41

A MULTI-SITE PERFUSION MONITORING SUB-SYSTEM

by

Myev A. Bodenhofer

Submitted to the

DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

December, 1997
[February 1998]

Signature of Author____
Department of Electrical Engineering and Computer Science, December 1997

Certified by____
H. Frederick Bowman, Thesis Supervisor

Accepted by____
Arthur C. Smith, Chair, Department Committee on Graduate Students

A Multi-Site Perfusion Monitoring Sub-System

by

Myev A. Bodenhofer

Submitted to the Department of Electrical Engineering and Computer Science
on December 23, 1997 in partial fulfillment of the
requirements for the degree of

Master of Science

in Electrical Engineering and Computer Science

ABSTRACT

Quantitative information about thermal conductivity and perfusion are critical for planning of hyperthermia treatments. The heterogeneity of perfusion in both time and space demands measurements at several sites for effective planning and thermal modeling. Previous technology allowed measurement at one site at a time at a frequency of 1 Hz. The new microprocessor controlled single-channel sub-system will serve as one component of a multi-channel system providing measurements at up to eight sites simultaneously. The sub-system microprocessor software has been written to accommodate a complete system with eight channels. To allow clinical use, safety features have been included to guard against voltage breakdown or excessive leakage current. The sub-system provides for measurement of either temperature or effective thermal conductivity at 10 Hz with a thermistor-based probe. Temperature is determined from thermistor resistance which can be measured to 0.05 Ω (approximately 0.001 °C), compared to 0.145 Ω (~0.003 °C) with the previous technology. Thermal properties, such as thermal conductivity and perfusion, are measured by self-heating the thermistor to a specified temperature and monitoring the power required to maintain this temperature increment. Experimental studies have shown that the sub-system can measure the power to 0.006%. The size of the temperature increment can be measured to approximately 0.02%, yielding a perfusion calculation with a resolution better than 0.7 %.

Thesis Advisor: Dr. H. Frederick Bowman

Title: Director, MIT Hyperthermia Program
Senior Academic Administrator for the Harvard-MIT Division of Health Science and Technology

Acknowledgments

I wish to express my gratitude to many people who assisted me during the course of this thesis. I could not have completed it on my own.

I'd like to thank Fred Bowman for suggesting this project and for many hours spent listening, reading, and advising. Although I know it wasn't always fun for him, I did appreciate his assistance. I also thank him for the funding.

For electrical engineering help, I thank Charlie Welch, T.A. "Balu" Balasubramaniam, Ken Szajda, and Steve Burns. Charlie supported this thesis with design ideas, trouble-shooting advice, and knowledge of the subtleties of the perfusion measurement, not to mention lots of assembly code. Balu lent his expertise to this project as an independent reviewer of the design and greatly assisted in the formal schematic, layout, and manufacturing stages of the project. Ken, who had been through much of this before, provided many helpful hints to help me avoid the mistakes he had made. Dr. Burns taught a great class on medical instruments and provided many answers when the behavior of the circuit seemed to make no sense.

For thermal property measurement expertise, I thank Greg Martin and Charlie Welch. Both helped me make sense of all my data and lent some direction to the course of my testing and evaluation. I also would like to thank Greg for his input on the background and history of perfusion measurement.

For reading drafts and offering constructive criticism, I thank Greg, Charlie, and my husband, Erik. I thank Erik, in particular, for plowing through pages of technical information that he would never have any need to know.

Finally, for much needed moral support (and for not pointing out that my complaints were getting rather repetitive), I thank Dan Sidney, Bryce Greenhalgh, Charlie, Greg, and Erik and the rest of my family. You have all been very patient with me during this ordeal. I owe you.

Table of Contents

List of Figures

List of Tables

1. Introduction

The goal of this thesis work has been to design, fabricate, and test a printed circuit card that will serve as a single channel sub-system in a multi-channel perfusion monitoring device. This instrument will have the capacity to quantify both perfusion and thermal properties such as thermal conductivity and thermal diffusivity in living tissue at up to eight sites simultaneously. Although this type of perfusion measurement is not new, this instrument will represent a considerable improvement in the measurement technology and will allow for expanded uses of this technology.

1.1 What is Perfusion and How is it Measured?

Perfusion, measured in ml/100g-min, is the volumetric flow rate of blood through a specified mass of biological tissue. Perfusion varies greatly over space, time, and tissue type. Perfusion can be modulated by changes in oxygen requirements, fluctuations in local temperature, and numerous other factors. Knowledge of perfusion is a critical component in assessment of tissue health and its response to treatments and interventions. The proposed instrument will allow simultaneous spatial and temporal data to be collected so that the processes that govern perfusion and the effects of high or low perfusion can be better understood.

Although other measurement techniques exist, this thesis deals with thermodilution measurements of perfusion. This technique has been developed over a twenty year period and is well tested. (Bowman, 1984) Effective thermal conductivity is measured using a thermistor. The thermally sensitive resistor is self-heated to a known temperature above the baseline temperature of the tissue. The amount of power required to maintain the temperature elevation is an indicator of the effective thermal conductivity of the tissue in the interrogated local volume surrounding the active thermistor sensor. This effective thermal conductivity is a function of both the intrinsic thermal conductivity of the tissue and of the perfusion level for that particular volume. Separation of these two components is possible with special algorithms.

1.1.1 Measurement Details

For thermal conductivity measurements, negative temperature coefficient thermistors are employed. These thermistors have resistances which decrease with increasing temperature, as seen in the typical resistance vs. temperature curve below.

The Steinhart-Hart equation, eq. 1.1, is used to relate resistance to temperature.

$$\frac{1}{T} = a_0 + a_1 \ln(R_T) + a_3 (\ln(R_T))^3 \tag{1.1}$$

Thermistors used for this application generally have a nominal resistance of 1 kΩ at body temperature (37 °C) and a diameter of approximately 0.030".

