Design and development of an automated temperature controller for curing ovens

Ruaan Morné Schoeman

9408029

A dissertation submitted in fulfilment of the requirements for the Magister Technologiae: Engineering: Electrical

Department:
Electronic Engineering
Faculty of Engineering and Technology
Vaal University of Technology
Vanderbijlpark

Supervisor: Dr JF Janse van Rensburg
December 2011
Declaration

I, Ruaan Morné Schoeman, declare that this project is my own, unaided work. It is being submitted for the requirements for the Magister Technologiae: Engineering: Electrical to the Department: Electronic Engineering at the Vaal University of Technology, Vanderbijlpark. It has not been submitted before for any assessment to any educational institution.

Ruaan Morné Schoeman

Date: 07/12/2011
Acknowledgements

I hereby wish to express my gratitude to the following individuals who enabled this document to be completed successfully:

- Prof DV Nicolae for guidance and encouragement throughout this study.
- Dr JF Janse van Rensburg for help in assisting with equipment and mentorship.
- Staff from the Department of Electronic Engineering and in particular Mr WJvZ Pienaar for his motivation and encouragement.
- Dr AJ Swart for guidance and encouragement with respect to completing the dissertation.
Dedication

This dissertation is dedicated to my wife Sharon, daughter Rushane and my parents, Gert and Evy.

Thank you for your love and support.
Abstract

Curing of materials in order to obtain different properties has been a practice for many years. New developments in composite materials increase the need to control certain variables during the curing process. One very significant variable is temperature. Temperature control by itself is an old practice, however when the need for repeatedly controlling the process accurately over long periods of time arises, a system is required that outperforms normal manual control.

One of the aspects within such a system that needs to be considered is the ability to replicate the temperatures within an oven which were originally used for a specific material's curing profile. This means that a curing profile would need to be defined, saved for later and finally be interpreted correctly by the controlling system.

Different control methods were simulated to enable the system to control the temperature which has been defined by literature. This dissertation introduces a variation on the standard control methods and shows improved results.

Switching the oven on and off in order to increase or decrease internal oven temperature seems simple, but can cause switching devices to decrease their operational life span, if not designed carefully. A combination switch was introduced which harnesses the advantages of two very common switching devices to form an improved combination switch.

Software for the personal computer environment, as well as software for the embedded environment were developed and formed a control system that produced acceptable results for temperature control. Accuracies of 98% and more were achieved and found to be acceptable according to standard engineering control practices.

An accurate temperature profile controller was designed, simulated and built in order to control the temperature inside a specific curing oven which, in turn, determined the curing properties of specific materials. The overall results were satisfactory which lead to achieving the objectives outlined in this dissertation.
# Table of contents

Declaration .......................................................... ii  
Acknowledgements ................................................... iii 
Dedication ............................................................. iv  
Abstract ............................................................... v  
Table of contents ..................................................... vi 
List of figures ......................................................... viii 
List of tables ........................................................ x 
Glossary of abbreviations and definitions ......................... xi 

Chapter 1  Introduction and overview ................................ 1  
1.1  Background ....................................................... 1  
1.2  Problem statement ............................................... 2  
1.3  Research methodology ......................................... 2  
1.4  Delimitations ..................................................... 3  
1.5  Importance of research ........................................ 3  
1.6  Overview of the dissertation .................................. 3  
1.7  Research outputs ............................................... 4  
1.8  Summary ......................................................... 4  

Chapter 2  Theoretical background ................................... 5  
2.1  Introduction ...................................................... 5  
2.2  Oven electrical equivalent model ............................... 5  
2.3  Thermal measuring devices ..................................... 10 
2.3.1  Thermocouples ................................................ 10  
2.3.2  Thermography ................................................ 13  
2.4  Electrical switching devices .................................... 14  
2.4.1  The relay ....................................................... 14  
2.4.2  Silicon controlled rectifiers ................................. 15  
2.4.3  Triode for alternating currents .............................. 16  
2.4.4  Snubber circuits .............................................. 17  
2.5  Control theory ................................................... 19  
2.5.1  Types of controllers ......................................... 21  
2.6  Summary ......................................................... 25
List of figures

Figure 1  Representation of an electrical oven as a plant (University of EXETER, 2010)  
Figure 2  Electrical equivalent of a thermal oven (University of EXETER, 2010) 
Figure 3  Simplified electrical equivalent of a thermal oven  
Figure 4  Thermocouple setup (Duff and Towey, 2010)  
Figure 5  Thermogram of electrical fuses (POWER PLUS ENGINEERING INC, 2003) 
Figure 6  Representation of a relay and its symbol  
Figure 7  Relay contact bounce during activation and deactivation process (Electronix Express, 2010)  
Figure 8  A SCR's characteristic voltage/current curve (American Microsemiconductor Inc., 2010)  
Figure 9  SCR PN layer equivalent circuit  
Figure 10  SCR diode equivalent circuit 
Figure 11  A TRIAC characteristic voltage/current curve (American Microsemiconductor Inc., 2010)  
Figure 12  A snubber circuit  
Figure 13  A DIAC used to drive a TRIAC circuit (Coates, 2008)  
Figure 14  DIAC characteristic curve (Uddin, 2011)  
Figure 15  A feedback control system  
Figure 16  An open-loop control system  
Figure 17  Sinusoidal input and transfer function  
Figure 18  Industrial PID controller (Nippon Instruments (India) pvt, 2011)  
Figure 19  Step response of a system  
Figure 20  Temperature measuring setup  
Figure 21  Data obtained for the oven in the ON position  
Figure 22  Normalized data obtained for the oven in the ON position  
Figure 23  Different objects’ temperature curves  
Figure 24  Object temperatures and corresponding oven inner air temperature curves  
Figure 25  MATLAB curve for data collected, mathematical model fitting and residuals  
Figure 26  CURVE EXPERT data, mathematical model fitting and residuals  
Figure 27  CURVE EXPERT mathematical model coefficients and standard error  
Figure 28  MATLAB representation of the simplified electrical oven model  
Figure 29  MATLAB model of the simplified oven model and controller  
Figure 30  Step response analysis graph  
Figure 31  MATLAB SISO tools stability analysis graph  
Figure 32  Control methods’ step response graphs (Schoeman, 2011)  

viii
Figure 33  Proteus circuit used for simulation of control methods  35
Figure 34  Proteus simulation of PI-BB control methods  35
Figure 35  A flowchart outlining the functions of the CCS C software program  36
Figure 36  Combination switch  37
Figure 37  Simulation model of combination switch  37
Figure 38  MATLAB simulated results of switching currents in relay, TRIAC and load  38
Figure 39  Measured current transfers between relay and TRIAC  39
Figure 40  Relay settling duration  39
Figure 41  Practical setup of a 3 kW oven and switching board  41
Figure 42  Curing profile creation capabilities of developed software  42
Figure 43  Curing profile saving capabilities of developed software  42
Figure 44  Curing profile saved in developed software  43
Figure 45  MAX6675 integrated circuit mounted on the prototype board  44
Figure 46  Combination power switch on the prototype board  44
Figure 47  Operator information display unit  45
Figure 48  Prototype control unit with combination power switch  46
Figure 49  Infrared thermometer with laser target indicator, pointed at the thermocouple tip  46
Figure 50  Predefined curing profile control results  47
Figure 51  The second curing profile control results  47
Figure 52  Predefined curing profile validity results  48
Figure 53  Over-shoot and under-shoot visualisation using MATLAB  48
List of tables

Table 1  Equivalence between thermal and electrical entities  6
Table 2  Voltage and current and current and voltage relationships, summarized  
          (Nise, 2000:52)  7
Table 3  Laplace transformations (Boyd, 2009:2)  9
Table 4  Thermocouple comparison (Efunda, 2010)  12
Table 5  Controller parameters for a step response method using Ziegler-Nichols 
          (Dogan, 2002)  25
# Glossary of abbreviations and definitions

<table>
<thead>
<tr>
<th>A</th>
<th>AC</th>
<th>Alternating current</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Armature</td>
<td>The moving iron part of a solenoid or relay</td>
</tr>
<tr>
<td>C</td>
<td>Curing</td>
<td>The process of setting or hardening of a material</td>
</tr>
<tr>
<td>D</td>
<td>DIAC</td>
<td>A two-terminal AC device that, once gated on by sufficient forward voltage, permits the flow of current until reverse biased</td>
</tr>
<tr>
<td></td>
<td>DITI</td>
<td>Digital Infrared Thermal Imaging</td>
</tr>
<tr>
<td>G</td>
<td>GUI</td>
<td>Graphical user interface used in the software environment</td>
</tr>
<tr>
<td>I</td>
<td>I²C</td>
<td>Inter-Integrated Circuit is a multi-master serial single-ended computer bus used to attach low-speed peripherals to a motherboard, embedded systems, cell phones, or other electronic devices</td>
</tr>
<tr>
<td>M</td>
<td>Microcomputer</td>
<td>A microcontroller including external storage and memory devices</td>
</tr>
<tr>
<td></td>
<td>Pneumatics</td>
<td>The study of the mechanical properties of air and other gases</td>
</tr>
<tr>
<td>R</td>
<td>RTD</td>
<td>Resistance temperature detectors</td>
</tr>
</tbody>
</table>

| S  | SCR | A current controlled four-layer device for high power and low speed applications electronic devices |
|    | Snubber | An auxiliary circuit used to control the rate of rise or fall of the current flowing into a power electronic device |
|    | SPI | Serial Peripheral Interface, a full-duplex synchronous serial interface for connecting low-/medium-bandwidth external devices using four wires |
|    | Thermalset | Having the property of becoming permanently hard and rigid when heated or cured |
|    | Thermistors | An electrical resistor whose resistance is greatly reduced by heating, used for measurement and control |
|    | Thermography | A technique wherein an infrared camera photographically portrays an object’s surface temperature |
|    | TRIAC | A power switch that is functionally a pair of converter-grade thyristors connected in anti-parallel |
|    | TSVUT | Technology Station of the Vaal University of Technology |
Chapter 1  
Introduction and overview

1.1 Background

Thermalset is the phase change of a compound into a substantially infusible and insoluble material when it is cured by the application of heat or by chemical means (Swartz, 1984:28). Curing of materials is done at various temperatures in specially designed ovens. A material, such as silica, cures at very high temperatures between 800°C and 1200°C, depending on the materials it is mixed with. The temperature inside an oven affects the properties of the cured material. Thus the curing temperature, which is the temperature at which a cast, moulded or extruded product is subject to during curing (Swartz, 1984:7), is very important and needs to be controlled.

There are a variety of commercial devices available that could be used to monitor the temperature of a curing oven. However, for high temperature measurements, only a few devices can be utilized, depending on the accuracy and temperature ranges required (Dogan, 2002:59). Accurate measurement of temperature is not easy and to obtain accuracies better than 0.5°C requires great care.

Radio thermometry devices measure the radiation emitted by hot objects and is based upon the emissivity of the object, which is usually not known and additionally may vary with time (Dogan, 2002:69). However, placing sensory devices inside these ovens exposes them to extreme heat environments, resulting in failure of sensors with cumulative exposure, sensory devices can also not be placed inside the curing materials as this would deform the required moulding shape.

The air temperature in an oven could be measured. However, air temperature fluctuations, variations in radiation, nonlinearities in sensor characteristics and sensor drifts need to be kept in mind (CAPGO Pty Ltd, 2010). The following sensory devices, among others, are available for measuring air temperature: thermocouples, resistance temperature detectors (RTD) and monolithic temperature sensors. These sensory devices are designed to function in harsh environmental conditions.

Switching devices are components which are used to switch loads ON and OFF. Switching either direct current (DC) or alternating current (AC) presents its own unique problems. One such problem is heat dissipation in the device itself. Thus, obtaining a suitable device that
can switch high currents required by ovens that are controlled by small signal devices (such as microcontrollers) will prove to be challenging in itself.

Sensory devices may be complemented by a microcomputer which is widely used to monitor environmental data and execute decisions based there-on. The term microcomputer is used to describe a system that includes a minimum of a microprocessor, program memory, data memory and input/output functions. Thus, a microcomputer system can be anything from a large computer having hard disks, floppy disks and printers, to a single chip computer (Dogan, 2002:5). For example, microcomputer systems could be implemented in the controlling of various processes, such as maintaining room temperature by activating or deactivating an air conditioning unit.

1.2 Problem statement

Manually controlling the operating temperatures of a curing oven at the Technology Station of the Vaal University of Technology (TSVUT) lacks accuracy and consistency. The curing process is often repeated with materials of the same type, yet the curing results with respect to the material’s properties vary. This mainly occurs because the operator has to manually switch the oven’s heating element ON or OFF according to personal feeling. Thus, no accurate control technique or constant evaluation parameters are applied to achieve consistency in the repetition of the thermalset or curing process.

1.3 Research methodology

- Temperature range, curing profiles, type of oven and materials to be cured will be obtained through interviews with staff at the TSVUT.
- The best switching technique for high current devices will be identified through a literature search.
- Literature on microcomputers and their communication techniques will be gathered and evaluated to determine what would be needed to control the heating modules.
- Control techniques will be investigated and applied to the embedded environment.
- Data will be collected with respect to the oven’s heating characteristics.
- An algorithm will be developed based on the data obtained.
- Simulation of the algorithm in MATLAB will be done.
- A prototype controller will be designed and simulated.
- Results will be evaluated and corrections to the design will be made.
• If simulations of the control technique and communication are satisfactory, then a prototype printed circuit board will be developed.
• The prototype will be then tested with a curing oven.
• Improvements will be made in order to ensure consistency of repeated curing processes.

1.4 Delimitations

• The research will not include the investigation of cooling systems which could assist in the rapid cooling of materials as this will require a large budget.
• The research will not focus on industrial ovens as this research is for a specific need at the TSVUT.
• No alterations or additions can be made to the oven as specified by the TSVUT.

1.5 Importance of research

The research will benefit the TSVUT, as they will be able to produce concurrent samples of a specific curing process. Industry will also benefit from this research, as they will have an entity that could produce consistent results for use in testing phases.

1.6 Overview of the dissertation

This dissertation reports on the development of a curing oven controller which will automatically determine the characteristics of an oven and deduce control parameters to be used in the controlling action.

