 961	−1 996	−2 032	−2 067	−2 103	−2 138	−2 173	−2 208	−50
−40	−1 527	−1 564	−1 600	−1 637	−1 673	−1 709	−1 745	−1 782	−1 818	−1 854	−40
−30	−1 156	−1 194	−1 231	−1 268	−1 305	−1 343	−1 380	−1 417	−1 453	−1 490	−30
−20	−778	−816	−854	−892	−930	−968	−1 006	−1 043	−1 081	−1 119	−20
−10	−392	−431	−470	−508	−547	−586	−624	−663	−701	−739	−10
0	0	−39	−79	−118	−157	−197	−236	−275	−314	−353	0

°C	0	1	2	3	4	5	6	7	8	9	°C
0	0	39	79	119	158	198	238	277	317	357	0
10	397	437	477	517	557	597	637	677	718	758	10
20	798	838	879	919	960	1 000	1 041	1 081	1 122	1 163	20
30	1 203	1 244	1 285	1 326	1 366	1 407	1 448	1 489	1 530	1 571	30
40	1 612	1 653	1 694	1 735	1 776	1 817	1 858	1 899	1 941	1 982	40
50	2 023	2 064	2 106	2 147	2 188	2 230	2 271	2 312	2 354	2 395	50
60	2 436	2 478	2 519	2 561	2 602	2 644	2 685	2 727	2 768	2 810	60
70	2 851	2 893	2 934	2 976	3 017	3 059	3 100	3 142	3 184	3 225	70
80	3 267	3 308	3 350	3 391	3 433	3 474	3 516	3 557	3 599	3 640	80
90	3 682	3 723	3 765	3 806	3 848	3 889	3 931	3 972	4 013	4 055	90
										357	
100	4 096	4 138	4 179	4 220	4 262	4 303	4 344	4 385	4 427	4 468	100
110	4 509	4 550	4 591	4 633	4 674	4 715	4 756	4 797	4 838	4 879	110
120	4 920	4 961	5 002	5 043	5 084	5 124	5 165	5 206	5 247	5 288	120
130	5 328	5 369	5 410	5 450	5 491	5 532	5 572	5 613	5 653	5 694	130
140	5 735	5 775	5 815	5 856	5 896	5 937	5 977	6 017	6 058	6 098	140
										357	
150	6 138	6 179	6 219	6 259	6 299	6 339	6 380	6 420	6 460	6 500	150
160	6 540	6 580	6 620	6 660	6 701	6 741	6 781	6 821	6 861	6 901	160
170	6 941	6 981	7 021	7 060	7 100	7 140	7 180	7 220	7 260	7 300	170
180	7 340	7 380	7 420	7 460	7 500	7 540	7 579	7 619	7 659	7 699	180
190	7 739	7 779	7 819	7 859	7 899	7 939	7 979	8 019	8 059	8 099	190
										357	
200	8 138	8 178	8 218	8 258	8 298	8 338	8 378	8 418	8 458	8 499	200
210	8 539	8 579	8 619	8 659	8 699	8 739	8 779	8 819	8 860	8 900	210
220	8 940	8 980	9 020	9 061	9 101	9 141	9 181	9 222	9 262	9 302	220
230	9 343	9 383	9 423	9 464	9 504	9 545	9 585	9 626	9 666	9 707	230
240	9 747	9 788	9 828	9 869	9 909	9 950	9 991	10 031	10 072	10 113	240