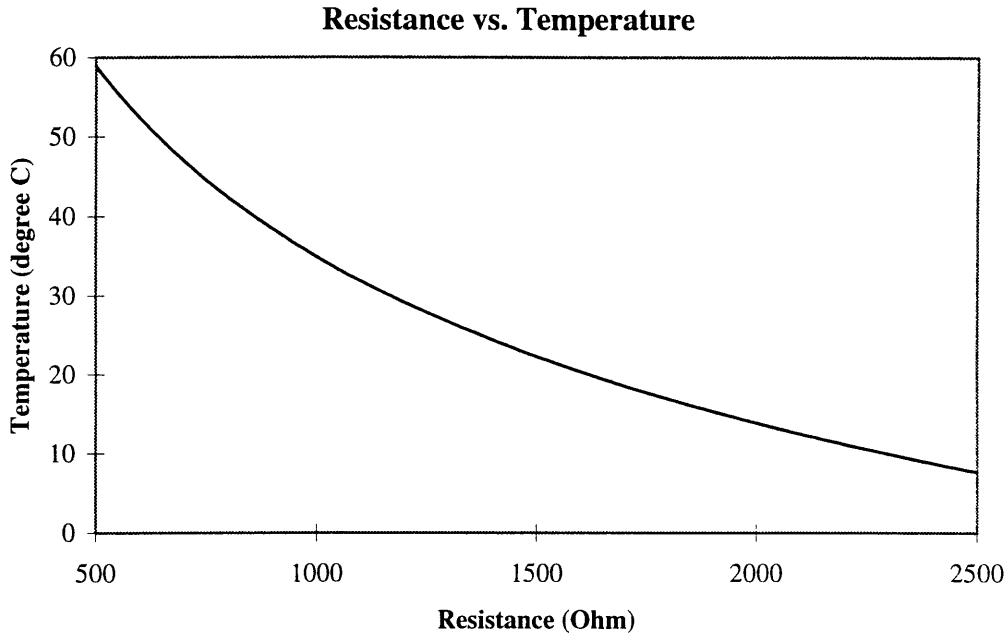


Figure 1-1: Typical Resistance vs. Temperature Curve

Thermal conductivity is measured by determining the power required for a given temperature step. Quantitatively, the relationship between the power dissipated in the thermistor and the intrinsic thermal conductivity and perfusion of the tissue can be understood with the bioheat transfer equation. (Bowman, 1984)

$$\rho c \frac{\partial T}{\partial t} = k\nabla^2 T - \rho_b c_b \omega_b (T - T_a) + q \qquad 1.2$$

Assuming a spherically symmetric thermistor of conductivity k_b in thermal contact with tissue with an intrinsic thermal conductivity of k_m, thermal diffusivity of α_m and a perfusion of ω, this differential equation can be solved. The resulting relationship between power and time is as follows. (Valvano, 1981)

$$P = \frac{4\pi a \Delta T}{\frac{1}{5k_b} + \frac{1}{k_m(1+\lambda a)}} \left[1 + \frac{\frac{a}{\sqrt{\pi \alpha_m}}}{1 + \lambda a + \frac{k_m(1-\lambda^2 a^2)}{5k_b}} f(t) \right] \qquad 1.3$$

where

$$\lambda = \sqrt{\frac{\omega \rho_{bl} \rho_t c_{bl}}{k_m}} \qquad 1.4$$

and

$$f(t) = \frac{e^{-\lambda^2 \alpha_m t}}{\sqrt{t}} - \sqrt{\lambda^2 \alpha_m \pi}\, erfc \sqrt{\lambda^2 \alpha_m t} \qquad 1.5$$

with a equal to the effective diameter of the spherical thermistor. For a constant temperature step above the baseline temperature, the waveform for the power vs. time looks like the following.

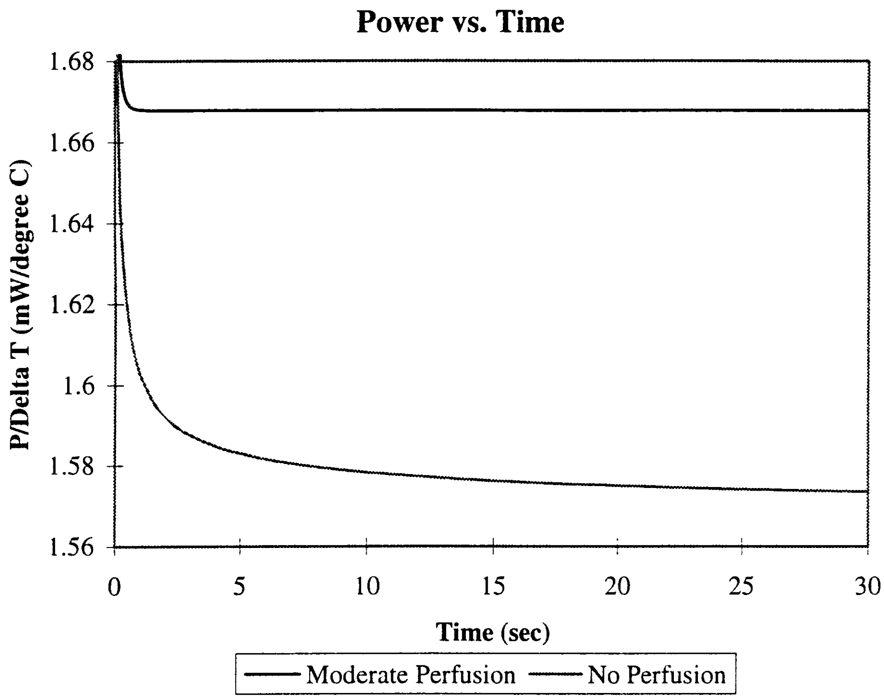


Figure 1-2: Typical Power/ Delta T vs. Time Curve

Special algorithms have been developed to extract the intrinsic thermal conductivity and perfusion from waveforms of this type. As the acquisition of the raw data is the subject of this thesis, the discussion of these algorithms is left to other sources. (Bowman, 1989)

1.2 Clinical Significance

This work has been sponsored by a Project Program Grant from the NIH for the Hyperthermia Program at the Dana Farber Cancer Institute and at MIT. The perfusion and thermal property measurements to be made by the device are especially useful to those evaluating the effects of localized heating in the treatment of cancer. Before a treatment, knowledge of the effective thermal conductivity of the tissue is obviously important when determining whether tissue at a particular site can be heated to a specified temperature.

In addition, these data are critical for the calculations which are performed during and after a treatment. Although temperature may be directly measured at only a few discreet sites during a hyperthermia treatment, the true goal of the treatment is to heat the entire tumor volume without unduly heating the surrounding healthy tissue. With the knowledge of intrinsic thermal conductivity and perfusion at distinct sites and the temperature data from these and other discrete temperature sensors, it is possible to perform thermal reconstructions to better understand the temperature profile over large volumes of the tissue. Increased density of perfusion measurements will allow for improved thermal reconstructions and treatment planning.

Also, we wish to better understand how hyperthermia affects the tissue. Prior experiments have indicated that perfusion levels can be significantly altered by localized heating. By measuring perfusion simultaneously at up to eight locations, we will be able to monitor these changes more effectively and gain insight into relationships of temperature, perfusion, time, and spatial extent.

Although this instrument will be specifically designed for hyperthermia applications, its usefulness is not limited to this area of research. The medical implications of knowledge of blood flow levels in tissue could encompass all medical specialties. Obviously, blood flow and the oxygen and nutrient supply that it entails are critical to life. However, the measurement of this blood flow is not a simple task. Some technologies, such as the laser Doppler technique, allow for a relative measurement of blood flow to be made and will track changes over time. Other techniques yield an absolute measure of the amount of blood flow, but these measurements require a no-flow calibration *in vivo*. While such measurements have been made in research laboratories, occluding blood flow is not practical or advisable in living patients. Hydrogen clearance provides an absolute measure of perfusion without a no-flow calibration, but the technique requires the patient to breathe a known concentration over a period of time and then provides a measure of average perfusion over a time window of a few minutes.