Chapter 2 presents the equivalent representation of an oven as an electrical circuit. It further describes different temperature sensory equipment currently available in industry. Different switching devices are presented along with various temperature control methods which are used in industrial applications.

Chapter 3 introduces the parameters of the oven which are obtained from the simulation model in MATLAB. Other critical components, such as the low loss AC switch, are also presented.

Chapter 4 presents the developed control software along with measurements and tests conducted in order to prove that a curing profile can be replicated.
Chapter 5 contains the conclusions and recommendations obtained from the study.

1.7 Research outputs

This study has already produced the following peer-reviewed research outputs published by IEEE Xplore, the digital database of the Institute of Electrical and Electronic Engineers.

- A conference paper for Optim 2010, entitled “Self-tuning curing oven control” and
- A conference paper for Africon 2011, entitled “Embedded PI-bang-bang curing oven controller”.

1.8 Summary

The accurate curing of materials is an important requirement of the work done by the TSVUT. It is important that the curing process of composite materials be reproduced repeatedly to ensure that the characteristics of the materials are consistent over time. Simulation, collection of data, design and development form an integral part of this research. Cooling systems will not be investigated to keep costs to a minimum in order to produce a control unit that is affordable to the TSVUT as well as to industry.
2.1 Introduction

The temperature profile during the curing process of composite materials determines the final characteristics of the cured sample (Bogetti & Gillespie, 1992:626). The problem however, of reproducing the same set of parameters with respect to different temperature settings for specific time periods necessitates the use of a precise controlling device.

The curing of composite materials is normally done in an industrial oven. A heating element of low resistance generates heat which is then transferred to the material or object. This can be seen in figure 1, where heat from the heating element is transferred to the object inside the oven. The heating element is normally a high power device which allows for quick heat generation. The switching of the heating element proves challenging due to the high values of current which flow through it.

2.2 Oven electrical equivalent model

Heat is the process of energy transfer from one body or system to another due to a difference in temperature (Kesidou & Duit, 1993:85). Thermal energy can be defined as the energy of a body which increases with its temperature. Energy transfer by heat can occur between objects through radiation, conduction and/or convection. Temperature can be used
as a measure of the internal energy. Analysis of the heat flow in an oven can either be done by means of thermodynamics or by using an electrical analogy of the heat flow path.

Heat flow can be modelled by an analogy to electrical parameters shown in table 1, where heat flow is represented by current, temperatures are represented by voltages (Schroder & De Doncker, 2000:114-117), heat sources are represented by constant current sources, thermal resistances are represented by resistors and thermal capacitances by capacitors (Birca-Galateanu, 2005).

Table 1 Equivalence between thermal and electrical entities

<table>
<thead>
<tr>
<th>Thermal quantity</th>
<th>Unit</th>
<th>Electrical quantity</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P$ – Heat flow, power</td>
<td>W</td>
<td>$I$ – Current flow</td>
<td>A</td>
</tr>
<tr>
<td>$\Delta \theta$ – Temperature difference</td>
<td>K</td>
<td>$V$ – Voltage difference</td>
<td>V</td>
</tr>
<tr>
<td>$R_{th}$ – Thermal resistance</td>
<td>K/W</td>
<td>$R$ – Electrical resistance</td>
<td>Ω</td>
</tr>
<tr>
<td>$C_{th}$ – Thermal mass, capacitance</td>
<td>J/K</td>
<td>$C$ – Electrical capacitance</td>
<td>F</td>
</tr>
<tr>
<td>$T_{th} = R_{th} \times C_{th}$ – Thermal RC constant</td>
<td>s</td>
<td>$T = R \times C$ – Electrical RC constant</td>
<td>s</td>
</tr>
</tbody>
</table>

This can be seen in figure 2, where

- $R_1$ represents the thermal resistance between the heating element and the oven,
- $R_2$ represents the resistance between the oven and the environment,
- $C_1$ represents the equivalent capacitance of the oven,
- $C_2$ represents the equivalent capacitance of the heating element,
- $\theta_a$ represents the ambient temperature of the environment,
- $W$ represents the power dissipated in the heating element,
- $\theta_e$ represents the oven temperature and
- $\theta_h$ represents the heating element temperature.

![Figure 2 Electrical equivalent of a thermal oven (University of EXETER, 2010)]
Capacitor \( C_2 \) can be neglected from the equivalent diagram as the capacity of the element to store heat is small in comparison to that of the oven capacity to store heat. Subsequently figure 3 shows a simplified diagram that can be used as the electrical equivalent of a thermal oven with input voltage \( v_i(t) \), input current \( i(t) \) and output voltage \( v_o(t) \).

![Figure 3 Simplified electrical equivalent of a thermal oven](image)

Finding the transfer function of this circuit is now simplified, which may be done by applying basic electrical principles (see table 2) of current and voltage rules as well as the relationships between voltage, current and impedance (Nise, 2000:52).

<table>
<thead>
<tr>
<th>Component</th>
<th>Voltage to Current</th>
<th>Current to Voltage</th>
<th>Impedance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacitor</td>
<td>( v_i(t) = \frac{1}{C} \int i(t) , dt )</td>
<td>( i(t) = C \frac{dv_i(t)}{dt} )</td>
<td>( Z(s) = \frac{1}{Cs} )</td>
</tr>
<tr>
<td>Resistor</td>
<td>( v_i(t) = R_i(t) )</td>
<td>( i(t) = \frac{v_i(t)}{R} )</td>
<td>( Z(s) = R )</td>
</tr>
<tr>
<td>Inductor</td>
<td>( v_i(t) = L \frac{di(t)}{dt} )</td>
<td>( i(t) = \frac{1}{L} \int v_i(t) , dt )</td>
<td>( Z(s) = Ls )</td>
</tr>
</tbody>
</table>

The time domain equation can be found by applying Kirchhoff’s voltage law:

\[
v_i(t) = R_i i(t) + v_o(t) \quad \text{... (1)}
\]

Where \( v_i(t) \) = input voltage in V

\( i(t) \) = input current in A

\( R_i \) = thermal resistance between heating element and oven in \( \Omega \)

\( v_o(t) \) = output voltage in V
Using table 2 the current flowing through the capacitor can be written as:

\[ i(t) = C \frac{d}{dt} v_o(t) \] \hspace{1cm} ... (2)

Where \( i(t) \) = charge or discharge time in s  
\( v_o(t) \) = resistance in series in \( \Omega \)  
\( C \) = capacitance of the circuit in F

By substituting (2) into (1) results in:

\[ v_i(t) = R_iC_i \frac{d}{dt} v_o(t) + v_o(t) \] \hspace{1cm} ... (3)

In Laplace format

\[ v_i(s) = [R_iC_i s + 1]v_o(s) \] \hspace{1cm} ... (4)

The overall transfer function is therefore defined as:

\[ \frac{v_o(s)}{v_i(s)} = \frac{1}{R_iC_i s + 1} \] \hspace{1cm} ... (5)

The time constant is often related directly to the circuit \( RC \) value (the product of the resistance in Ohms and the capacitance in Farads) or to its \( L/R \) value (the ratio of the inductance in Henrys to its resistance in Ohms)\textit{(IEEE EED, 2000:710)}, which is expressed mathematically as:

\[ T = RC \] \hspace{1cm} ... (6)

Where \( T \) = charge or discharge time in s  
\( R \) = resistance in series in \( \Omega \)  
\( C \) = capacitance of the circuit in F

Thus, by substituting equation (6) into equation (5) yields following transfer function:
\[
\frac{V_o(s)}{V_i(s)} = \frac{1}{Ts + 1} \quad \ldots \ (7)
\]
\[
\frac{V_o(s)}{V_i(s)} = \frac{T^{-1}}{s + T^{-1}} \quad \ldots \ (8)
\]

In order to obtain a transfer function with reference to time requires the application of the inverse Laplace transformations (Nise, 2000:40), as can be seen in table 3.

### Table 3 Laplace transformations (Boyd, 2009)

<table>
<thead>
<tr>
<th>Function</th>
<th>Laplace</th>
<th>Function</th>
<th>Laplace</th>
<th>Function</th>
<th>Laplace</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( \frac{1}{s} )</td>
<td>( \cos at )</td>
<td>( \frac{s}{(s^2 + a^2)} )</td>
<td>( \sin at )</td>
<td>( \frac{a}{(s^2 + a^2)} )</td>
</tr>
<tr>
<td>( e^{at} )</td>
<td>( \frac{1}{s - a} )</td>
<td>( e^{at} \cos bt )</td>
<td>( \frac{(s - a)}{[(s - a)^2 + b^2]} )</td>
<td>( e^{at} \sin bt )</td>
<td>( \frac{b}{[(s - a)^2 + b^2]} )</td>
</tr>
<tr>
<td>( t^n )</td>
<td>( \frac{n!}{s(n+1)} )</td>
<td>( uc(t) )</td>
<td>( \frac{e^{-cs}}{s} )</td>
<td>( uct(t) y(t - c) )</td>
<td>( e^{-cs} F(s) )</td>
</tr>
<tr>
<td>( t^p, p &gt; -1 )</td>
<td>( \frac{G(p+1)}{s(p+1)} )</td>
<td>( t^n e^{at} )</td>
<td>( \frac{n!}{(s - a)(n+1)} )</td>
<td>( e^{ct} f(t) )</td>
<td>( F(c - s) )</td>
</tr>
</tbody>
</table>

Applying the inverse Laplace transform for a unit step function with a magnitude of one, results in the following:

\[
V_i(s) = \frac{1}{s} \quad \ldots \ (9)
\]
\[
V_o(s) = V_i(s) \frac{T^{-1}}{s + T^{-1}}
\]
\[
V_o(s) = \frac{T^{-1}}{s(s + T^{-1})} \quad \ldots \ (10)
\]
\[
V_o(t) = 1 - e^{-\frac{t}{T}} \quad \ldots \ (11)
\]

Substituting equation (6) into equation (11) yields an equation which represents the output according to electrical terminology:

\[
V_o(t) = 1 - e^{-\frac{t}{RC}} \quad \ldots \ (12)
\]
2.3 Thermal measuring devices

Heat measurements can be made by making use of various different sensory devices. These could include thermistors, RTD's, thermocouples and thermography sensors. However, when temperatures exceed 800°C, few sensory devices have the ability to measure and interpret temperatures correctly.

2.3.1 Thermocouples

Only two devices can be used in the measurement of very high temperatures, namely thermocouples and thermography sensors (OMEGA Engineering, 2005).

A thermocouple (TC) consists of two wires of different conductive material, connected to each other by means of two junctions forming an electrical circuit. If one junction is at temperature $\theta_{\text{ref}}$ and the other at $\theta_{\text{tip}}$, then an electromotive force (EMF) is generated in the circuit, which is dependant on the materials and temperatures $\theta_{\text{ref}}$ and $\theta_{\text{tip}}$, known as the Seebeck effect (Tong, 2001).

Thomas Johann Seebeck discovered the existence of thermoelectric currents while observing electromagnetic effects associated with bismuth-copper and bismuth-antimony circuits (ASTM Committee E-20 on Temperature Measurement, 1974). The Seebeck effect entails the overall conversion of thermal energy into electrical energy where the Seebeck voltage can be represented as:

$$ E = a (\theta_{\text{tip}} - \theta_{\text{ref}}) $$

... (13)

Where

- $E$ = electromagnetic force in mV
- $a$ = proportionality constant known as the Seebeck coefficient
- $\theta_{\text{ref}}$ = hot junction temperature in °C
- $\theta_{\text{tip}}$ = cold junction temperature in °C

Assume that the Seebeck coefficients of two different metallic materials and their lead wires are $S_A$, $S_B$, and $S_{\text{Lead}}$ respectively. All three Seebeck coefficients are functions of temperature. The output voltage ($v_{\text{out}}$) measured at the gauge seen in figure 4 can be represented as:
\[ v_{out} = \int_{\text{Ref}}^{\theta_{\text{tip}}} S_{\text{lead}} (\theta) \frac{d\theta}{dx} dx + \int_{\text{Ref}}^{\theta_{\text{tip}}} S_A (\theta) \frac{d\theta}{dx} dx + \int_{\text{Ref}}^{\theta_{\text{tip}}} S_{\text{lead}} (\theta) \frac{d\theta}{dx} dx + \int_{\text{Ref}}^{\theta_{\text{tip}}} S_B (\theta) \frac{d\theta}{dx} dx \]

\[ = \int_{\theta_{\text{ref}}}^{\theta_{\text{tip}}} S_A (\theta) d\theta + \int_{\theta_{\text{ref}}}^{\theta_{\text{tip}}} S_B (\theta) d\theta \]

\[ = \int_{\theta_{\text{ref}}}^{\theta_{\text{tip}}} [S_A (\theta) - S_B (\theta)] d\theta \]

\[ \cdots (14) \]

Where \( \theta_{\text{Ref}} \) = temperature at the reference point in °C
\( \theta_{\text{Tip}} \) = temperature at the probe tip in °C
\( S_{\text{Lead}} \) = Seebeck coefficients for connecting lead
\( S_A \) = Seebeck coefficient, material A
\( S_B \) = Seebeck coefficient, material B
\( v_{out} \) = electromagnetic force produced by system in mV

According to equation (14) the voltage induced by the temperature and/or material mismatch of the lead wires will cancel, whereas in reality the lead wires will introduce noise into the system. If the Seebeck coefficient functions of the two thermocouple wire materials are pre-calibrated and the reference temperature \( \theta_{\text{Ref}} \) is known, then the temperature at the probe tip becomes the only unknown and can be directly related to the output voltage.