Unit: μV

°C	0	1	2	3	4	5	6	7	8	9	°C
250	10 153	10 194	10 235	10 276	10 316	10 357	10 398	10 439	10 480	10 520	250
260	10 561	10 602	10 643	10 684	10 725	10 766	10 807	10 848	10 889	10 930	260
270	10 971	11 012	11 053	11 094	11 135	11 176	11 217	11 259	11 300	11 341	270
280	11 382	11 423	11 465	11 506	11 547	11 588	11 630	11 671	11 712	11 753	280
290	11 795	11 836	11 877	11 919	11 960	12 001	12 043	12 084	12 126	12 167	290
300	12 209	12 250	12 291	12 333	12 374	12 416	12 457	12 499	12 540	12 582	300
310	12 624	12 665	12 707	12 748	12 790	12 831	12 873	12 915	12 956	12 998	310
320	13 040	13 081	13 123	13 165	13 206	13 248	13 290	13 331	13 373	13 415	320
330	13 457	13 498	13 540	13 582	13 624	13 665	13 707	13 749	13 791	13 833	330
340	13 874	13 916	13 958	14 000	14 042	14 084	14 126	14 167	14 209	14 251	340
350	14 293	14 335	14 377	14 419	14 461	14 503	14 545	14 587	14 629	14 671	350
360	14 713	14 755	14 797	14 839	14 881	14 923	14 965	15 007	15 049	150 91	360
370	15 133	15 175	15 217	15 259	15 301	15 343	15 385	15 427	15 469	15 511	370
380	15 554	15 596	15 638	15 680	15 722	15 764	15 806	15 849	15 891	15 933	380
390	15 975	16 017	16 059	16 102	16 144	16 186	16 228	16 270	16 313	16 355	390
400	16 397	16 439	16 482	16 524	16 566	16 608	16 651	16 693	16 735	16 778	400
410	16 820	16 862	16 904	16 947	16 989	17 031	17 074	17 116	17 158	17 201	410
420	17 243	17 285	17 328	17 370	17 413	17 455	17 497	17 540	17 582	17 624	420
430	17 667	17 709	17 752	17 794	17 837	17 879	17 921	17 964	18 006	18 049	430
440	18 091	18 134	18 176	18 218	18 261	18 303	18 346	18 388.	18 431	18 473	440
450	18 516	18 558	18 601	18 643	18 686	18 728	18 771	18 813	18 856	18 898	450
460	18 941	18 983	19 026	19 068	19 111	19 154	19 196	19 239	19 281	19 324	460
470	19 366	19 409	19 451	19 494	19 537	19 579	19 622	19 664	19 707	19 750	470
480	19 792	19 835	19 877	19 920	19 962	20 005	20 048	20 090	20 133	20 175	480
490	20 218	20 261	20 303	20 346	20 389	20 431	20 474	20 516	20 559	20 602	490
500	20 644	20 687	20 730	20 772	20 815	20 857	20 900	20 943	20 985	21 028	500
510	21 071	21 113	21 156	21 199	21 241	21 284	21 326	21 369	21 412	21 454	510
520	21 497	21 540	21 582	21 625	21 668	21 710	21 753	21 796	21 838	21 881	520
530	21 924	21 966	22 009	22 052	22 094	22 137	22 179	22 222	22 265	22 307	530
540	22 350	22 393	22 435	22 478	22 521	22 563	22 606	22 649	22 691	22 734	540
550	22 776	22 819	22 862	22 904	22 947	22 990	23 032	23 075	23 117	23 160	550
560	23 203	23 245	23 288	23 331	23 373	23 416	23 458	23 501	23 544	23 586	560
570	23 629	23 671	23 714	23 757	23 799	23 842	23 884	23 927	23 970	24 012	570
580	24 055	24 097	24 140	24 182	24 225	24 267	24 310	24 353	24 395	24 438	580
590	24 480	24 523	24 565	24 608	24 650	24 693	24 735	24 778	24 820	24 863	590
600	24 905	24 948	24 990	25 033	25 075	25 118	25 160	25 203	25 245	25 288	600
610	25 330	25 373	25 415	25 458	25 500	25 543	25 585	25 627	25 670	25 712	610
620	25 755	25 797	25 840	25 882	25 924	25 967	26 009	26 052	26 094	26 136	620
630	26 179	26 221	26 263	26 306	26 348	26 390	26 433	26 475	26 517	26 560	630
640	26 602	26 644	26 687	26 729	26 771	26 814	26 856	26 898	26 940	26 983	640
650	27 025	27 067	27 109	27 152	27 194	27 236	27 278	27 320	27 363	27 405	650
660	27 447	27 489	27 531	27 574	27 616	27 658	27 700	27 742	27 784	27 826	660
670	27 869	27 911	27 953	27 995	28 037	28 079	28 121	28 163	28 205	28 247	670
680	28 289	28 332	28 374	28 416	28 458	28 500	28 542	28 584	28 626	28 668	680
690	28 710	28 752	28 794	28 835	28 877	28 919	28 961	29 003	29 045	29 087	690
700	29 129	29 171	29 213	29 255	29 297	29 338	29 380	29 422	29 464	29 506	700
710	29 548	29 589	29 631	29 673	29 715	29 757	29 798	29 840	29 882	29 924	710
720	29 965	30 007	30 049	30 090	30 132	30 174	30 216	30 257	30 299	30 341	720
730	30 382	30 424	30 466	30 507	30 549	30 590	30 632	30 674	30 715	30 757	730
740	30 798	30 840	30 881	30 923	30 964	31 006	31 047	31 089	31 130	31 172	740
750	31 213	31 255	31 296	31 338	31 379	31 421	31 462	31 504	31 545	31 586	750
760	31 628	31 669	31 710	31 752	31 793	31 834	31 876	31 917	31 958	32 000	760
770	32 041	32 082	32 124	32 165	32 206	32 247	32 289	32 330	32 371	32 412	770
780	32 458	32 495	32 536	32 577	32 618	32 659	32 700	32 742	32 783	32 824	780
790	32 865	32 906	32 947	32 988	33 029	33 070	33 111	33 152	33 193	33 234	790