The thermodiffusion technology, particularly with the use of probes at multiple sites, will allow for more in depth study of blood flow and the factors that effect it. How perfusion levels are regulated over time and during periods of stress to the tissue are still largely unknown. Since the typical sensor interrogates a volume with a radius of approximately only 2 mm, a high degree of spatial resolution may be achieved. Simple experiments performed at the Biomedical Engineering Center have shown that perfusion can vary greatly even over such short distances. For the data shown below, two probes were placed superficially on the hand of a human subject, in the fold of skin between the thumb and the rest of the hand. Probe 1, nearer to the palm side of the hand, was placed approximately 1 cm from Probe 2, which was placed nearer to the back side. From 400 to 430 seconds, blood flow to the hand was partially occluded by tourniquet around the upper arm.

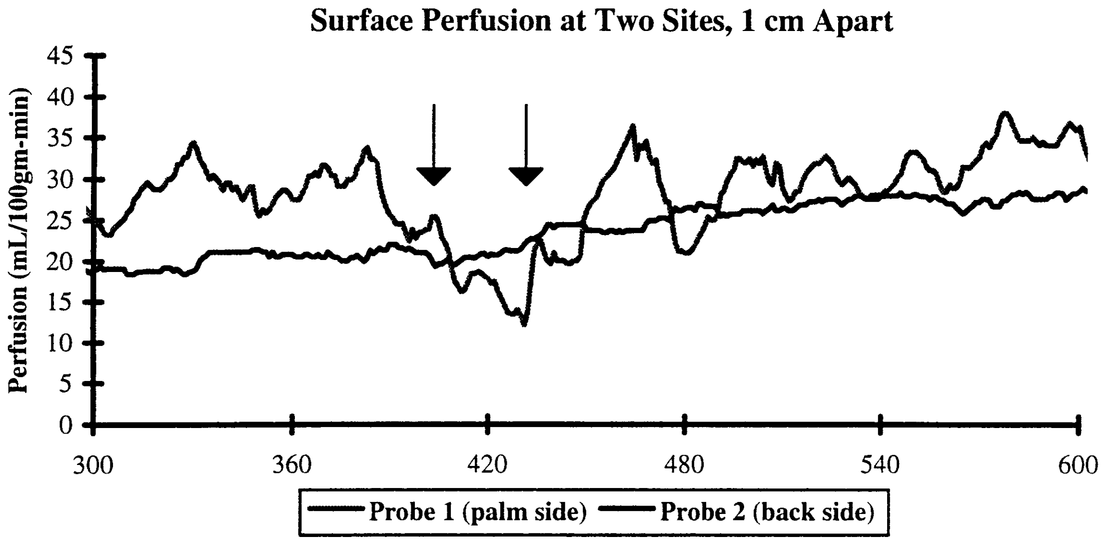


Figure 1-3: Perfusion at Two Sites on Human Hand

Although both probes measured an intrinsic conductivity of 4.0 mW/cm-°C, the perfusion at the two locations was quite different. The perfusion measured by Probe 1 shows much more vaso-motor activity. Perfusion measured by Probe 2 shows less vaso-motor activity and a long-term rising trend not seen at the other location. Clearly, there would be no way to predict the behavior at one site by measuring perfusion at the other.

While the study of mechanisms for perfusion control may continue for years, this technology is currently in use for other medical applications. In particular, knowledge of blood flow is very important for organ transplantation. Dr. Ernst Klar of Heidelberg, Germany, has been using the thermodiffusion technique in conjunction with liver transplants to assess graft viability with significant success.

As knowledge of liver microcirculation may be important for early detection of post-operative complications, Klar has performed studies to validate the measurement in the human liver and to compare perfusion levels to success of liver transplants. Porcine studies showed that the thermodiffusion technology compares favorably with hydrogen clearance perfusion measurements.(Klar, 1995) The thermodiffusion probes were placed in the liver for periods up to five days and provided reliable readings throughout that time. Interventions to reduce perfusion, including occlusion of the hepatic artery and portal vein, supplying blood to the liver, resulted in the expected drop in perfusion, as measured by the probes.

The first study using the thermodiffusion technique in human livers is quite promising. (Klar 1996, 1997) Patients with successful transplants all showed similar levels of liver microcirculation 12 hours after surgery. Two patients with complications had significantly lower levels of microcirculation. Perhaps most significantly, for one patient, a reduction of hepatic perfusion occurred two days prior to liver rejection, seven days after surgery. While more studies will be required to determine if such measurements might provide a reliable indicator of complications, Klar's research has shown that such measurements are safe and potentially an important tool for human liver transplants.

Dr. Dennis Orgill has also explored the use of this perfusion measurement technique for reconstructive surgery, using the technology for rabbit experiments and for human patients. (Newman) Unlike skin grafts, some reconstructive surgery involved transplantation of a tissue flap with a vascular network and a pedicle, which includes an artery and a vein which feed the vasculature in the flap. During the entire transplantation procedure, blood supply to the flap is crucial for success. Although qualitative measures of perfusion to the flap can be made, the thermodiffusion technique may provide critical quantitative information for reconstructive surgery.

As the measurement technology is improved and the uses in liver transplantation and skin grafts are better understood, we believe that many more applications will be discovered.

1.3 History of the Measurement

Thermal techniques have been used to measure flow for many years. (Bowman, 1984) With the coupled solution of the bioheat equation for the special case of a thermistor in contact with tissue, Bowman and Balasubramiam began to measure thermal conductivity and thermal diffusivity. (Balasubramaniam 1974, 1977) In 1977, this work led directly to the determination of perfusion using the same thermodiffusion technique. (Bowman, 1977) Although state of the art electronics at that time could not offer the precision available today, these initial studies verified the technique and justified further development.

In the late 1970's, the thermal diffusion project was selected as one of four potentially valuable real-time medical instruments supported by an NIH Institute of Bio Medical Sciences Program Project Grant. This grant was the means for launching the HST Biomedical Engineering Center for Clinical Instrumentaion at MIT. The BME Center was devoted to the development of the first microprocessor based clinical instruments. The solution of the coupled time-dependent tissue-probe diffusion equation, and the design and fabrication of a four channel thermal diffusion probe with a magnetic tape mass storage device was the basis of Dr. Valvano's Ph. D. research. (Valvano, 1981) The measurement technique was reported in 1984 (Valvano) and tested against a rat liver system, designed as an independent flow system, used to validate the accuracy of the perfusion measurement. (Valvano, 1984)

In 1986, the Thermal Diffusion Probe (TDP) technology was licensed by MIT to Thermal Technologies, Inc. (TTI). The original work in the feedback control loop to heat the thermistors without the use of a ladder resistance network was started in the late 1970's by T. A. Balasubramaniam and J. Rollins. Dr. Balasubramaniam fabricated the first four versions of the TDP and also spear headed the use of newer, sigma-delta, analog to digital converters while employed at TTI.

The first readily portable, high precision, embedded microprocessor controlled thermodiffusion system was the TDP100. This system included a single-thermistor probe, measurement electronics to make temperature measurements with the thermistor, a control loop to allow the thermistor to be heated to a known temperature, and a personal computer. Each probe was individually calibrated for both temperature and conductivity.