Figure 4 Thermocouple setup (Duff and Towey, 2010)
If the Seebeck coefficients are constant across the targeted temperature range, then the integral in equation (14) can be simplified as seen in equation (15), allowing for the temperature at the probe tip to be calculated as:

\[ v_{\text{out}} = (\theta_A - \theta_B)(\theta_{\text{tp}} - \theta_{\text{ref}}) \]
\[ \therefore \theta_{\text{tp}} = \theta_{\text{ref}} + \frac{v_{\text{out}}}{\theta_A - \theta_B} \]

... (15)

In practice, manufacturers will provide calibration functions for their products. These functions are usually high order polynomials and are calibrated with respect to a certain reference temperature, normally 0°C or 32°F. Assume that the coefficients of the calibration polynomials are \( \theta_0, \theta_1, \theta_2, \) up to \( \theta_n \), then the temperature at the probe tip can then be related to the output voltage as shown in equation (16) (Potter, 1997).

\[ \theta_{\text{tp}} = \theta_0 + \theta_1 v_{\text{out}} + \theta_2 v_{\text{out}}^2 + \cdots + \theta_n v_{\text{out}}^n \]

... (16)

A thermocouple is not an absolute temperature sensor (Bentley, 1984). In other words, a thermocouple requires a reference of known temperature which may be provided by ice water as illustrated by figure 4. While ice water is easy to obtain and has a well known reference, it’s not a practical solution out side the laboratory. Thus, common commercialized thermocouples often include another temperature sensor, such as a thermistor, to provide a reference of the ambient temperature.

### Table 4 Thermocouple comparison (Efunda, 2010)

<table>
<thead>
<tr>
<th>Type</th>
<th>Temperature range °C (continuous)</th>
<th>Temperature range °C (short term)</th>
<th>Tolerance class one (°C)</th>
<th>Tolerance class two (°C)</th>
<th>BS Colour code</th>
<th>ANSI Colour code</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>0°C to +1100°C</td>
<td>−180°C to +1300°C</td>
<td>±1.5°C between −40°C and 375°C</td>
<td>±2.5°C between −40°C and 333°C</td>
<td>±0.004°C ×T between 375°C and 1000°C</td>
<td>±0.0075°C ×T between 333°C and 1200°C</td>
</tr>
<tr>
<td>J</td>
<td>0°C to +700°C</td>
<td>−180°C to +800°C</td>
<td>±1.5°C between −40°C and 375°C</td>
<td>±2.5°C between −40°C and 333°C</td>
<td>±0.004°C ×T between 375°C and 750°C</td>
<td>±0.0075°C ×T between 333°C and 750°C</td>
</tr>
<tr>
<td>R</td>
<td>0°C to +1600°C</td>
<td>−50°C to +1700°C</td>
<td>±0.5°C between 0°C and 1100°C</td>
<td>±1.5°C between 0°C and 600°C</td>
<td>[1°C × 0.003°C × (T − 1100°C)] between 1100°C and 1600°C</td>
<td>±0.0025°C ×T between 600°C and 1600°C</td>
</tr>
</tbody>
</table>
According to EN/ANSI standards and most common industrial applications, the thermocouple is often used for temperature measurements between -40°C and 1800°C (European Solar Thermal Industry Federation, 2007:11). Thermocouples can be classified with respect to their material and their operational temperature (see table 4).

2.3.2 Thermography

Every object radiates heat and emits a certain wavelength which corresponds to the energy it is radiating. Thus thermography can be defined as images produced by infrared cameras showing temperature differences in objects by making use of the infrared spectrum (Dictionary.com, 2010). Thermography had its early origin in 480 BC when Hippocrates experimented with mud which he applied to a human body in order to determine which parts of the body would dry the mud first (Proactive Wellness & Imaging Center, 2011). During the early 1950’s, infrared spectrometry was used to monitor the movement of soldiers during night time exercises. Since then, the technology has been made available for commercial use and research within this field has increased rapidly.

Thermography has then been applied to the electrical environment (see figure 5). Fuses emit different heat levels that are represented by different colours. Calibrating these colours (spectrum occupancies) with radiating heat levels enables one to determine the temperature values for different parts of the fuse. The approximate temperature range of thermography is from -20°C to over 2,000°C. This makes it suitable for measuring high temperature values during a curing process.

![Thermogram of electrical fuses](POWER PLUS ENGINEERING INC, 2003)
2.4 Electrical switching devices

One of the simplest devices found in electrical circuits would be the switch. This device is used to close (complete) or break an electrical circuit, which in turn will allow current either to flow through or not to flow. Switching devices are abundant and come in a variety of different shapes and sizes, yet can be grouped into either mechanical or solid-state devices.

2.4.1 The relay

A device which allows an electrical circuit to control the ON and OFF state without human intervention is a relay (grouped as a mechanical switching device). As can be seen in figure 6, an inductor is used to generate a magnetic field which in turn attracts a contact, thereby closing the circuit and allowing current to flow. This inductor can be energised by other circuits which becomes the controller circuit. Contact bounce is always present during the switching period (Johler, 2000:83-93).

![Figure 6 Representation of a relay and its symbol](image)

Contact bounce relates to when contacts strike each other, their momentum and elasticity acts together to cause bounce. This often forms arcing between the contacts, which will in turn lead to contact burn, the process where carbon is deposited onto the contact surfaces of the relay. Repeated carbon deposits increase the resistance between the contacts, thereby limiting the flow of current flow and reducing the performance of the switching device. Several techniques have been developed to reduce carbon deposits during contact bounce.

Some manufacturers allow the contacts to be submerged in oil, thereby reducing the oxygen needed for burning (Buschart & Kuczka, 1992:293-299). Contact bounce also leads to high transients, which could be detrimental to surrounding circuits. Figure 7 illustrates contact bounce which exits during the switch activation and deactivation periods respectively.
2.4.2 Silicon controlled rectifiers

A silicon controlled rectifier (SCR) can be defined as a four-layer device controlled by current for high power and low speed applications (IEEE EED, 2000:646). SCRs can only be ON or OFF, with no intermediate operating states like transistors. Once latched on, the gate current can be removed and the device will remain on until the anode current becomes negative, or the current through the SCR falls below its holding current. A disadvantage is that a commutation circuit is often needed for forced turn-off (IEEE EED, 2000:646). Figure 8 illustrates the layer composition and ON/OFF states of this device.

A thyristor or SCR is seen as a switch (Taib et al., 1992:568-580). The advantage of this device is that it can operate without involving arcing, thereby negating mechanical wear. The drawback of this device is the internal resistance $R_f$ present between the positive and
negative (PN) junctions which will generate heat when current is allowed to flow. Figure 9 illustrates the equivalent circuit for the PN junctions, while figure 10 illustrates the forward resistance, which will generate heat due to current flow.

![Physical Diagram](image1)

Figure 9 SCR PN layer equivalent circuit

![Equivalent Schematic](image2)

Figure 10 SCR diode equivalent circuit

### 2.4.3 Triode for alternating currents

A triode for alternating currents (TRIAC) is a component which is equivalent to two SCRs joined in anti-parallel (paralleled but with the polarity reversed) configuration, having their gates connected together. The formal name for a TRIAC is a bidirectional triode thyristor (Gentry et al., 1965). This results in a bidirectional electronic switch that can conduct current in either direction when it is triggered, and thus does not have any polarity. Figure 11 shows the voltage/current curve of the TRIAC and the areas where the device conducts.
The device can be triggered by either a positive or a negative voltage being applied to its gate electrode. Once triggered, the device continues to conduct until the current through the device drops below a certain threshold value (the holding current), such as at the end of a half-cycle of an alternating current. This makes the TRIAC a very convenient switch for AC circuits, allowing the control of very large currents with milliampere-scale control currents.

TRIACs are able to achieve fast switching speeds with no contact bounce as they are a solid-state devices. However, one drawback exists which is the internal resistance, which is present between the PN junctions in the activated state (reverse and forward). Following Ohms law, the internal resistances $R_t$ and $R_r$ will produce heat, which will need to be dissipated when current is flowing. Additional heat dissipation devices, such as heat sinks or cooling fans, are therefore required to keep the device operating correctly.

### 2.4.4 Snubber circuits

A snubbing circuit can be defined as an auxiliary circuit used to control the current rise and fall rate, flowing into a device or the voltage fall or rise rate across the device during turn-off (IEEE EED, 2000:658). Using a snubber circuit assists in turning-off a device as well as to prevent premature triggering, which may be caused by voltage spikes originating from the mains supply.

The function of the snubber circuit is to protect semiconductor devices by

- limiting device voltages during turn-off transients,
- limiting device currents during turn-on transients,
• limiting the rate-of-rise (\(di/dt\)) of currents through the semiconductor device at device turn-on,
• limiting the rate-of-rise (\(dv/dt\)) of voltages across the semiconductor device at device turn-off and 
• shaping the switching trajectory.

Figure 12 is an example of such a circuit which could be used to protect a SCR against switching transients.

A gate resistor or capacitor may be connected between the gate and anode to further prevent false triggering. That, however, increases the required trigger current and / or adds latency (capacitor charging). A DIAC is often used to drive the gates of both TRIACs and SCRAs as can be seen in figure 13.
The DIAC is designed to have a particular break over voltage, which can be in the region of approximately 29 V, when a smaller voltage is applied of either polarity, the device remains in a high resistance state with only a small leakage current flowing. When the break over voltage is achieved of either polarity, the device exhibits a negative resistance which results in its characteristic curve as shown in figure 14.

When the voltage across the DIAC exceeds the break-over voltage, an increase in current is observed accompanied by a drop in the voltage across the DIAC. Applying Ohm's law an increase in current through a component should cause an increase in voltage across that component. However, the opposite effect is happening here, exhibiting negative resistance at break-over.

### 2.5 Control theory

Standard control theory suggests that there are two classic types of control methodologies namely feedforward and feedback control (Cervin et al., 2002:25). The input to a feedback controller is the same as what it is trying to control (the controlled variable is fed back into the controller). A sensory device measures the controlled variable, feeds it back to the controller which in turn adjusts the output. However, feedback control normally results in periods where the controlled variable is not at the desired set-point. This is where feedforward control can be an asset to the control process, as it could avoid the slowness of feedback control. Figure 15 illustrates how the feedback controller can be simplified. Evaluating feedforward control, disturbances are measured and potentially accounted for before they affect the system. A disadvantage of feedforward control lies in the fact that the effect of the disturbances needs to be predicted as accurately as possible, and all disturbances must be measured.
Achieving the advantages of both these methodologies, combinations of feedback and feedforward control is applied to control processes. Some examples include dead-time compensation and inverse response compensation. Dead-time compensation is used to control devices that take a long time to show any change to a change in input, for example the temperature rise inside an oven when it has been switched on.

Dead-time compensation control uses an element to predict how changes made by the controller will affect the controlled variable in the future. The controlled variable is also measured and used in feedback control. Inverse response compensation involves controlling systems where a change at first affects the measured variable one way, but later affects it in the opposite way.

A variety of applications use feedback control. The benefits of this control method are the possibility of keeping a parameter’s value at a fixed level or changing it quickly, despite disturbances that might occur. The basic goal of any controller is to achieve stability in controlling the process (Beardmore, 2006). This means that a controlled system should remain stable within its assigned parameters. The quality of the control can be measured by analysing the accuracy, speed and robustness of a control system.

In feedback control it is desirable to have the output signal follow the reference signal as precisely as possible. If the output signal of the process increases, the measurement signal needs to follow it. This results in the error signal and control signal becoming smaller. This forces the process output signal to decrease, which means that the error signal tends towards zero and the output of the process towards the reference value. Analysing the stability of a control system can be done with frequency response of an open-loop system.
Frequency response of an open-loop system indicates how the system behaves without the feedback loop. The structure of an open-loop system is presented in figure 16.

![Figure 16 An open-loop control system](image)

Frequency response is a function of $\omega$ (frequency) (Tham, 1999). The frequency function gives the gain (amplitude) and phase of a sinusoid signal, which is fed into the system, at every frequency of the output. The frequency response can be calculated from the transfer function of the system by replacing $s$ with $j\omega$ shown in figure 17.

$$u(t) = \sin \omega t$$

$$y(t) = |G(j\omega)| \sin(\omega t + \arg[G(j\omega)])$$

![Figure 17 Sinusoidal input and transfer function](image)

Drawbacks of open-loop control are that it requires perfect knowledge of the system and it assumes that there are no disturbances in the system.

### 2.5.1 Types of controllers

Many previous control valve systems were implemented using mechanical systems or solid state electronics. Pneumatics was often used to transmit information and apply control using pressure. However, most modern industrial control systems now rely on computers as the controller. Obviously it is much easier to implement complex control algorithms on a computer than using a mechanical system.

For feedback controllers there exists a few examples. The most common is a thermostat that just turns the power on if the temperature falls below a certain value and turns the power off if it exceeds a certain value. This is called bang-bang (BB) control (Dogan, 2002). Another simple type of controller is a proportional (P) controller, where, the controller output (control action) is proportional to the error in the measured variable. The error is defined as the difference between the current value (measured) and the desired value (set-point). If the
error is large, then the control action is large. This can be represented mathematically by equation 17:

\[ c(t) = K_c e(t) + c_s \] … (17)

Where \( e(t) \) = error value fed back  
\( K_c \) = controller's gain value  
\( c_s \) = steady state control value

It is necessary to maintain the variable at the steady state when there is no error. The gain \((K_c)\) will be positive if an increase in the input variable requires a decrease in the output variable (direct-acting control), and it will be negative if an increase in the input variable requires an increase in the output variable (reverse-acting control). A typical example of a reverse-acting system is controlling the flow of cooling water, if the temperature increases, the flow must be increased to maintain the desired temperature.

Although P-control is simple to understand, it has drawbacks. The biggest problem is that for most systems it will never entirely reduce the error. This is because when the error is zero, the controller only provides the steady state control action so the system will settle back to the original steady state. Having large gains can lead to system instability or can require physical impossibilities, such as infinitely large valves. In systems with high controller gains, instability is mediated primarily through overcompensation (Loannou & Kokotovic, 1984). Subsequently P-control is not capable of making the output equal to the reference and there will be steady-state deviation.

Another example of a controller is a proportional-integral (PI) controller which adds another term to the controller equation (see equation (18)).

\[ u(t) = K_c \left( e(t) + \frac{1}{T} \int_0^t e(t) dt \right) \] … (18)

Where \( T \) = integration time constant  
\( K_c \) = controller's gain value  
\( e(t) \) = error value

If the controller is tuned to be slow and \( T \) is large, then the controller first acts such as a P-controller. However as the integration starts to take affect, the steady-state deviation goes
slowly to zero. PI-control is therefore a control process where the value of the output signal is based on the error between the target value and the actual (measured) value of the controlled variable, as well as how long that error has existed.