APPENDIX C

T- Distribution Table

Value of $t_\rho(\upsilon)$ from the t-distribution for degree of freedom V that defines an Interval-$t_\rho(\upsilon)$ to $+t_\rho(\upsilon)$ that encompasses the fraction ρ of the distribution.

Degree of Freedom v	Fraction *p* in percent					
	68.27*	**90.00**	**95.00**	**95.45**	**99.00**	**99.73***
1	1.84	6.31	12.71	13.97	63.66	235.8
2	1.32	2.92	4.30	4.53	9.92	19.21
3	1.20	2.35	3.18	3.31	5.84	9.22
4	1.14	2.13	2.78	2.87	4.60	6.62
5	1.11	2.02	2.57	2.65	4.03	5.51
6	1.09	1.94	2.45	2.52	3.71	4.90
7	1.08	1.89	2.36	2.43	3.50	4.53
8	1.07	1.86	2.31	2.37	3.36	4.28
9	1.06	1.83	2.26	2.32	3.25	4.09
10	1.05	1.81	2.23	2.28	3.17	3.96
11	1.05	1.80	2.20	2.25	3.11	3.85
12	1.04	1.78	2.18	2.23	3.05	3.76
13	1.04	1.77	2.16	2.21	3.01	3.69
14	1.04	1.76	2.14	2.20	2.98	3.64
15	1.03	1.75	2.13	2.18	2.95	3.59
16	1.03	1.75	2.12	2.17	2.92	3.54
17	1.03	1.74	2.11	2.16	2.90	3.51
18	1.03	1.73	2.10	2.15	2.88	3.48
19	1.03	1.73	2.09	2.14	2.86	3.45
20	1.03	1.72	2.09	2.13	2.85	3.42
25	1.02	1.71	2.06	2.11	2.79	3.33
30	1.02	1.70	2.04	2.09	2.75	3.27
35	1.01	1.70	2.03	2.07	2.72	3.23
40	1.01	1.68	2.02	2.06	2.70	3.20
45	1.01	1.68	2.01	2.06	2.69	3.18
50	1.01	1.68	2.01	2.05	2.68	3.16
100	1.005	1.660	1.984	2.025	2.626	3.077
	1.000	1.645	1.960	2.000	2.576	3.000

*For a quantity Z described by a normal distribution with expectation μ_z and standard deviation σ, the interval $\upsilon_z \pm k\sigma$ encompasses ρ = 68.27, 95.45 AND 99.73 percent of the distribution for k = 1, 2 and 3 respectively.

APPENDIX D

PCI-1710HG DATA SHEET