Temperature calibration involves measuring the resistance of the thermistor at several known temperatures and fitting the results to the Steinhart-Hart thermistor equation. Thermal conductivity calibration is performed by measuring the required power to maintain a particular thermistor volume averaged temperature step in two media of known thermal conductivity. Using these data and the equations above, the k_b and a for each probe are determined.

In 1989, Szajda designed and fabricated translation electronics for measuring perfusion at multiple sites as part of the profilometer project. (Szajda) This instrument could multiplex eight different probes, one at a time. He also implemented a control loop that could be used for both passive temperature measurement and for self-heating of the thermistor in either a constant temperature or constant power mode. In addition, this instrument used a faster analog to digital converter for the sampled measurements and included a separate analog to digital converter to allow for integration of the power vs. time waveform.

The TDP200, based largely on the TDP100, included the use of a second thermistor as a passive temperature sensor. A probe designed for use with this instrument is shown in Figure 4.

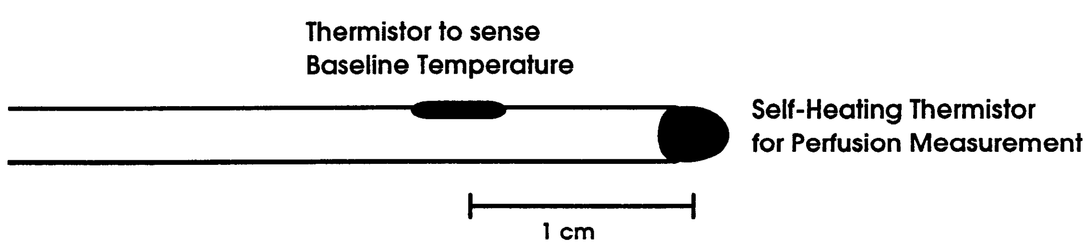


Figure 1-4: Single-Site Perfusion Probe

By tracking changes in the baseline temperature near the measurement site, but outside the thermal field of the heated thermistor, the measurement can be carried out over longer periods of time in the presence of a changing baseline temperature. Precise measurements can be made even when the tissue is slowly heating or cooling.

After the TDP200 was completed, development focused on the algorithms to calculate perfusion from the measured effective thermal conductivity. An important improvement was the ability to calculate perfusion in real time. Previous algorithms had required that a measurement be completed before the perfusion could be determined. While such algorithms are still the most accurate, by making the simplifying assumption that perfusion has no effect on the initial transient (up to 30 seconds) of the power vs. time waveform, it is possible to estimate intrinsic thermal conductivity from the initial data and use that estimate to calculate perfusion while the rest of the measurement is going on. (Bowman, 1989) Although this technique always overestimates intrinsic conductivity slightly, it does yield quite good values, especially when perfusion is not very high. This ability to track perfusion in real time is invaluable to clinicians.

The TDP200 and the real-time perfusion algorithms have now been in use for several years. Tsai, investigating the non-invasive measurement of thermal properities, studied ways to improve these technologies for increased accuracy in measures of thermal conductivity and found several areas for improvement. (Tsai, 1995) By increasing the sampling rate beyond the 1 Hz rate of the TDP200, the study showed that more precise measures of intrinsic thermal conductivity may be made. Any improvement in the measurement of intrinsic thermal conductivity translates directly into more accurate determinations of perfusion levels from the effective thermal conductivity calculated from the power vs. time waveform.

Tsai's research also demonstrates that increased control over the size of the temperature step can improve the accuracy of measurements. The calibration of the thermal conductivity measurements is somewhat dependent on the size of the temperature step. The TDP200 can self-heat the thermistor to a fixed resistance chosen from a finite set of resistances, in steps of approximately 5 Ω. The set resistance is a function of the resistance to temperature characteristics of the thermistor, the desired temperature step, and the measured baseline temperature of the tissue. For a typical thermistor at body temperature, 5 Ω translates to approximately 125 mdeg. Therefore, depending on the conditions at the time, for a desired temperature step of 6 °C, the actual temperature step would be expected to be within approximately 1% of the target. This error is larger for smaller temperature steps. Tsai showed that the calibration does vary with the size of the temperature step. Increasing the precision of control over the set resistance is one solution to this problem.

The new sub-system, described in this thesis, will build upon the prior technologies and incorporate improvements based on the suggestions of Balasubramaniam, Tsai, and other users. Most importantly, the system, when completed, will have the ability to make measurements at eight sites simultaneously. This density of quantitative perfusion measurements, much needed for heat transfer modeling and hyperthermia thermal planning, has not been previously achieved by any other instrument. The goal of the project is an advanced, user-friendly system that will allow measurements of thermal conductivity and perfusion to be made in many settings.

2. System Design

The multi-site perfusion measurement system includes an instrument, probes, and software. The instrument encompasses all of the measurement electronics, the interface to the host, the connectors for the probes, and the necessary power supplies. Although this thesis focuses on only some of the electronics of the instrument, the system as a whole is discussed below to provide a framework for understanding this project.

The basic design of the new system builds on the single-site TDP200 system which includes a probe, hardware to control the measurement and to convert the data to a digital format, and a host PC running software which provides the user interface and allows the data to be processed and viewed. The probe is composed of two thermistors. A "sense" thermistor is always used to monitor baseline temperature. The "heat" thermistor can operate in two distinct modes: perfusion-sensing and temperature-sensing. When in perfusion mode, the heat thermistor is powered to maintain a fixed temperature determined from the initial baseline temperature and a desired temperature increment above this baseline. For the multi-site system, like the TDP200, each sensor will be composed of a pair of thermistors: "heat" and "sense."

2.1 Operational Specifications

In consultation with the other members of the team, the specific operational goals for the multi-site system have been defined. The system will be capable of simultaneous perfusion measurements at eight independent sites with a total of 16 thermistors (8 heat and 8 sense.) Like the TDP200, both perfusion-sensing and temperature-sensing modes will be available. The mode of operation for each of the eight sites will be controlled independently. The new system will provide for backward compatibility of probes by using the same probe specifications (same resistance and temperature ranges). To be used in the clinic, the multi-site instrument must also meet safety specifications for breakdown voltage and leakage current.

In addition to the increased number of measurements, the new system will include several other improvements over the single-site system. The data will be sampled with delta-sigma A/D converters to allow for better precision and noise immunity as well as a faster sampling rate (10 Hz. vs. 1 Hz.). Andy Tsai's research indicates that the increased sampling rate is especially important to the intrinsic conductivity calculations made with data from the first few seconds of a measurement. The new instrument will also include more steps in the resistance ladder that controls the self-heating, yielding more precision for the desired temperature above baseline. On the suggestion of current users of the TDP200, calibration of the A/D converters will be included, and a thermistor simulator will be added to the circuit board to allow for easier testing, and stability problems with the TDP200 control loop will be addressed.

Probes will be developed independently with up to eight thermistor pairs spaced along a needle to allow perfusion measurements at evenly spaced intervals (Fig. 2-1). If the user wishes to locate the sensors further apart, the option of eight single-site probes is still available.

Figure 2-1: Multi-Site Needle Probe

The software for the host computer will be upgraded to accommodate data from all eight sensors and will be made more user-friendly. Existing real-time perfusion calculation algorithms will be incorporated along with real-time statistics that indicate thermal stability of the tissue and average perfusion over a given time window.