The proportional-integral-derivative (PID) control can be defined as a control scheme whereby the signal that drives the actuator equals the weighted sum of the difference between the time integral of the difference, and time derivative of the difference between the input and the measured actual output (IEEE EED, 2000). Figure 18 shows a commercially available embedded PID-controller.

![Industrial PID controller](image)

Figure 18 Industrial PID controller (Nippon Instruments (India) pvt, 2011)

The equation of a PID-controller (equation (19)) has three terms, P, I and D-terms. The derivative term acts as a predictor, because the speed of change of the error signal affects the control signal. The derivative term has a large effect in systems where disturbances are present, because disturbances often occur quickly. This means that fast changes in an error signal might push the process into an unstable state. On the other hand, the derivative term might speed up the controlled system.

\[
    u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{d}{dt} e(t)
\]

... (19)

Where

- \( K_p \) = proportional gain
- \( K_i \) = integral gain
- \( K_d \) = derivative gain
- \( e(t) \) = error
The requirements set for a controller can be contradictory. A controlled close-loop system should be stable and the performance of the controller should be efficient, meaning that the control must be quick and reliable. For example, the better the performance of the controller, the less stable the close-loop system. That is why a compromise has to be made to fulfil all the requirements. The goal is to make the system fast, but not to push it close to the oscillation boundary, where small sudden changes in the system could make the system unstable.

The choosing of the controller’s parameters is process sensitive. One need to use different parameters in fast and slow processes. The disturbances also affect the choice of parameters. Techniques used for choosing the parameters used in P, PI, or PID-control, include the step response method or oscillation boundary method, which is based on frequency analysis. The step response method can be used to tune the parameters of a controller, if it is possible to feed a step into an open-loop system (no control or feedback).

If a step of height $K_1$ (in °C) is fed into the system and the output of the system behaves as in figure 19, then a tangent line can be drawn at the infliction point of the output signal. The height of the output signal is then $K_2$ (in °C), the delay time is $T_1$ (in seconds) and the rising time is $T_2$ (in seconds), which can be calculated using figure 19.

![Figure 19 Step response of a system](image-url)
Once values for $K_1$, $K_2$, $T_1$, and $T_2$ have been obtained, then the gain, integration time and derivative could be calculated for each controller by either the Ziegler-Nichols or Cohen equation tables. The Ziegler-Nichols method of calculating the parameters for P, PI or PID controllers is shown in table 5.

Table 5 Controller parameters for a step response method using Ziegler-Nichols (Dogan, 2002)

<table>
<thead>
<tr>
<th>Controller</th>
<th>Gain $K$</th>
<th>Integration Time $T_1$</th>
<th>Derivation Time $T_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>$\frac{K,T_i}{K,T_i}$</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>PI</td>
<td>$0.9\frac{K,T_i}{K,T_i}$</td>
<td>$3.3T_1$</td>
<td>Not applicable</td>
</tr>
<tr>
<td>PID</td>
<td>$1.2\frac{K,T_i}{K,T_i}$</td>
<td>$2T_i$</td>
<td>$0.5T_i$</td>
</tr>
</tbody>
</table>

2.6 Summary

In this chapter, an equivalent electrical circuit representing the properties of a thermal oven have been presented. Different types of thermal measuring techniques and devices have been discussed and control methods used in industry have been presented.

The next chapter will present characteristic data of the oven, a MATLAB model and algorithm. The progression from an analogue- to digital solution will be discussed and the power switch combination will be shown.
Chapter 3    Simulation and experimental validation

3.1 Introduction

This chapter will introduce data of a curing oven. An analysis of this data will be shown in order to verify the mathematical model in Chapter 2. It will further present the microcontroller logic and design of the combination switch.

3.2 Temperature measuring setup

In order to work towards the model presented in equation (12), measurements with respect to temperatures in and around the oven had to be obtained. Figure 20 presents the temperature measuring setup where five type K-thermocouples were used to obtained temperatures, which were interpreted by a PICOSCOPE data logger (TC-08).

The PICOSCOPE incorporates Seebeck coefficients for type J and K thermocouples and different coefficients can be used with specific thermocouples on specific channels. Each channel had been setup to interpret type K-thermocouples. All data collected was
transferred in real time to a personal computer for analysis and normalization. Thermocouples one to five represent:

- TC1 the ambient temperature,
- TC2 the insulation temperature,
- TC3 the heating element temperature,
- TC4 the oven air temperature and
- TC5 the object temperature.

A multimeter (FLUKE Fl223) was used to obtain readings of the AC voltage source applied to the element. Another multimeter (FLUKE Fl223 with AC current probe FLUKE FC212) was used to obtain readings of AC current drawn by the heating element. A variable AC transformer was used to limit the amount of energy applied to the heating element in order to keep the measurements within range of the test equipment as shown in figure 20.

### 3.3 Initial oven characteristic data

Using the equipment shown in figure 20, various data points were obtained and processed in graphical form using MATLAB 7.6.0 (see figure 21). The set-point temperature for the experiment was set to 684°C as this was within limits of the thermocouples and other test equipment. In order to analyse the data and obtain an algorithm for the curves, normalization had to be applied (see figure 22). This normalization will also be relevant to later mathematical models used to simulate the controller. Contrasting figure 21 and figure 22, and specifically the object temperature reveals that the heating element does store heat energy, which will be an important factor when designing the controller.

![Figure 21 Data obtained for the oven in the ON position](image)
Figure 22 Normalized data obtained for the oven in the ON position

To verify that the shape of the object temperature was not accidental, different shaped objects were placed inside the oven with the experiment repeated. Figure 23 shows different object temperatures measured over a period of 4500 seconds. This proves that the temperature curves follow the same basic shape and that applying curve analysis to this type of graph would be acceptable.

Figure 23 Different objects' temperature curves

Figure 24 shows different object temperatures with corresponding measurements of the oven’s air temperature.
The readings are those obtained from TC2 and TC3 shown in figure 20. This figure was also used to determine if a relationship exists between air and object temperatures.

Curve analysis is the process, by which the equivalent mathematical model is compared to the data of the experimental model. MATLAB was used to obtain a curve fitted mathematical model (see figure 25).

CURVE EXPERT a software package was used to verify results presented from MATLAB. The fitting software uses either standard mathematical models or models developed by the user. Figure 26 and figure 27 show the analysis done by CURVE EXPERT.
The blue line in figure 25 represents the mathematical model fitted to the data curve shown in green. The following mathematical model equation fits the data set with some residual.

\[ y = a(1 - e^{-bx}) \]  \hspace{1cm} \ldots (20)

Where

- \( y \) = temperature in °C
- \( a \) = constant as 9.75223
- \( b \) = constant as 1.93725
- \( x \) = time in seconds
The residuals from figure 25 and figure 27 are similar and therefore prove that the model used in figure 3 in Chapter 2 to represent the oven in its electrical form is correct. The derived equation (12) and the experimental acquired equation (20) are similar confirming the approach used is correct.

3.4 MATLAB model

The simplified basic model from Chapter 2 has been redrawn in MATLAB. This was done in order to test the fitted curve found as applied to the model for simulation purposes. The model shown in figure 28 was then later developed to include a control system, which could be simulated in MATLAB before applying it to the microcontroller for regulating the temperature of the oven.

3.4.1 Control of air temperature using Bang-Bang, PI and PID control

Using the Ziegler-Nichols criterion, the parameters of the controller can be calculated from the measurements taken. The parameters of the controller are presented in table 5.
MATLAB was used to implement a control method to the model as shown in figure 29. Using the Ziegler-Nichols step response technique, control parameters of the oven plant were determined and simulated in MATLAB for the PI control method. The PI control method was chosen because of its smaller over-shoot when nearing the set-point. The simulation setup used in MATLAB is shown in figure 29.

![Figure 29 MATLAB model of the simplified oven model and controller](image)

**3.4.2 Stability analysis of the analogue solution**

Using MATLAB’s SISO tools, a stability analysis was done using a step response method (see figure 30). As a result of a step input, the Open-Loop Bode and Root Locus results (see figure 31), a conclusion can be drawn that the control parameters for $K_p$ and $K_i$ (determined using table 5), for this specific model are within range and that the control loop is stable.
3.5 Progression from analogue to digital solution

According to Schoeman (2011), it is suggested that each control technique (BB, PI and PID) has its own individual advantages and disadvantages. The BB technique has a very fast rise time, but once the set-point is reached the variant between the switch on and switch off
temperatures is high. The PI control technique has a very stable and small variant when the set-point is reached, but its rise time towards the set-point is slow.

PID is an improvement on PI in respect that the rise time toward the set-point is much faster, but not as fast as BB (Schoeman, 2011). It also has a characteristic of over-shoot when the set-point is reached and the settling time is therefore increased when compared to PI control. This can be seen in figure 32, which was obtained using simulation models in PROTEUS VIRTUAL SYSTEM MODELLING (PVSM) 7.2. Different control methods were applied via embedded software programs using a microcontroller circuit (see figure 33).

PVSM 7.2 combines mixed mode SPICE circuit simulation, animated components and microprocessor models to facilitate co-simulation of complete microcontroller based designs (Labcenter Elctronics, 2010). It is possible to develop and test such designs before a physical prototype is constructed. At the heart of PVSM is ProSPICE. This is an established product that combines a SPICE3f5 analogue simulator kernel with a fast event-driven digital simulator to provide seamless mixed-mode simulation (Labcenter Elctronics, 2010). The use of a SPICE3f5 kernel allows for the use of the numerous manufacturer-supplied SPICE models, around 6000 of these are included with the package. PVSM 7.2 includes a number of virtual instruments including an oscilloscope, logic analyser, function generator, pattern generator, counter timer and virtual terminal as well as simple voltmeters and ammeters. In addition, dedicated Master/Slave/Monitor mode protocol analysers are provided for SPI and I²C communications.

Figure 32 Control methods’ step response graphs (Schoeman, 2011)
The parameters used \((K_1, K_2, T_1, T_2)\) in the microcontroller software were determined using Ziegler-Nichols’ method shown in table 5. In order to overcome the disadvantages of PI and BB control, it was decided to apply a combination control where the controller would initially use BB control to achieve a faster rise time towards the set-point and then switch over to PI control at 90% of the setpoint value.

Initial simulations showed very little improvement. However, once the PI parameters where updated during the BB control phase, the application of using two control methods showed a faster rise time as well as a faster settling time in comparison to only using PI control. Figure 34 shows the simulated graphs of the combination method with updated and non-updated PI parameters during the BB control phase.
3.5.1 Microcontroller program flow chart

According to Damij (2007) a flowchart is a good technique to represent the logical flow of a process that a program needs to follow. The microcontroller program was written in CCS C, a software package that uses the structure of C, but is optimised for microcontrollers. The program was developed for a PIC18F4220. Annexure B shows the program which was written in CCS C for a PIC microcontroller that follows the flowchart shown in figure 35.

![Flowchart Diagram]

**Figure 35** A flowchart outlining the functions of the CCS C software program

3.6 Design of a combination switch

As mentioned in Chapter 2, to counteract the disadvantages of a relay and of a TRIAC, a combination switch is used. Figure 36 shows this combination which encompasses a relay and a TRIAC to form a complete switching device.
The switching control circuit is very important, as it will activate the two devices in such a way that only the advantages of each device are realized. The timing of the switching actions will be controlled by the microcontroller shown in figure 33.

### 3.6.1 Switch simulation data

The simulation of the combination switch was initially done using MATLAB Simulink. Figure 37 shows the design where two SCRs form the circuit of a TRIAC device. The control circuit will need to switch the relay and the TRIAC on simultaneously. The TRIAC is a solid-state device which will switch on faster, as the relay’s coil must first be energised before enough magnetic energy is available to engage the contacts.
This means that the load current will first flow through the TRIAC, thereby reducing the arcing of the relay contacts during switch on. As the relay is energised, contact bounce will still occur but with minimal arcing, as the load current flows through the TRIAC. Figure 38 show the results of the MATLAB simulation where current is seen to shift from the TRIAC to the relay during switch on. However, the load current remains constant during the switch on period.

![MATLAB simulated results of switching currents in relay, TRIAC and load](image)

3.6.2 Switch practical results

The current through the TRIAC and relay was measured using an oscilloscope and two Fluke FC212 current probes, one measuring the relay contact current and the other the TRIAC current. Figure 39 shows when the contacts of the relay close (the resistance between them is less than that of the TRIAC’s internal resistance $R_f$ and $R_r$), current is transferred from the TRIAC to the relay contacts. As the contacts open the resistance between them increases so that it becomes more than that of the TRIAC, which causes
current to flow through the TRIAC. This occurs even during contact bounce, which is due to the mechanical forces between the relay contact plates. This transfer of current will continue until the contact bounce process settles, which was measured to be around 3 ms in this particular case.

![Figure 39 Measured current transfers between relay and TRIAC](image)

Figure 39 indicates the relay switching settling time. The settling time will differ from relay to relay, due to different coils and voltages used by various manufacturers.

![Figure 40 Relay settling duration](image)

Figure 40 Graph showing the settling duration.
3.7 Summary

This chapter presented the analog system, oven characteristic curves, mathematical algorithm in the electrical environment, and a digital controller. The simulation and practical results of a combination switch in order to prolong longevity were also shown. The next chapter describes the controller and software which were developed to control the oven for a preset curing profile, along with relevant results.
Chapter 4  Experimental results

4.1 Introduction

This chapter will present the practical setup used to obtain results, the software interface developed for use in the curing profile control and the hardware prototypes that were developed. Results will be evaluated and experiments repeated to determine stability of control parameters during curing profile runs.

4.2 Practical setup and curing control results

A practical setup in the laboratory was done using a PIC3 development board, a PIC18F4220, switching board, sensor board, a 3 kW oven and developed software as mentioned in the previous chapter. Figure 41 shows the configuration of the setup used to obtain the results.