2.2 System Architecture

The system architecture, designed with other engineers working on aspects of the project, is based on the need for safety and modularity in the design. For medical safety, the probes must be electrically isolated from the host PC , from the general electrical system of the hospital or laboratory, and from the patient. The insulated thermistors do not make electrical contact with the patient, so the primary concern is the isolation from other electrical apparatus. As the probes cannot be isolated from the electronics which drive them, the instrument must be isolated from the host PC. Therefore, opto-isolated serial communication is the only link between the host computer and the instrument.

For optimum safety in the unlikely event of a fault in the insulation of more than one of the thermistors, each sensor pair must also be electrically isolated from every other sensor pair. To meet this requirement, the instrument has been designed such that each perfusion measurement pair is controlled by a dedicated printed circuit board known as the "probe board." Each probe board uses an isolated power supply and communicates with the rest of the instrument through opto-isolated serial communication. All control of the heat and sense thermistors is performed by the probe board, including measurement of resistance and power and conversion of this data to digital format. Figure 2-2, below, shows the basic system architecture.

Each probe board includes an 8051-based microprocessor to control the measurements and communicate with the master microprocessor, a 386 processor. For fast sampling and high precision, sigma-delta analog to digital converters are employed, with one dedicated to each thermistor. The two A/D converters on each board send data to the microprocessor which then sends data to the master microprocessor.

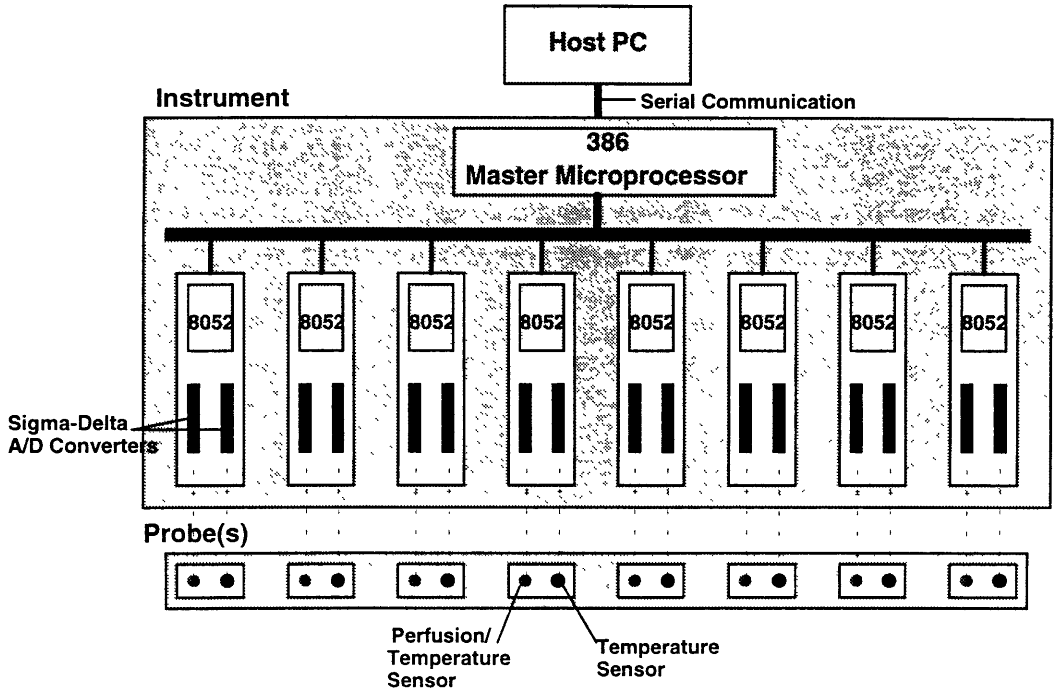


Figure 2-2: Schematic of System Architecture

The communication protocol or process by which this data is sent is an important component of the system design. Somehow, the host PC must receive data from eight probe boards controlling 16 sensors without confusion. Since each measurement is made at a rate of 10 Hz, timing may be a constraint. To simplify the data transmission, each probe board will maintain a buffer of data and will pass that data to the master microprocessor when polled. Polling of each probe board should occur at least once a second.

By keeping each probe board independent, the system is very modular. Any malfunctions on one board will not at all affect any other sensors. Even the communication protocol is unaffected by the number of boards in use at once. With more boards, each probe board might be polled for data less frequently, but the data transmission does not depend on a particular sequence of data reaching the master microprocessor. This protocol makes the operational specification of ability to start and stop each sensor independently easier to implement.

2.2.1 The Probe Board

This thesis describes the design and testing of the probe board sub-system which represents a distinct portion of the work for the multi-site perfusion measurement system. The analog design of the probe board includes circuitry to control the thermistors for passive temperature sensing ("sense") measurements and for active perfusion ("heat") measurements along with switches to control the different modes and necessary calibrations. The digital part of the design involves the interfacing between the microprocessor and the A/D converters, the analog switches, and the master microprocessor. The third part of the design is the firmware for the microprocessor. This microprocessor software is entirely separate from the software written for the host PC. The details of the design of the probe board are described in Chapters 3, 4, and 5 of this thesis.

3. Analog Design

The thermal measurement of perfusion is inherently an analog measurement. As such, the analog circuitry is the most important part of the perfusion measurement board. Two distinct modes of operation must be implemented with the analog electronics: temperature measurement and effective thermal conductivity measurement. Each requires its own circuitry.

3.1 Temperature Measurement

Temperature measurement is the simplest function of the perfusion measurement system. The perfusion probes are composed of negative temperature coefficient thermistors with a known relationship between temperature and resistance. Measurement of this resistance is accomplished by sending a fixed current through the thermistor and measuring the resulting voltage.

To make the most accurate measurement, it would be desirable to generate a large voltage across the thermistor. However, if too much power is dissipated in the thermistor, the thermistor will self-heat. Obviously, this self-heating defeats the purpose of accurately measuring temperature. As self-heating cannot be avoided completely, it is best to attempt to minimize the power and account for any small amount of self-heating in the calibration process. Assuming that the resistance temperature calibration uses the same current as the measurement and is performed in a medium with thermal properties similar to tissue, the self-heating will be accounted for by the calibration.

Keeping these goals in mind, a measurement current of 50 μA was selected. The TDP200 uses a 100 μA current source. Equation 3.1 can be used to calculate the order of magnitude of self-heating. This equation assumes no perfusion and a steady-state condition.

$$\Delta T = \frac{1000\left(1+\frac{5k_b}{k_m}\right)i^2 R}{20\pi a k_b} \qquad 3.1$$

For typical conditions such as k_b=1 mW/cm-°C, k_m=5.5 mW/cm-°C, R=2 KΩ, a=0.05 cm, and i=100 μA, the expected temperature rise is approximately 12 m°C. By reducing the current by a factor of two, the power and self-heating are reduced by a factor of four. The measurement voltage is also reduced, however, an improved analog to digital converter will compensate for the smaller voltage.

The current source in the TDP200 is simply 15 V connected to a large resistor in series with the thermistor. Although the linearity of this source is probably acceptable for this application, particularly with the calibration procedure, the 15 V source from the power supply may not be stable enough to assure the validity of long term measurements.

Like Szajda's instrument, the probe board uses a current source designed for constant current over a range of loads. The current source, shown in Figure 3-1, is composed of an opamp and an npn transistor. The feedback of the opamp sets the collector voltage to 4V. 1 V is dropped across the 20 KΩ resistor, causing a constant 50 μA to flow into the transistor. As long as emitter voltage is below 4 V (the load is less than 80 KΩ), the current output is unchanged. Assuming that the thermistor has a resistance of 1 KΩ at 37 °C and the resistance varies 4%/ °C, measurements could be made for temperatures down to -74 °C with this current source.

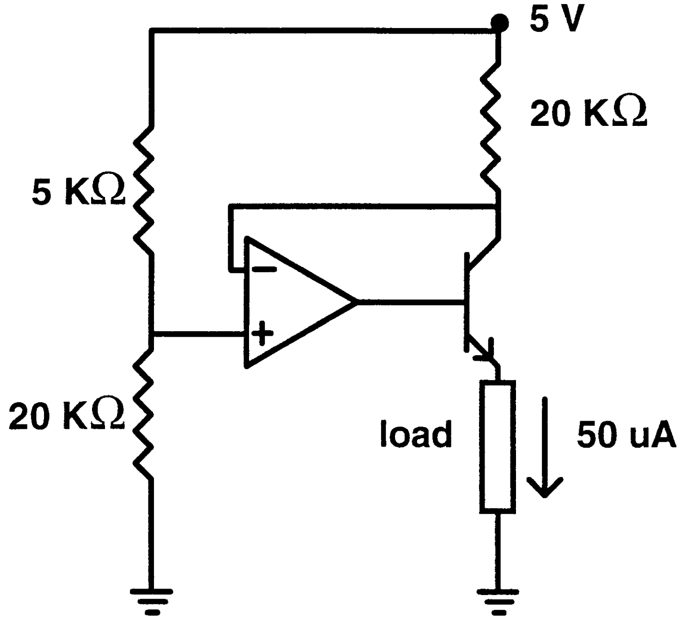


Figure 3-1: Current Source for Sensing Temperature

Although the stability of the current is crucial for precise temperature measurements, the resistors used for this current source do not need to be precise. Resistors precise to 1% will be sufficient because the current source will be calibrated as part of the calibration process at the start of each measurement. Using two precision resistors, 500 Ω and 2.5 KΩ, the relationship between resistance and measured voltage will be calculated. The calibration process may be repeated periodically to account for any drift that may be caused by changes in ambient temperature or variations in the power supply.

This basic temperature measurement circuit is repeated twice on the probe board. One current source can be connected to the "heat" thermistor generally used for perfusion measurement. The second current source supplies current to the "sense" thermistor which is only used to measure temperature.

3.2 Perfusion Measurement

Effective thermal conductivity and perfusion are measured with a very different technique. The thermistor is purposely self-heated to a known temperature above the baseline temperature. This heating is accomplished by using a control loop to drive the thermistor to a fixed resistance and, therefore, temperature. The control loop is designed such that when the resistance decreases (the temperature increases), the current drops, reducing the power to the thermistor. Conversely, an increase in resistance results in more power to the thermistor.

3.2.1 Control Loop

The stability of this control loop is crucial to the perfusion measurement. Thermal conductivity and perfusion calculations are based on the measurement of power divided by the size of the induced temperature step. The temperature of the thermistor must be held at a fixed so that the temperature step can be determined accurately.

Previous perfusion instruments have employed different schemes for control. The TDP200 and Szajda's instrument both use control loops that rely on negative feedback. Each utilizes the basic principle of integrating the difference between the desired value and the actual value. However, the integrator in the TDP200 control loop, depicted in Figure 3-2, sets a voltage. Szajda's integrator drives a current source, as shown in Figure 3-3. The two systems also differ in how the desired resistance is set. Szajda's design relies on a programmable scaling network to amplify the voltage across one fixed resistor. This feedback loop balances the voltage across the thermistor with this amplified voltage out of the scaling network. The TDP200 employs a ladder of fixed resistors and relays to set which resistors are included in the set resistance. The control circuitry balances the voltages across the thermistor and the set of fixed resistors in the same loop.

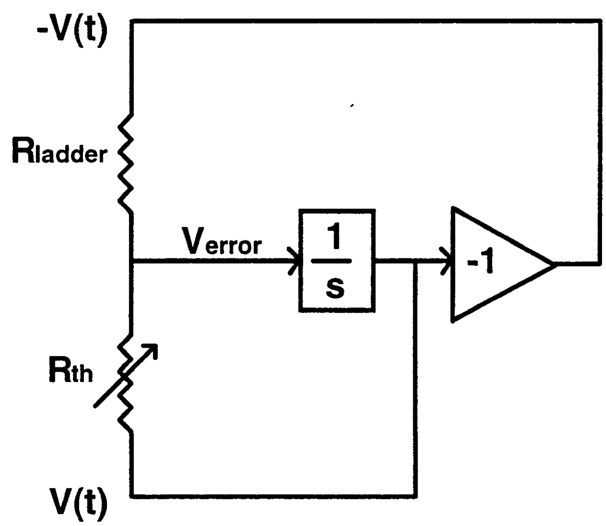


Figure 3-2: Feedback Used by TDP200

Although the TDP200 voltage-based control loop has worked well for many years, a current-based control loop such as Szajda's has several advantages. First, it is simpler, more easily implemented, and uses up less printed circuit board real estate. The scaling network, described below, requires less space than an array of resistors which are included or excluded from the ladder using individual relays. In addition, microprocessor control of the set resistance is simpler with a single amplifier than with multiple relays.

The current-based feedback system also eliminates a problem occasionally found with the TDP200 control loop. The polarity of Szajda's design is predetermined. Current always flows into the current sink. With the TDP200 loop, there is no such restriction and current may flow in either direction. Although theoretically the sign of the voltage across the thermistor should have no effect on the power measurement, the opamps used for the integration and inversion do have a finite offset voltage. If the input voltage is inverted, the output shows a small change that can affect the calibration.