![Figure 41 Practical setup of a 3 kW oven and switching board](image)

A graphical user interface (GUI) was developed in Microsoft Visual Basic 6 (annexure A), which allows the user to enter and save a desired curing profile. This is vital in order to ensure that the same curing profile could be applied at a later stage to the same type of material. Figure 42 and figure 43 shows the capability of the developed software that allow the creating and storing facility of a specific curing profile.
The blue dotted line in figure 44 represents the curing profile entered into the personal computer and acts as a visual representation of the curing profile required by the operator. The oven is initially set to full on. During this process, the PIC18F4220 microcontroller determines the control parameters as discussed using the Ziegler-Nichols method. Once this is done, the oven is ready to start the controlling process based on the predefined curing profile as in figure 44.
Figure 44 Curing profile saved in developed software

The time lapsed graph shown in figure 44, is updated every minute. The microcontroller updates the control parameters, taking an average of one hundred temperature readings, and determines the error with respect to the previous control parameters. This error is fed back into the control loop, ten times a minute, meaning that the control process has six seconds to read the temperature, average the readings, determine the new control parameters and apply it to the combination switch device.

4.3 Hardware

The software on the personal computer is used to provide a new set-point every minute that co-insides with the curing profile curve. It further visually shows the progress of the control process that is being applied to the oven with respect to the curing profile. The MAX6675 (annexure E) integrated circuit was used to convert the temperature readings from the thermocouple into a digital format to be used by the microcontroller. It has an accuracy of 0.5°C and a temperature range of -100°C to 2200°C. This device incorporates a coldjunction reference which is factory calibrated during the manufacturing process, thereby reducing the need for additional procedures and equipment during setup. Figure 45 shows the MAX6675 mounted on a prototype board for easy use with a common bread board.
Data is sent via an I²C bus to the microcontroller and interpreted as a digital word. This is used to update the control parameters as mentioned earlier. The switching board can now be activated. Both TRIAC and relay are energised, but because of the time taken to fully energise the coil of the relay, a delay is experienced between these two devices. This is ideal as the TRIAC will conduct the current first, and then the relay when the contacts come to rest. The switching board is shown in figure 46. Connections to the oven’s heating element are visible and the live wire is either closed or opened by either device.
Visual indications, in the form of two light emitting diodes (leds) are provided to show which device is activated. Combination switch activation was set to six seconds, in order to obtain the correct number of readings per minute as the processing time and communication between the temperature sensor and microcontroller is lengthy. Adapting the ON/OFF time of the switch in this way helped to achieve balance between activation time and delays due to processing.

Figure 47 shows a liquid crystal display (LCD) which is used to display the current set-point received from the personal computer and the average temperature measured during the processing time.

![Liquid crystal display showing setpoint and current temperature](image)

**Figure 47 Operator information display unit**

Figure 48 indicates the prototype controller (including microcontroller and a RS232 integrated circuit) which is used for communicating with the personal computer. The reason for using the RS232 communication protocol was due to the fact that it is included in the microcontroller, the PIC18F4220. This helps to reduce costs and minimize the total number of components required.

The control circuit uses a DC supply, but needs to switch an AC circuit. Therefore it was decided to isolate these two supply circuits. This was done by using an optocoupler driver (MOC3040 - see annexure F) which activated the TRIAC device. The relay provides its own isolation between the two circuits, as DC is applied to the coil, and AC to the relay contacts.
Calibration of the system was done using an infrared thermometer (LUTRON TM-2000). The thermometer uses an infrared beam to measure the amount of energy radiated by an object in order to determine its temperature. Figure 49 shows the infrared thermometer being used to measure the temperature of the thermocouple tip.

Figure 48 Prototype control unit with combination power switch

Figure 49 Infrared thermometer with laser target indicator, pointed at the thermocouple tip
Once calibration was complete the control parameters were automatically determined by the controller. A curing profile from the TSVUT was loaded in order to determine if the controller could follow it. Figure 50 shows the results of this process. A second curing profile was used to verify that the controller was following the loaded profile, as shown in figure 51.

![Figure 50 Predefined curing profile control results](image)

From figure 50 and figure 51 it can be seen that the curing oven could not be controlled during certain times of the cooling process. This was found due to the fact that the rate of required cooling was outside the limits of the ovens normal cooling characteristics. The requested cooling profile was much faster than the natural cooling of the oven, and thus the controller could not follow it. As no additional changes to the oven were allowed, it was deemed acceptable. A second run was completed in order to determine the validity of the results (see figure 52).

![Figure 51 The second curing profile control results](image)
The second run proved successful with a slight difference in the starting area of the profile, after which the controller effectively controlled the temperature as defined by the curing profile. Analysing a section of figure 52 using MATLAB indicates that there is still some over-shoot and under-shoot, as suggested by the simulated results.

The over-shoot was found to be less than 5% over the duration of the curing profile. This was found to be valid for both curing profiles, including the second confirmation run of the initial curing profile. Figure 53 shows a sample of the maximum over- and under-shoot of an area during the curing profile.
The over-shoot was measured after 534 minutes and was found to be 503.8°C at the 494°C set-point. Using the previous reading at 532 minutes of 487.5°C an over-shoot and under-shoot percentage can be calculated with equation (21) and equation (22).

\[
\% Err = \frac{T_{\text{measured}} - T_{\text{setpoint}}}{T_{\text{setpoint}}} \times 100 \quad \ldots \ (21)
\]

\[
\% Err = \frac{503.8°C - 494°C}{494°C} \times 100
\]

\[
% Err = 1.98%
\]

A 1.98% over-shoot exits.

\[
\% Err = \frac{T_{\text{measured}} - T_{\text{setpoint}}}{T_{\text{setpoint}}} \times 100 \quad \ldots \ (22)
\]

\[
% Err = \frac{494°C - 487.5°C}{494°C} \times 100
\]

\[
% Err = 1.31%
\]

A 1.31% under-shoot exists.

**4.4 Summary**

This chapter considered the hardware and software design of the controller. It showed results of the controlling process after calibration was completed. The controlling process was repeated using different profiles in order to establish consistency and validity. The error obtained for under- and over-shoots were shown.

Chapter 5 presents some conclusions and discusses a few recommendations that have come to light.
Chapter 5  Conclusions and recommendations

5.1 Concluding comments

This chapter presents the conclusions reached with regard to the design and development of an automated temperature controller for curing ovens. The initial objectives will be compared to the outcomes of the project. Recommendations for future research will then be highlighted.

5.2 Conclusions attained from the study

The theoretical study revealed that the most suitable sensor available for measuring high temperatures is a type K-thermocouple. Cold compensation was needed in order to provide the correct reference for calculations of temperature and the MAX6675 was therefore chosen due to its accuracy, cost effectiveness and availability. It has the added benefit of being able to convert analogue temperature voltages into a digital form for processing by the microcontroller. The thermal conversion had to be calibrated and an infrared thermometer (LUTRON TM2000) was used to achieve this. Due to high inrush currents during switching, the MAX6675 was initially found to be unstable. A filtering device on the input of the MAX6675 was subsequently included, which allowed the sensor to provide stable readings in digital format to the microcontroller for processing.

The design and development of the prototype controller was accomplished by using a PIC18F4220 microcontroller. The controller software was written using CCS C, a C programming language used for PIC microcontrollers. A PIC3 development board (see figure 48) was used to embed the software code in the PIC18F4220. The code is provided in annexure B.

The switching unit proved successful in harnessing the advantages of a TRIAC and a relay by reducing heat, noise and contact burn. However, if a faster control is required, then other switching devices need be considered, as the relay has a delayed activation time.

5.3 Recommendations

The use of RTD’s as sensory devices for low temperature applications (below 600°C), can be investigated in order to provide a simple interface to the microcontroller, thereby
eliminating the need for the MAX6675 device. This could reduce the cost of the overall development.

Different switching methods with respect to the technique used to drive the combination switch can be re-evaluated, in order to find another technique that will help to reduce transients during higher switching speeds.

Placing the sensory device inside the oven at different positions could help to ensure a better temperature representation of the heat inside the oven. This could lead to an enhanced representation of the object’s temperature inside the oven.

5.4 Other applications

The designed controller could also be used for:

- Determining and controlling temperatures in fuel cells; and
- Geyser control in solar geyser systems.

The combination switch could be used in power electronic applications, such as lighting and pump control.
References

List of sources cited


**List of sources consulted**


Annexure A  Visual Basic 6 curing profile control program

Public dintcnt As Integer
Public dintcnt1 As Integer
Public tmrcnt As Integer
Public tmrcnt1 As Integer
Public tmrcnt As Integer
Public Maxcnt As Integer
Private Sub Command1_Click()
    Dim a As Integer
    Exit Sub
    If lblCommport.Caption < "1" Then
        frmSplash.lblInfo.Caption = "Select Comm Port First ... 
        frmSplash.Show
        DoEvents
        aa = Timer
        bb = aa + 3
        Do While aa < bb
            DoEvents
            aa = Timer
        Loop
        frmSplash.Hide
        DoEvents
        Exit Sub
    End If
    DoEvents
    frmSplash.lblInfo.Caption = "Loading Curing Profile ... 
    frmSplash.Show
    DoEvents
    DoEvents
    Dim t As Double
    Dim inc As Double
    Maxcnt = 0
    ghCurve.Visible = False
    a = 0: inc = 1: t = 0
    t = 25                                            ' Room Temperature
    a = 1560 * 1.02                                    ' Total number of minutes
    ghCurve.ChartArea.Axes.Item(1).DataMax.Value = a
    ghCurve.ChartArea.Axes.Item(1).DataMax = a
    a = 0
    For a = 1 To 1560                                   ' Intervals in minutes
        If a = 1 Then inc = 0.729
        If a = 241 Then inc = 1
        If a = 961 Then inc = -0.792
    Next a
t = t + inc
ghCurve.ChartGroups(1).Data.Y(1, a) = t          ' Curing Plot
ghCurve.ChartGroups(2).Data.Y(1, a) = 2          ' Incoming Plot
If t > Maxcnt Then Maxcnt = t
Next a
ghCurve.ChartArea.Axes.Item(3).DataMax = Maxcnt + (Maxcnt * 0.05)
ghCurve.ChartArea.Axes.Item(2).DataMax = Maxcnt + (Maxcnt * 0.05)
ghCurve.Refresh
ghCurve.Visible = True
DoEvents
frmSplash.Hide
DoEvents
If lblCommport.Caption > "0" Then Command2.Enabled = True
End Sub
Private Sub Command2_Click()
If UCase(Command2.Caption) = "START CONTROL" Then
Command1.Enabled = False
Command2.Caption = "Stop Control"
MSComm1.CommPort = lblCommport.Caption
MSComm1.PortOpen = True
Label2.Caption = "Coms Port  Open ";
dincnt = 0
doutcnt = 0
tmrcnt = 0: tmrcnt1 = 0
stra = "25.00001"
Label3.Caption = stra
Label9.Caption = 1
MSComm1.Output = Trim(stra) & vbCrLf
tmrplot.Enabled = True
tmrComms.Enabled = True
tmrcoms = 0
Label14.Caption = Now
tmrplot_Timer
DoEvents
Exit Sub
End If
If UCase(Command2.Caption) = "STOP CONTROL" Then
Command1.Enabled = True
Command2.Caption = "Start Control"
stra = "02.00000"
DoEvents
Label10.Caption = "...................