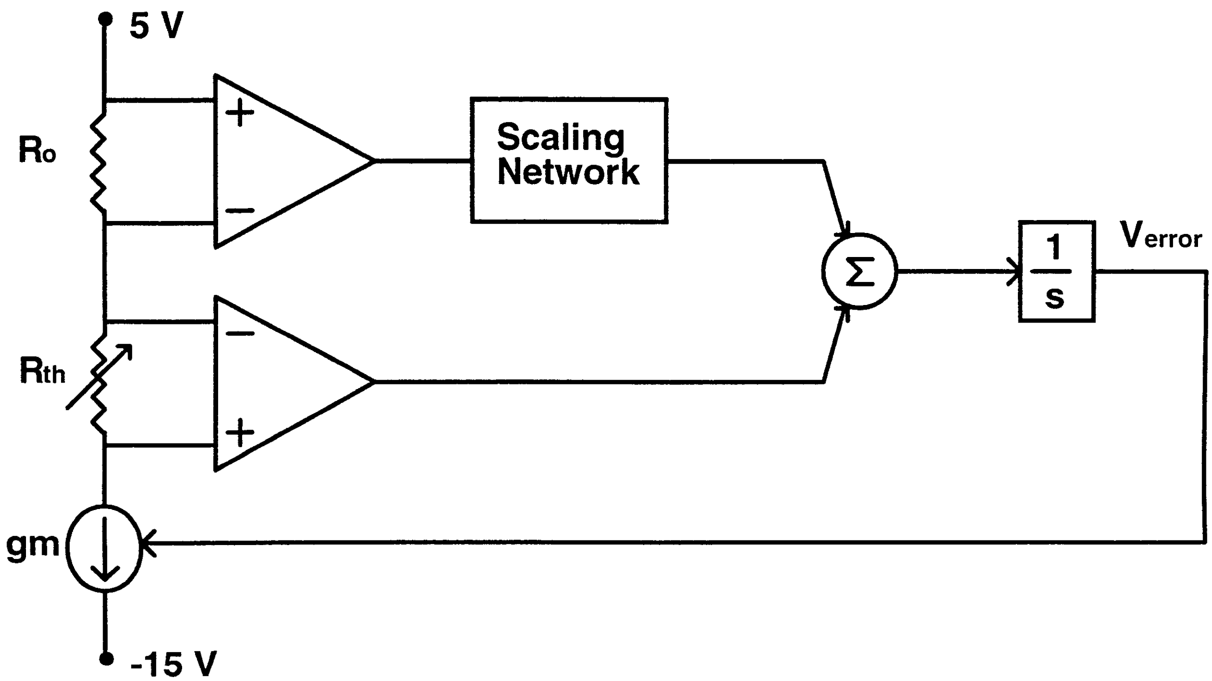


Figure 3-3: Basic Configuration of Control Loop

Finally, the control loop shown in Figure 3-3 may provide an extra degree of safety in the case of probe failure. Although the probe should always be insulated from the body, there is a small possibility that the probe could break and the thermistor could be shorted out. In this case, the apparent resistance of the thermistor would drop radically. Both control schemes would reduce the power to the probe in an effort to increase the resistance. To balance the loop, the current or voltage would be driven to zero. However, with the voltage-based control, a more sizable amount of current could be delivered to the shorted thermistor. In the first instant after breakage, the voltage would remain the same, but the resistance would drop giving rise to a possible spike in current. The current-based feedback loop inherently limits the current to the probe. For all of these reasons, a control loop very similar to that used by Szajda has been selected for the probe board.

3.2.2 Scaling Network

The design for the scaling network, shown in Figure 3-4, is slightly different than what Szajda used. Most significantly, the design has been altered so that it is impossible to select a set resistance of zero. Although the software should be able to avoid this condition, it is safer to set up this restriction in the hardware. A set resistance of 0 Ω would result in saturation of the current source and very high amounts of power to the thermistor until the operator noticed a problem and shut the system down. For safety purposes, the scaling network is designed for a minimum set resistance of 500 Ω, like the TDP200. Again, assuming a thermistor with a resistance of 1 KΩ at 37 °C, a resistance of 500 Ω corresponds to a temperature of approximately 54 °C.

The input to the scaling network is from the differential amplifier measuring the voltage across the fixed resistor in Figure 3-3. This input serves as the voltage reference for a multiplying digital to analog converter (DAC) which works with an inverting amplifier to yield a voltage output. Depending on the setting of the DAC, the output may vary from 0 V to $-V_{ref}$. The 6 KΩ and 5 KΩ resistors serve as a voltage divider, with the output of the DAC and the original input to the scaling network as inputs. The voltage is then amplified by 11 with the non-inverting amplifier. The output voltage can be determined using Equation 3.2. The mismatch of the resistors of the voltage divider make it impossible to generate an output of zero, regardless of the functioning of the DAC and opamp.

$$R_{set} = R_o\left(6 - 5\frac{Code}{4096}\right) \qquad 3.2$$

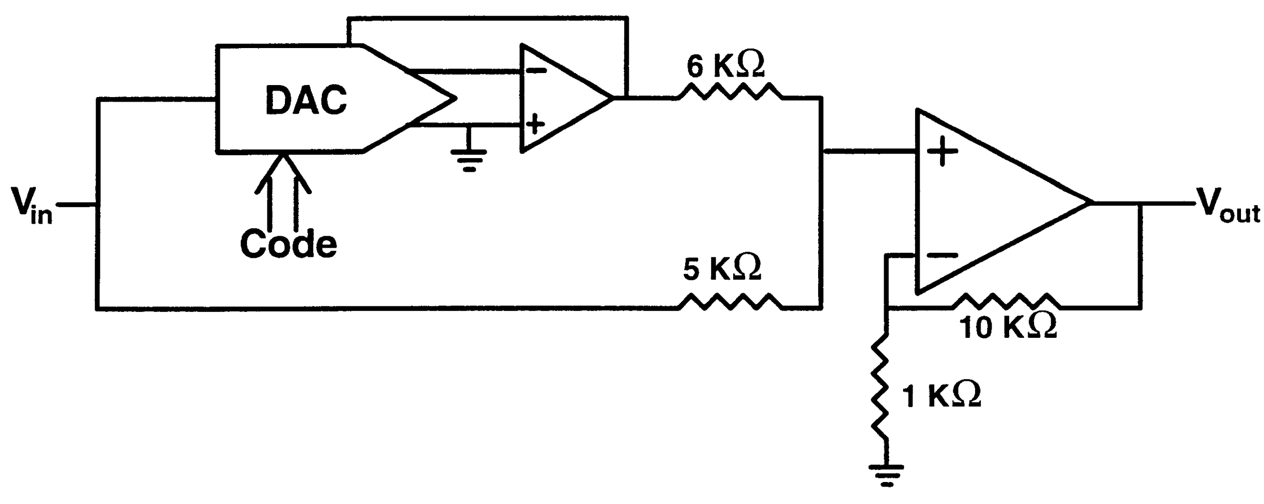


Figure 3-4: Scaling Network for Control Loop

With the Maxim MAX543 12-bit serial input DAC, the "Code" can range from 0 to 4095. For R_o=500 Ω, the maximum set resistance is 3 KΩ. Since most of the thermistors used to build perfusion probes have a nominal resistance of 1.5 KΩ at room temperature, there should be no problem with an upper limit of 3 KΩ for physiological measurements. If future measurements require a different range of set resistance, a change could be accommodated by changing the gain of the opamp. The 12-bit DAC will allow for precision of 0.625 Ω in the R_{set} value. The TDP200 is set up for resistance steps of 5 Ω. The improved precision will allow for better control of the set temperature, implementing one of Tsai's recommendations.