DoEvents
 tmrComms.Enabled = False
tmrplot.Enabled = False
MSComm1.Output = Trim(stra) & vbCrLf
MSComm1.PortOpen = False
Label2.Caption = "Coms Port Closed"
din cnt = 1
dout cnt = 0
tmrcnt = 0: tmrcnt1 = 0
tmrcoms = 0
DoEvents
Exit Sub
End If
End Sub
Private Sub set-pointcomms()
End Sub
Private Sub Label16_Click()
End Sub
Private Sub mnuExit_Click(Index As Integer)
End
End Sub
Private Sub mnuLoad_Click()
CommonDialog1.Filter = "All Files (*.*)|*.|Text Files (*.txt)|*.txt|Batch Files (*.bat)|*.bat" ' Specify default filter.
CommonDialog1.FilterIndex = 2 ' Display the Open dialog box.
CommonDialog1.ShowOpen ' get a free file number
FileName = CommonDialog1.FileName
file1 = FreeFile ' open the file
If FileName = "" Then Exit Sub
DoEvents
frmSplash.lblInfo.Caption = "Loading Curing Profile ... "
frmSplash.Show
DoEvents
DoEvents
Maxcnt = 0
ghCurve.Visible = False
Open CommonDialog1.FileName For Input As file1 ' Add some text to the file
Input #file1, Profilename ' Specify default filter.
Input #file1, tempmax1
ghCurve.ChartArea.Axes.Item(3).DataMax = tempmax1
Input #file1, tempmax2
ghCurve.ChartArea.Axes.Item(2).DataMax = tempmax2
Input #file1, minmax
ghCurve.ChartArea.Axes.Item(1).DataMax = minmax
Loopin:
Input #file1, i, cp, mcp
If i = "" Or i = " " Then GoTo loopout
ghCurve.ChartGroups(1).Data.Y(1, i) = Val(cp)
ghCurve.ChartGroups(2).Data.Y(1, i) = Val(mcp)
If Val(i) = Val(minmax) Then GoTo loopout
GoTo Loopin
loopout:
Close #file1
ghCurve.Refresh
ghCurve.Visible = True
DoEvents
frmSplash.Hide
DoEvents
If lblCommport.Caption > "0" Then Command2.Enabled = True
CommonDialog1.FileName = ""
FileName = ""
End Sub
Private Sub mnuNew_Click(Index As Integer)
frmCPsetup.Show
End Sub
Private Sub mnuPort_Click()
frmPort.Show
End Sub
Private Sub mnuPrint_Click()
Dim PrintDc As Long
Dim Y As Long
Dim iHeight As Long
Dim iWidth As Long
Dim iMargin As Long
Dim iSpace As Long
Dim qInfo As String
Dim qInfo1 As String
Dim qResult As Integer
'Make sure the user really wants to print the charts
qInfo = "Is it ok to print to the following printer?"
qInfo1 = Printer.DeviceName
qInfo = qInfo + Chr$(13) + Chr$(13) + qInfo1 + Chr$(13)
qResult = MsgBox(qInfo, 4, "OK To Print?"
If qResult = 7 Then
Exit Sub
End If
Printer.Print "                          
Printer.Print "Curing Profile Name : " & lblProfileName.Caption  'Print header
Printer.Print "                          
Printer.Print "                          
Printer.Print "Date Started              : " & Label14.Caption
Printer.Print "Run Period                : " & Label11.Caption
Printer.Print "                          
Printer.Print "Printed on                  : " & Now
Printer.Print "                          "
Y = Printer.CurrentY + 10                   'Set position for line
Printer.Line (0, Y)-(Printer.ScaleWidth, Y) 'Draw a line across the page
iMargin = 1000 / Printer.TwipsPerPixelY + 400
iHeight = Printer.Height / Printer.TwipsPerPixelY - (iMargin * 2)
iWidth = Printer.Width / Printer.TwipsPerPixelX - iMargin
iSpace = 100 / Printer.TwipsPerPixelX
Result = ghCurve.DrawToDC(Printer.hDC, oc2dFormatEnhMetafile, oc2dScaleToFit, 0, Margin, iWidth, iHeight)
Printer.EndDoc
End Sub
Private Sub mnuReset_Click()
  Dim a As Integer
  DoEvents
  frmSplash.lblInfo.Caption = "Resetting Measured Curing Profile ... "
  frmSplash.Show
  DoEvents
  DoEvents
  Maxcnt = 0
  ghCurve.Visible = False
  For a = 1 To Val(ghCurve.ChartArea.Axes.Item(1).DataMax.Value)
    ghCurve.ChartGroups(2).Data.Y(1, a) = 2          ' Incoming Plot
  Next a
  ghCurve.Refresh
  ghCurve.Visible = True
  DoEvents
  frmSplash.Hide
  DoEvents
  If lblCommport.Caption > "0" Then Command2.Enabled = True
End Sub
Private Sub mnuSave_Click(Index As Integer)
  ' Set filters.
  CommonDialog1.Filter = "All Files (*.*)|*.*|Text Files (*.txt)|*.txt|Batch Files (*.bat)|*.bat"    ' Specify default filter.
  CommonDialog1.FilterIndex = 2    ' Display the Open dialog box.
  CommonDialog1.ShowSave    ' get a free file number
  FileName = CommonDialog1.FileName
  file1 = FreeFile    'open the file
  If FileName = "" Then Exit Sub
  Open CommonDialog1.FileName For Output As file1    ' Add some text to the file
  Print #file1, lblProfileName.Caption
  Print #file1, (Val(ghCurve.ChartArea.Axes.Item(3).DataMax) * 1.05)
  Print #file1, (Val(ghCurve.ChartArea.Axes.Item(2).DataMax) * 1.05)
  Print #file1, Val(ghCurve.ChartArea.Axes.Item(1).DataMax)
  For i = 1 To Val(ghCurve.ChartArea.Axes.Item(1).DataMax)
    If i = 1561 Then
      a = a
    End If
    If Val(ghCurve.ChartGroups(1).Data.Y(1, i)) < 2 Then ghCurve.ChartGroups(1).Data.Y(1, i) = 2
  Next i
  Close file1
End Sub
If Val(ghCurve.ChartGroups(2).Data.Y(1, i)) < 2 Then ghCurve.ChartGroups(2).Data.Y(1, i) = 2
If Val(ghCurve.ChartGroups(2).Data.Y(1, i)) = 1E+308 Then ghCurve.ChartGroups(2).Data.Y(1, i) = 2
Print #file1, i & vbTab & "," & ghCurve.ChartGroups(1).Data.Y(1, i) & vbTab & "," & ghCurve.ChartGroups(2).Data.Y(1, i)
Next i
Close #file1
ghCurve.Refresh
ghCurve.Visible = True
DoEvents
DoEvents
CommonDialog1.FileName = ""
FileName = ""
End Sub
Private Sub MSComm1_oncomm()
Dim str As String
Dim stra As String
Dim bufin
Dim bufins
Select Case Me.MSComm1.CommEvent
Case comEvReceive
' This is used when data is received
stra = MSComm1.Input
Label4.Caption = stra
dincnt = dincnt + 1
Label8.Caption = dincnt
tmrcoms = 0
DoEvents
If dinct = 10000 Then dinct = 2
MSComm1.InBufferCount = 0
End Select
End Sub
Private Sub tmrComms_Timer()
tmrcoms = tmrcoms + 1
TimetoAdd = TimeValue(Now) - TimeValue(Label14.Caption)
Label11.Caption = Format(TimetoAdd, "hh:mm:ss ")
If tmrcoms = 57 Then
tmrComms.Enabled = False
tmrplot.Enabled = False
Command2_Click
frmSplash.lblInfo.Caption = "Comms Lost....... 
frmSplash.Show
DoEvents
aa = Timer
bb = aa + 2
Do While aa < bb
DoEvents  ' Yield to other processes.
aa = Timer
Loop
frmSplash.Hide
DoEvents
DoEvents
Command2.SetFocus
End If
DoEvents
End Sub
Private Sub tmrplot_Timer()
Dim str, stra
TimetoAdd = TimeValue(Now) - TimeValue(Label14.Caption)
Label11.Caption = Format(TimetoAdd, "hh:mm:ss ")
stra = Label4.Caption
Label10.Caption = Val(Label10.Caption) + 1
For i = 1 To 10
If Mid(stra, i, 1) = "," Then GoTo hop1
Next i
hop1:
str = Val(Mid(stra, 1, i - 1))
Label7.Caption = str
If Val(str) > 2 And Val(str) < 1101 Then
ghCurve.ChartGroups(2).Data.Y(1, Val(Label10.Caption)) = str
ghCurve.Refresh
DoEvents
End If
If Val(Label10.Caption) < 2 Then
If Val(Label7.Caption) < (Val(Label3.Caption) - 0.5) Then
Label10.Caption = Val(Label10.Caption) - 1
tmrcnt = 0
'GoTo jj
End If
End If
End If
End If
End Sub
End If
End If
End If
End If
End Sub
str = Val(str) + 0.00001
str = Val(str)
hop11:
stra = "00000001"
Mid(stra, 1, Len(str)) = str
Label3.Caption = stra
Label9.Caption = doutcnt
If Command2.Caption = "STOP CONTROL" Then
MSComm1.Output = Trim(stra) & vbCr
If Val(Label3.Caption) > Val(Label7.Caption) - 0.5 Then
Else
din cnt = 1: doutcnt = 1
Exit Sub
End If
End If
End If
tmrcnt1 = 111
End If
End If
If tmrcnt = 1 Then
doutcnt = doutcnt + 1
If Command2.Enabled = True Then
If doutcnt + 1 > 9363 Then doutcnt = doutcnt - 10
str = Mid((ghCurve.ChartGroups(1).Data.Y(1, doutcnt)), 1, 8)
str = Round(str, 2)
For u = 1 To Len(str)
rr = Mid(str, u, 1)
If rr = "." Then GoTo hop12
Next u
str = Val(str) + 0.00001
str = Val(str)
hop12:
stra = "00000001"
Mid(stra, 1, Len(str)) = str
Label3.Caption = stra
Label9.Caption = doutcnt
MSComm1.Output = (stra) & vbCr
tmrcnt = 0: tmrcnt1 = 111
End If
End If
DoEvents
End Sub
Annexure B  PIC18F4220 CCS C program

#include "main.h"
#include <stdlib.h>
#include <float.h>
#include "flex_lcd.c"
#include "max6675.c"
#include "PIparameterCalc.c"

char Received[9];
char msg[8];
int16 count,cnt1,cnt2,cnt3,dd;
float D,Td,T1,K,T,Kp,Ti,a,b,c;
float value1,value,mult,set-point,rkt,ekt,pkt,pkt1,ukt,qkt,max,min,pkt_1,ekt_1;

#define rda

void handle_data(void)
{
    gets(received);
    set-point=atof(received);
    if (set-point<1)
    {
        //set-point=0;
    }
    else
    {
        rkt=set-point;
    }
}

void main_on()
{
    output_high(PIN_a1);
    delay_ms(1);
    output_high(PIN_a2);
    delay_ms(30);
    output_low(PIN_a1);
    delay_ms(1);
}

void main_off()
{
    output_high(PIN_a1);
    delay_ms(1);
    output_low(PIN_a2);
    delay_ms(30);
    output_low(PIN_a1);
    delay_ms(1);
}

void main()
{
WDT_ON;
    setup_wdt(WDT_2304MS);
setup_adc_ports(NO_ANALOGS|VSS_VDD);
setup_adc(ADC_OFF|ADC_TAD_MUL_0);
setup_psp(PSP_DISABLED);
setup_spi(SPI_SS_DISABLED);
setup_wdt(WDT_OFF);
setup_timer_0(RTCC_INTERNAL|RTCC_DIV_16|RTCC_8_bit);
setup_timer_1(T1_DISABLED);
setup_timer_2(T2_DISABLED,0,1);
setup_timer_3(T3_DISABLED,T3_DIV_BY_1);
setup_comparator(NC_NC_NC_NC);
setup_vref(FALSE);
enable_interruptions(INT_RDA);
enable_interruptions(GLOBAL);
setup_oscillator(OSC_8MHZ|OSC_TIMER1|OSC_31250|OSC_PLL_OFF);
cnt1=0;cnt2=0;cnt3=0;
lcd_init();
lcd_putchar("Good day\n");
delay_ms(950);
lcd_putchar("Oven Control\n");
delay_ms(950);
lcd_gotoxy(1,2);
lcd_putchar("Temp =       C");
T1=2400000;
T=5000;                                  //Sample time in miliseconds
Td=1000;
K=680;
Kp=(0.9*T1)/(K*Td);
T=3.3*Td;
//D=0.5*Td;
a=Kp;
b=Kp*(T/Ti);
//c=(Kp*D)/T;
max=1000-60-49;                          //maximum on time of heater in miliseconds, 60
miliseconds for the ON/OFF of the switch and 80 milliseconds for the comms
min=1;                                    //minimum off time of heater in miliseconds
// define set-point in Degrees C
set-point=2.00;
rkt=set-point;
pkt_1=0.0;
ekt_1=1.0;
count=0;
value=0;
dd=0;
do
hop1:
value1=0; cnt3=0; value=0; mult=0.49995;
restart_wdt();
for (cnt3=1;cnt3<=7;++cnt3)
{
    value=do_everything();
    value=value-49;
    value1=value1+value;
}
value=value1/7;
if ((value>1) & (value<119)) { mult=0.5060;}  // 0 Deg to 60 Deg
if ((value>120) & (value<262)) { mult=0.4998;}  // 60.25 Deg to 130 Deg
if ((value>263) & (value<4003)) { mult=0.5029;}  // 130.25 Deg to 190 Deg
value=value*mult;
value=value+10.00;
sprintf(msg,"%.8.2f",value);
    // calculate error
ekt=rkt-value;
    // calculate i term
pkt=(b*ekt)+pkt_1;
    // calculate p term
pktp=(a*ekt);
    // calculate d term
    //qkt=c*(ekt-ekt_1);
    // calculate pid output
ukt=pkt+pktp; //+qkt;
    // check min and max limits
if (ukt>=max)
{
    pkt=pkt_1;
    ukt=max;
}
else if (ukt<=min)
{
    pkt=pkt_1;
    ukt=min;
}
    // LCD OUTPUT
lcd_gotoxy(8,2);lcd_putchar(msg[1]="\nlcd_gotoxy(9,2);lcd_putchar(msg[2]="\n;
lcd_gotoxy(10,2);lcd_putchar(msg[3]="\n;
lcd_gotoxy(11,2);lcd_putchar(msg[4]="\n;
lcd_gotoxy(12,2);lcd_putchar(msg[5]="\n;
lcd_gotoxy(13,2);lcd_putchar(msg[6]="\n;
lcd_gotoxy(14,2);lcd_putchar(msg[7]="\n;
sprintf(msg,"%.8.2f",set-point);
lcd_gotoxy(17,1);lcd_putchar("SetP = % ");
lcd_gotoxy(24,1);lcd_putchar(msg[1]="\n;
lcd_gotoxy(25,1);lcd_putchar(msg[2]="\n;
lcd_gotoxy(26,1);lcd_putchar(msg[3]="\n;
lcd_gotoxy(27,1);lcd_putchar(msg[4]="\n;
lcd_gotoxy(28,1);lcd_putchar(msg[5]="\n;
lcd_gotoxy(29,1);lcd_putchar(msg[6]="\n;")
lcd_gotoxy(30,1);lcd_putstr(msg[7]);msg[7]="";
//      LCD OUTPUT
//      RS232   OUTPUT
printf("%08.2f",value);printf(" , ");
//printf("%08.2f",value1);printf(" , ");
printf("%09.2f",ekt);printf(" , ");
printf("%09.2f",pkt);printf(" , ");
printf("%09.0f",pktp);printf(" , ");
printf("%09.0f",qkt);printf(" , ");
printf("%09.0f",ukt);printf(" , ");
printf("%09.0f",(max-ukt));printf(" : ");
//printf(" \r\n");
//      RS232   OUTPUT

// send control for heater switching
if (value<(rkt-3))
{
    main_on();
    cnt1=0;
    delay_ms(1000);
    cnt1=0;
    main_off();
    delay_ms(1);
    cnt1=0;
}
if (value>(rkt+2))
{
    main_off();
    cnt1=0;
}
if (rkt<3)
{
    output_low(PIN_a1);
    output_low(PIN_a2);
    cnt1=0;
}
if (ukt<1)
{
    cnt1=0;
    main_off();
goto hop2;
}
main_on();
    cnt1=0;
    delay_ms((int16)ukt);
    cnt1=0;
    main_off();
ay_ms((int16)(max-ukt));
cnt1=0;
hop2:

    //save variables for use in next loop
    pkt_1=pkt;
    ekt_1=ekt;
    output_low(PIN_a1);
    output_low(PIN_a2);
    value=0;
} while (dd==0); //While end

Include Files used by the controller

<main.h>
<stdlib.h>
<float.h>
<flex_lcd.c>
<max6675.c>
<P1parameterCalc.c>
The temperature profile during the curing process of composite materials determines the final characteristics of the cured sample. The problem however of reproducing the same set of parameters with respect to different temperature settings for specific time periods makes for a control device to be present. Curing composite materials is normally done in an industrial oven. A heating element of low resistance generates heat which is then transferred to the material or object. This can be seen in figure 1 below, where heat from the heating element is transferred to the object inside the oven. Figure 1 Representation of electrical oven as a plant 2.2 Oven electrical equivalent model Heat is the process of energy transfer from one body or system to another due to a difference in temperature. Thermal energy can be defined as the energy of a body which increases with its temperature. Energy transfer by heat can occur between objects by radiation, conduction and convection (Kesidou and Duit 1993). Temperature can be used as a measure of the internal energy. Analysis of the heat flow in the oven can either be done by means of thermodynamics or by using and electrical analogy of the heat flow path. Heat flow can be modelled by an analogy to an electrical circuit seen in table 1, where heat flow is represented by current, temperatures are represented by voltages, heat sources are represented by constant current sources, thermal resistances are represented by resistors and thermal capacitances by capacitors. Table 1 Equivalence between thermal and electrical entities Thermal quantity Unit Electrical quantity Unit P, Heat flow, power W, I, Current flow A, Q, Power, Energy V, Voltage V, R, Resistance Ω, C, Capacitance F, τ, Time constant s T = R°C, Electrical RC constant s This can be seen in figure 2 were R1 represents the thermal resistance between the heating element and the oven, R2 represents the resistance between the oven and the environment, C1 represents the heat capacity of the element, C2 represents the heat capacity of the oven, 8a represents the ambient temperature of the environment, W represents the power dissipated in the heating elements, θv the oven temperature and θh represents the heating element temperature. The capacitor C1 can be neglected from the equivalent diagram as the capacity of the element to store heat is very little in comparison to that of the oven capacity to store heat. Figure 2 Electrical equivalent of thermal oven Therefore figure 3 shows the diagram that can be used as the electrical equivalent of a thermal oven with input voltage vi(t) input current i(t) and output voltage vo(t). Figure 3 Electrical equivalent of thermal oven, simplified Finding the transfer function of this circuit now becomes simplified, by applying basic electrical principals of current and voltage rules as well as the relationships between voltage, current and impedance as can be seen in table 2. Table 2 Voltage and current and current and voltage relationships, summarized. The time domain equation can be found by applying Kirchhoff's voltage law. (1) Now by using table two the current flowing through the capacitor can be written (2) By substituting (2) in (1) all terms in
Annexure E  MAX6675 datasheet

**General Description**

The MAX6675 performs cold-junction compensation and digitizes the signal from a type-K thermocouple. The data is output in a 12-bit resolution, SPI™-compatible, read-only format.

This converter resolves temperatures to 0.25°C, allows readings as high as +1024°C, and exhibits thermocouple accuracy of ±1.5°C for temperatures ranging from 0°C to +700°C.

The MAX6675 is available in a small, 8-pin SO package.

**Applications**

- Industrial
- Appliances
- HVAC
- Automotive

**Features**

- Direct Digital Conversion of Type-K Thermocouple Output
- Cold-Junction Compensation
- Simple SPI-Compatible Serial Interface
- 12-Bit, 0.25°C Resolution
- Open Thermocouple Detection

**Ordering Information**

<table>
<thead>
<tr>
<th>PART</th>
<th>TEMP RANGE</th>
<th>PIN-PACKAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX6675/TA</td>
<td>-40°C to +185°C</td>
<td>8 SO</td>
</tr>
</tbody>
</table>

**Pin Configuration**

![PIN CONFIGURATION Diagram]

**Typical Application Circuit**

![TYPICAL APPLICATION CIRCUIT Diagram]

For pricing, delivery, and ordering information, please contact Maxim/Dallas Direct! at 1-888-629-4642, or visit Maxim's website at www.maxim-ic.com.
Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

**ABSOLUTE MAXIMUM RATINGS**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply Voltage (Vcc to GND)</td>
<td>VCC</td>
</tr>
<tr>
<td>-3V to +5V</td>
<td></td>
</tr>
<tr>
<td>SO, SD, SS, T1, T2, T3 to GND</td>
<td>VCC</td>
</tr>
<tr>
<td>-0.5V to +3.3V</td>
<td></td>
</tr>
<tr>
<td>SO Current</td>
<td>Icc</td>
</tr>
<tr>
<td>-10mA</td>
<td></td>
</tr>
<tr>
<td>ESD Protection (Human Body Model)</td>
<td>Vcc</td>
</tr>
<tr>
<td>±2000V</td>
<td></td>
</tr>
<tr>
<td>Continuous Power Dissipation (Tamb = +70°C)</td>
<td>Pd</td>
</tr>
<tr>
<td>±200W</td>
<td></td>
</tr>
<tr>
<td>Operating Temperature Range</td>
<td>Tamb</td>
</tr>
<tr>
<td>-20°C to +85°C</td>
<td></td>
</tr>
<tr>
<td>Junction Temperature</td>
<td>Tamb</td>
</tr>
<tr>
<td>-68°C to +150°C</td>
<td></td>
</tr>
<tr>
<td>Storage Temperature Range</td>
<td>Tamb</td>
</tr>
<tr>
<td>-68°C to +150°C</td>
<td></td>
</tr>
<tr>
<td>Vapor Phase (50%)</td>
<td>Vph</td>
</tr>
<tr>
<td>±215°C</td>
<td></td>
</tr>
<tr>
<td>Infrared (15s)</td>
<td>Vph</td>
</tr>
<tr>
<td>±220°C</td>
<td></td>
</tr>
<tr>
<td>Lead Temperature (soldering, 10s)</td>
<td>Vph</td>
</tr>
<tr>
<td>±300°C</td>
<td></td>
</tr>
</tbody>
</table>

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

**ELECTRICAL CHARACTERISTICS**

(VCC = +3.3V to +5.5V, Tamb = -20°C to +85°C, unless otherwise noted; Typical values specified at +25°C.) (Note 1)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Conditions</th>
<th>MIN</th>
<th>TYP</th>
<th>MAX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature Error</td>
<td></td>
<td><strong>THERMOCOUPLE = +700°C, Tamb = +25°C (Note 2)</strong></td>
<td>Vcc = +3.3V</td>
<td>-5</td>
<td>+5</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>THERMOCOUPLE = +600°C, Tamb = +25°C (Note 2)</strong></td>
<td>Vcc = +3.3V</td>
<td>-6</td>
<td>+6</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>THERMOCOUPLE = +0°C to +100°C, Tamb = +25°C (Note 2)</strong></td>
<td>Vcc = +3.3V</td>
<td>-8</td>
<td>+8</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>THERMOCOUPLE = +100°C to +1000°C, Tamb = +25°C (Note 2)</strong></td>
<td>Vcc = +3.3V</td>
<td>-17</td>
<td>+17</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Vcc = +3.3V</strong></td>
<td>Vcc = +3.3V</td>
<td>-19</td>
<td>+19</td>
</tr>
<tr>
<td>Thermocouple Conversion Constant</td>
<td></td>
<td></td>
<td></td>
<td>10.25</td>
<td></td>
</tr>
<tr>
<td>Cold Junction Compensation Error</td>
<td></td>
<td><strong>Tamb = -20°C to +85°C (Note 2)</strong></td>
<td>Vcc = +3.3V</td>
<td>-3.0</td>
<td>+3.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Vcc = +5V</strong></td>
<td>Vcc = +5V</td>
<td>-3.0</td>
<td>+3.0</td>
</tr>
<tr>
<td>Resolution</td>
<td></td>
<td></td>
<td></td>
<td>0.25</td>
<td>°C</td>
</tr>
<tr>
<td>Thermocouple Input Impedance</td>
<td></td>
<td></td>
<td></td>
<td>60</td>
<td>kΩ</td>
</tr>
<tr>
<td>Supply Voltage</td>
<td>Vcc</td>
<td></td>
<td></td>
<td>3.0</td>
<td>±0.5 V</td>
</tr>
<tr>
<td>Supply Current</td>
<td>Icc</td>
<td></td>
<td></td>
<td>0.7</td>
<td>1.5 mA</td>
</tr>
<tr>
<td>Power-On Reset Threshold</td>
<td>Vcc</td>
<td></td>
<td></td>
<td>1</td>
<td>2.5 V</td>
</tr>
<tr>
<td>Power-On Reset Hysteresis</td>
<td></td>
<td></td>
<td></td>
<td>50</td>
<td>mV</td>
</tr>
<tr>
<td>Conversion Time</td>
<td>Vcc</td>
<td></td>
<td></td>
<td>0.17</td>
<td>0.22 s</td>
</tr>
</tbody>
</table>

**SERIAL INTERFACE**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Conditions</th>
<th>MIN</th>
<th>TYP</th>
<th>MAX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Low Voltage</td>
<td>VIL</td>
<td>0.3 x Vcc</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input High Voltage</td>
<td>VIH</td>
<td>0.7 x Vcc</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input Leakage Current</td>
<td>ILEAK</td>
<td>Vcc = GND or Vcc</td>
<td>±5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input Capacitance</td>
<td>Cin</td>
<td>1 pF</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

ELECTRICAL CHARACTERISTICS (continued)

(VCC = ±3.3V to ±5.5V, TA = -25°C to +85°C, unless otherwise noted. Typical values specified at +25°C.) (Note 1)

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>SYMBOL</th>
<th>CONDITIONS</th>
<th>MIN</th>
<th>TYP</th>
<th>MAX</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output High Voltage</td>
<td>VOH</td>
<td>ISOURCE = 1.6mA</td>
<td>VCC - 0.4</td>
<td>V</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Output Low Voltage</td>
<td>VOL</td>
<td>ISM = 1.6mA</td>
<td>0.4</td>
<td>V</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TIMING

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>SYMBOL</th>
<th>CONDITIONS</th>
<th>MIN</th>
<th>TYP</th>
<th>MAX</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Clock Frequency</td>
<td>fSCK</td>
<td></td>
<td>4.3</td>
<td>MHz</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCK Pulse High Width</td>
<td>tCH</td>
<td></td>
<td>100</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCK Pulse Low Width</td>
<td>tCL</td>
<td></td>
<td>100</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSB Fall to SCK Rise</td>
<td>tCSS</td>
<td>CCL = 10pF</td>
<td>100</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSB Fall to Output Enable</td>
<td>tDV</td>
<td>CCL = 10pF</td>
<td>100</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSB Rise to Output Disable</td>
<td>tm</td>
<td>CCL = 10pF</td>
<td>100</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSB Fall to Output Data Valid</td>
<td>tDO</td>
<td>CCL = 10pF</td>
<td>100</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note 1: All specifications are 100% tested at TA = ±25°C. Specification limits over temperature (TA = TMIN to TMAX) are guaranteed by design and characterization, not production tested.

Note 2: Guaranteed by design. Not production tested.

Typical Operating Characteristics

(VCC = ±3.3V, TA = ±25°C, unless otherwise noted.)
Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

**Pin Description**

<table>
<thead>
<tr>
<th>PIN</th>
<th>NAME</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GND</td>
<td>Ground</td>
</tr>
<tr>
<td>2</td>
<td>T-</td>
<td>Alumel Lead of Type-K Thermocouple. Should be connected to ground externally.</td>
</tr>
<tr>
<td>3</td>
<td>T+</td>
<td>Chromel Lead of Type-K Thermocouple</td>
</tr>
<tr>
<td>4</td>
<td>VCC</td>
<td>Positive Supply. Bypass with a 0.1μF capacitor to GND.</td>
</tr>
<tr>
<td>5</td>
<td>SCK</td>
<td>Serial Clock Input</td>
</tr>
<tr>
<td>6</td>
<td>CS</td>
<td>Chip Select. Set CS low to enable the serial interface</td>
</tr>
<tr>
<td>7</td>
<td>SD</td>
<td>Serial Data Output</td>
</tr>
<tr>
<td>8</td>
<td>N.C</td>
<td>No Connection</td>
</tr>
</tbody>
</table>

**Detailed Description**

The MAX6675 is a sophisticated thermocouple-to-digital converter with a built-in 12-bit analog-to-digital converter (ADC). The MAX6675 also contains cold-junction compensation sensing and correction, a digital controller, an SPI-compatible interface, and associated control logic.

The MAX6675 is designed to work in conjunction with an external microcontroller (μC) or other intelligence in thermostatic, process-control, or monitoring applications.

**Temperature Conversion**

The MAX6675 includes signal-conditioning hardware to convert the thermocouple’s signal into a voltage compatible with the input channels of the ADC. The T+ and T- inputs connect to internal circuitry that reduces the introduction of noise errors from the thermocouple wires. Before converting the thermoelectric voltages into equivalent temperature values, it is necessary to compensate for the difference between the thermocouple cold-junction side (MAX6675 ambient temperature) and a 0°C virtual reference. For a type-K thermocouple, the voltage changes by 41μV/°C, which approximates the thermocouple characteristic with the following linear equation:

\[ V_{OUT} = (41\mu V/°C) \times (T_R - T_{AMB}) \]

**Cold-Junction Compensation**

The function of the thermocouple is to sense a difference in temperature between two ends of the thermocouple wires. The thermocouple’s hot junction can be read from 0°C to +1023.75°C. The cold end (ambient temperature of the board on which the MAX6675 is mounted) can only range from -20°C to +85°C. While the temperature at the cold end fluctuates, the MAX6675 continues to accurately sense the temperature difference at the opposite end.

The MAX6675 senses and corrects for the changes in the ambient temperature with cold-junction compensation. The device converts the ambient temperature reading into a voltage using a temperature-sensing diode. To make the actual thermocouple temperature measurement, the MAX6675 measures the voltage from the thermocouple’s output and from the sensing diode. The device’s internal circuitry passes the diode’s voltage (sensing ambient temperature) and thermocouple voltage (sensing remote temperature minus ambient temperature) to the conversion function stored in the ADC to calculate the thermocouple’s hot-junction temperature.

Optimal performance from the MAX6675 is achieved when the thermocouple cold junction and the MAX6675 are at the same temperature. Avoid placing heat-generating devices or components near the MAX6675 because this may produce cold-junction-related errors.

**Digitization**

The ADC adds the cold-junction diode measurement with the amplified thermocouple voltage and reads out the 12-bit result onto the SO pin. A sequence of all zeros means the thermocouple reading is 0°C. A sequence of all ones means the thermocouple reading is +1023.75°C.
Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

Applications Information

Serial Interface
The Typical Application Circuit shows the MAX6675 interfaced with a microcontroller. In this example, the MAX6675 processes the reading from the thermocouple and transmits the data through a serial interface. Force CS low and apply a clock signal at SCK to read the results at SO. Forcing CS low immediately stops any conversion process. Initiate a new conversion process by forcing CS high.

Force CS low to output the first bit on the SO pin. A complete serial interface read requires 16 clock cycles. Read the 16 output bits on the falling edge of the clock. The first bit, D15, is a dummy sign bit and is always zero. Bits D14-D3 contain the converted temperature in the order of MSB to LSB. Bit D2 is normally low and goes high when the thermocouple input is open. D1 is low to provide a device ID for the MAX6675 and bit D0 is three-state.

Figure 1a is the serial interface protocol and Figure 1b shows the serial interface timing. Figure 2 is the SO output.

Open Thermocouple
Bit D2 is normally low and goes high if the thermocouple input is open. In order to allow the operation of the open thermocouple detector, T must be grounded. Make the ground connection as close to the GND pin as possible.

Noise Considerations
The accuracy of the MAX6675 is susceptible to power-supply coupled noise. The effects of power-supply noise can be minimized by placing a 0.1µF ceramic bypass capacitor close to the supply pin of the device.

Thermal Considerations
Self-heating degrades the temperature measurement accuracy of the MAX6675 in some applications. The magnitude of the temperature errors depends on the thermal conductivity of the MAX6675 package, the mounting technique, and the effects of airflow. Use a large ground plane to improve the temperature measurement accuracy of the MAX6675.

The accuracy of a thermocouple system can also be improved by following these precautions:
- Use the largest wire possible that does not shunt heat away from the measurement area.
- If small wire is required, use it only in the region of the measurement and use extension wire for the region with no temperature gradient.
- Avoid mechanical stress and vibration, which could strain the wires.
- When using long thermocouple wires, use a twisted-pair extension wire.
- Avoid steep temperature gradients.
- Try to use the thermocouple wire well within its temperature rating.
- Use the proper sheathing material in hostile environments to protect the thermocouple wire.
- Use extension wire only at low temperatures and only in regions of small gradients.
- Keep an event log and a continuous record of thermocouple resistance.

Reducing Effects of Pick-Up Noise
The input amplifier (A1) is a low-noise amplifier designed to enable high-precision input sensing. Keep the thermocouple and connecting wires away from electrical noise sources.

Chip Information
TRANSISTOR COUNT: 6720
PROCESS: BICMOS
Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

Figure 1a. Serial Interface Protocol

Figure 1b. Serial Interface Timing

<table>
<thead>
<tr>
<th>BIT</th>
<th>DUMMY/ SIGN BIT</th>
<th>13-BIT TEMPERATURE READING</th>
<th>THERMOCOUPLE INPUT</th>
<th>DEVICE ID</th>
<th>STATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>14</td>
<td>13</td>
<td>12</td>
<td>11</td>
<td>10 5</td>
</tr>
<tr>
<td>0</td>
<td>MSB</td>
<td>0</td>
<td>15</td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 2. SO Output
Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

Block Diagram

MAX6675
Annexure F  MOC3040 datasheet

**ZERO VOLTAGE CROSSING**

**OPTICALLY ISOLATED TRIAC DRIVER**

This device consists of a gallium arsenide infrared emitting diode optically coupled to a monolithic silicon detector performing the function of a Zero Voltage Crossing unilateral triac driver.

They are designed for use with a triac in the interface of logic systems to equipment powered from 220 Vac lines, such as solid-state relays, industrial controls, motors, solenoids, and consumer appliances, etc.

- Simplifies Logic Control of 220 Vac Power
- Zero Voltage Crossing
- High Breakdown Voltage: \( V_{DRM} = 400 \text{ V Min} \)
- High Isolation Voltage: \( V_{ISO} = 7500 \text{ V Min} \)
- Small, Economical, 6-Pin DIP Package
- Same Pin Configuration as MOC3020/3021
- UL Recognized, File No. E54915
- dv/dt of 100 V/μs Typ

### MAXIMUM RATINGS (\( T_A = 25^\circ \text{C} \) unless otherwise noted)

<table>
<thead>
<tr>
<th>Rating</th>
<th>Symbol</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>INFRARED EMITTING DIODE MAXIMUM RATINGS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reverse Voltage</td>
<td>( V_R )</td>
<td>60</td>
<td>Volts</td>
</tr>
<tr>
<td>Forward Current – Continuous</td>
<td>( I_f )</td>
<td>50</td>
<td>mA</td>
</tr>
<tr>
<td>Total Power Dissipation @ ( T_A = 25^\circ \text{C} )</td>
<td>( P_D )</td>
<td>1.20</td>
<td>mW</td>
</tr>
<tr>
<td>Negligible Power in Quiescent Driver</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Derate above ( 25^\circ \text{C} )</td>
<td></td>
<td>1.33</td>
<td>mW/°C</td>
</tr>
<tr>
<td>OUTPUT DRIVER MAXIMUM RATINGS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Offset Output Terminal Voltage</td>
<td>( V_{ORM} )</td>
<td>400</td>
<td>Volts</td>
</tr>
<tr>
<td>On-State RMS Current</td>
<td>( I_{F(RMS)} )</td>
<td>100</td>
<td>mA</td>
</tr>
<tr>
<td>Full-Cycle, 50 to 60 Hz</td>
<td>( I_{T(A)} )</td>
<td>50</td>
<td>mA</td>
</tr>
<tr>
<td>Peak Negligible Surge Current (10 μs)</td>
<td>( I_{TSM} )</td>
<td>1.2</td>
<td>A</td>
</tr>
<tr>
<td>Derate above ( 25^\circ \text{C} )</td>
<td>( P_D )</td>
<td>300</td>
<td>mW</td>
</tr>
<tr>
<td>TOTAL DEVICE MAXIMUM RATINGS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Isolation Surge Voltage (1)</td>
<td>( V_{ISO} )</td>
<td>7500</td>
<td>Vac</td>
</tr>
<tr>
<td>Total Power Dissipation @ ( T_A = 25^\circ \text{C} )</td>
<td>( P_D )</td>
<td>320</td>
<td>4.4 mW/°C</td>
</tr>
<tr>
<td>Derate above ( 25^\circ \text{C} )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Junction Temperature Range</td>
<td>( T_J )</td>
<td>–40 to +100</td>
<td>°C</td>
</tr>
<tr>
<td>Ambient Operating Temperature Range</td>
<td>( T_A )</td>
<td>–40 to +70</td>
<td>°C</td>
</tr>
<tr>
<td>Storage Temperature Range</td>
<td>( T_{STG} )</td>
<td>–40 to +150</td>
<td>°C</td>
</tr>
<tr>
<td>Soldering Temperature (10 s)</td>
<td></td>
<td>260</td>
<td>°C</td>
</tr>
</tbody>
</table>

(1) Isolation surge voltage, \( V_{ISO} \), is an internal device dielectric breakdown rating.
FIGURE 3 - HOT-LINE SWITCHING APPLICATION CIRCUIT

Typical circuit for use when hot line switching is required. In this circuit the "hot" side of the line is switched and the load connected to the cold or neutral side. The load may be connected to either the neutral or hot line.

$R_{in}$ is calculated so that $I_{F}$ is equal to the rated $I_{F}$ of the part, 15 mA for the MOC3041 or 30 mA for the MOC3040. The 38 ohm resistor and 0.01 $\mu$F capacitor are for snubbing the triac and may or may not be necessary depending upon the particular triac and load used.

FIGURE 4 - INVERSE-PARALLEL SCR DRIVER CIRCUIT

Suggested method of firing two, back-to-back SCR's, with a Motorola triac driver. Diodes can be 1N4001; resistors, R1 and R2, are optional 330 ohms.

Motorola reserves the right to make changes to any products herein to improve reliability, function or design. Motorola does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others.
### ELECTRICAL CHARACTERISTICS (\(T_A = 25^\circ\text{C}\) unless otherwise noted)

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Symbol</th>
<th>Min</th>
<th>Typ</th>
<th>Max</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>LED CHARACTERISTICS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reverse Leakage Current ((V_P = 6.0\text{ V}))</td>
<td>(I_P)</td>
<td>–</td>
<td>0.05</td>
<td>100</td>
<td>(\mu\text{A})</td>
</tr>
<tr>
<td>Forward Voltage ((I_P = 30\text{ mA}))</td>
<td>(V_P)</td>
<td>–</td>
<td>1.3</td>
<td>1.5</td>
<td>Volts</td>
</tr>
<tr>
<td>DETECTOR CHARACTERISTICS ((I_P = 0\text{ unless otherwise noted}))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peak Blocking Current, Either Direction (Rated (V_{DRM}), Note 3)</td>
<td>(I_{DRM1})</td>
<td>–</td>
<td>2.0</td>
<td>100</td>
<td>(\mu\text{A})</td>
</tr>
<tr>
<td>Peak On-State Voltage, Either Direction ((I_{TS} = 100\text{ mA Peak}))</td>
<td>(V_{TM})</td>
<td>–</td>
<td>1.3</td>
<td>3.0</td>
<td>Volts</td>
</tr>
<tr>
<td>Critical Rate of Rise of Off-State Voltage</td>
<td>(\text{d}v/\text{d}t)</td>
<td>–</td>
<td>100</td>
<td>–</td>
<td>(\text{V/\mu}s)</td>
</tr>
<tr>
<td>COUPLING CHARACTERISTICS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LED Trigger Current, Current Required to Latch Output (Main Terminal Voltage = 3.0\text{ V}; Note 2)</td>
<td>(I_{FT})</td>
<td>–</td>
<td>30</td>
<td>–</td>
<td>mA</td>
</tr>
<tr>
<td>Holding Current, Either Direction</td>
<td>(I_H)</td>
<td>200</td>
<td>–</td>
<td>–</td>
<td>(\mu\text{A})</td>
</tr>
<tr>
<td>ZERO CROSSING CHARACTERISTICS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inhibit Voltage</td>
<td>(I_P = \text{Rated }I_{FT}; MT1-MT2 \text{ Voltage above which device will not trigger})</td>
<td>(V_{IH})</td>
<td>–</td>
<td>15</td>
<td>40</td>
</tr>
<tr>
<td>Leakage in Inhibited State</td>
<td>(I_P = \text{Rated }I_{FT}; \text{Rated }V_{DRM}; \text{Off State})</td>
<td>(I_{DRM2})</td>
<td>–</td>
<td>100</td>
<td>300</td>
</tr>
</tbody>
</table>

Note:
1. Test voltage must be applied within clamping range.
2. All devices are guaranteed to trigger at an \(I_P\) value less than or equal to max \(I_{FT}\). Therefore, recommended operating \(I_P\) lies between max \(I_{FT}\) (100 mA for MOC3040, 15 mA for MOC3041) and absolute max \(I_P\) (60 mA).

### TYPICAL ELECTRICAL CHARACTERISTICS

**FIGURE 1** - ON STATE CHARACTERISTICS

**FIGURE 2** - TRIGGER CURRENT versus TEMPERATURE
<table>
<thead>
<tr>
<th>Temperature (°C)</th>
<th>Type K Thermocouple Voltage (mV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>4.05</td>
</tr>
<tr>
<td>50</td>
<td>10.53</td>
</tr>
<tr>
<td>100</td>
<td>17.01</td>
</tr>
<tr>
<td>200</td>
<td>23.49</td>
</tr>
<tr>
<td>300</td>
<td>30.00</td>
</tr>
<tr>
<td>400</td>
<td>36.57</td>
</tr>
<tr>
<td>500</td>
<td>43.15</td>
</tr>
<tr>
<td>600</td>
<td>49.73</td>
</tr>
<tr>
<td>700</td>
<td>56.31</td>
</tr>
<tr>
<td>800</td>
<td>62.89</td>
</tr>
<tr>
<td>900</td>
<td>70.47</td>
</tr>
<tr>
<td>1000</td>
<td>78.05</td>
</tr>
<tr>
<td>1100</td>
<td>85.64</td>
</tr>
<tr>
<td>1200</td>
<td>93.22</td>
</tr>
<tr>
<td>1300</td>
<td>100.80</td>
</tr>
<tr>
<td>1400</td>
<td>108.38</td>
</tr>
<tr>
<td>1500</td>
<td>115.96</td>
</tr>
<tr>
<td>1600</td>
<td>123.54</td>
</tr>
<tr>
<td>1700</td>
<td>131.12</td>
</tr>
<tr>
<td>1800</td>
<td>138.70</td>
</tr>
<tr>
<td>1900</td>
<td>146.28</td>
</tr>
<tr>
<td>2000</td>
<td>153.86</td>
</tr>
</tbody>
</table>
## Revised Thermocouple Reference Tables

<table>
<thead>
<tr>
<th>Type</th>
<th>Reference Tables</th>
<th>N.I.S.T. Monograph 175 Revised to ITS-90</th>
</tr>
</thead>
</table>

### Thermocouple Grade

- **Nickel-Chromium vs. Nickel-Aluminum**

### Maximum Temperature Range

- **Thermocouple Grade**
  - 32°F to 2200°F
  - 0°C to 1200°C
- **Extension Grade**
  - 32°F to 390°F
  - 0°C to 200°C

### Limits of Error

- **Standard**: ±2.7°C or 0.975% Above 0°C
- **Special**: ±1.1°C or 0.69%

### Comments, Bare Wire Environment:

- Clean, Oxidizing and Inert; Limited Use in Vacuum or Reducing; Wide Temperature Range; Most Popular Calibration

### Temperature in Degrees °F

**Reference Junction at 32°F**

### Thermocouple Voltage in Millivolts

<table>
<thead>
<tr>
<th>°F</th>
<th>0</th>
<th>2</th>
<th>4</th>
<th>6</th>
<th>8</th>
<th>10</th>
<th>12</th>
<th>14</th>
<th>16</th>
<th>18</th>
<th>20</th>
<th>22</th>
<th>24</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
</tr>
</tbody>
</table>

---

**Z-219**

---

87