3.2.3 Differential Amplifiers

The characteristics of the differential amplifiers used in the feedback loop determine the stability and accuracy of the control. The Burr Brown INA114 differential amplifiers were selected for the high impedance and high common mode rejection ratio that they offer. The typical impedance of 10 MΩ, compared to approximately 2 KΩ for the thermistor, insures that the current through the fixed resistor, the thermistor, and into the current sink is virtually undisturbed by the voltage measurement.

For optimal accuracy, the output of the differential amplifiers must faithfully represent the voltage drop across the inputs, regardless of the common mode voltage. When used with a gain of 1, the typical common mode rejection ratio is 90 dB. Therefore, a common mode voltage of 10 V would result in a 0.3 mV disturbance in the output. To reduce the effect of common mode voltages, the circuit has been designed with a maximum voltage of 5 V. This voltage is high enough to allow sufficient power to the thermistor during the initial heating transient without allowing excessive common mode voltages when the voltage requirement settles to a typical value of approximately 1 Volt.

3.2.4 Current Source for Control Loop

Szajda's design used a relatively complicated current source, with three bipolar transistors, so that one source could be used for both passive temperature measurement and active conductivity measurement. For the new probe board, the design has been simplified. For current outputs in the mA range, as needed for perfusion measurements, similar functionality can be achieved with a single transistor, as shown in Figure 3-5.

There is no need for a more complicated current source, such as a cascode design, because the current source is inside the feedback control loop. If the output drifts slightly with temperature or with V_{BE}, this drift will be compensated for by the feedback.

The current source has been designed to generate the expected current of approximately 1 mA with a input voltage near the middle of the range of voltages from the integrator. With a voltage input of 0, the current source provides approximately 5 mA of current. For a typical 2 KΩ thermistor, this generates 10 V across the thermistor. This value is comparable to the initial voltage requirement seen in previous TDP200 measurements. For 1 mA output, the input would be approximately -5.5 V. The control loop is designed so that the output of the integrator, the input of the current source, can vary from 15 V to -15 V. The -5.5 V expected during normal operation is a comfortable distance from either of these limits. The current source cuts off at approximately -6.6 V. A SPICE simulation of the relationship between the input voltage and output current for a load of 2 KΩ is shown in Figure 3-6.

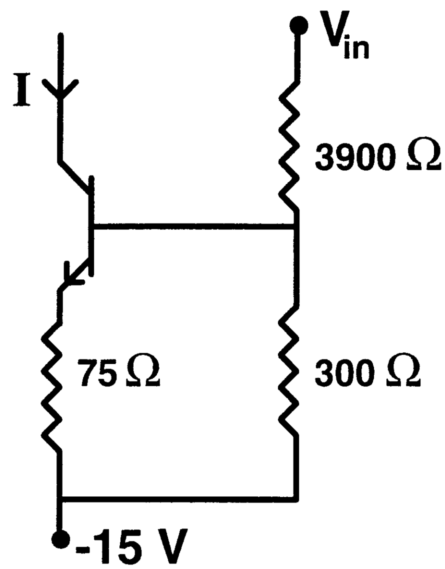


Figure 3-5: Current Source for Control Loop

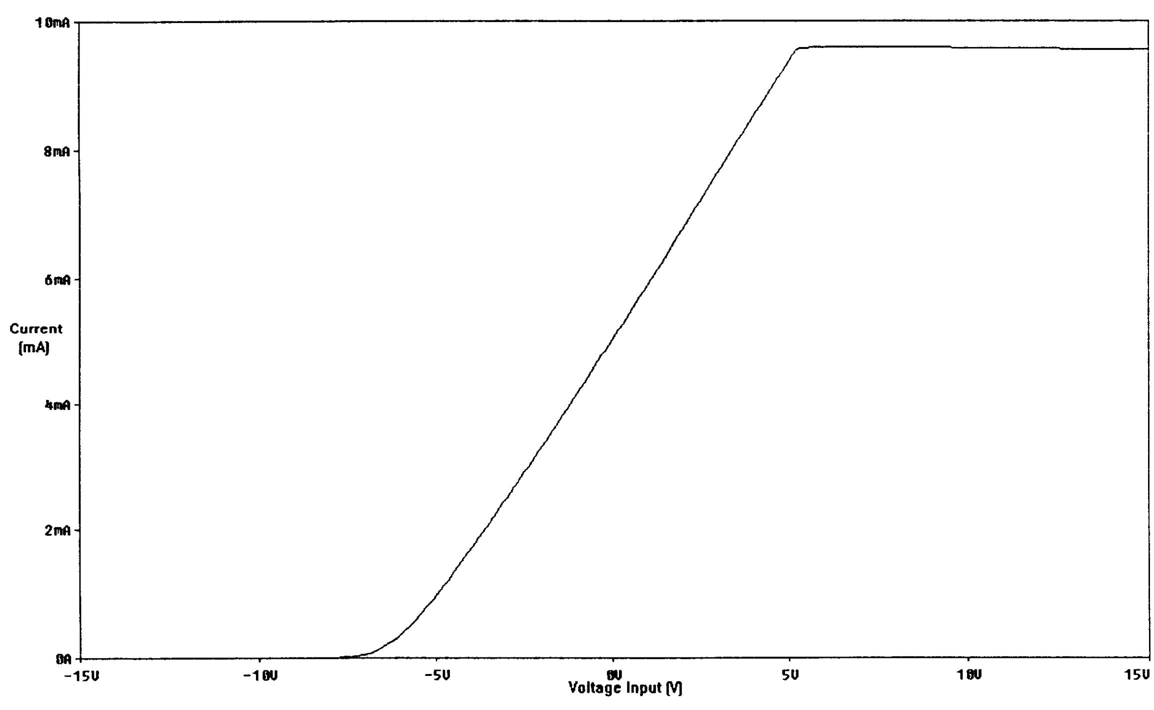


Figure 3-6: Voltage Input to Current Output

The maximum possible current is determined by the resistive load. Assuming that the thermistor were shorted out, there would still be a 500 Ω load. In this case, the maximum input voltage of 15 V would cause approximately 19.2 mA to be sent through the short. Although this value is considerably higher than the 10 μA of continuous leakage current that is allowed for a medical device, the control loop would almost immediately reduce this current to zero in the event of a shorted thermistor.

3.3 Switches for calibration and testing

The analog design must include provisions for testing of the circuit and for the necessary calibration of the probe board. For simplicity, it is important to minimize the number of switches provided while still maintaining the flexibility to allow the circuit to be divided into smaller segments for diagnostic purposes.

Figure 3-7 shows the complete control circuit with the switches labeled with letters. Switches B and C are actually each two switches which are controlled together. With just six switch controls, the measurement, calibration, and testing modes can all be selected.

3.3.1 Modes of operation

The first mode of operation is passive temperature monitoring. For this mode, the thermistor must be connected to the sense current source and to ground. For calibration purposes, two precision resistors (2.5 KΩ and 500 Ω) are connected in series as well. The position of each switch for this mode is described in Table 3-1.

Table 3-1: Switch Positions for Sense Mode

Switch	Position
A	right, connecting fixed resistor to thermistor
B	left, selecting thermistor
C	right, connecting thermistor to ground
D	left, connecting fixed resistor to current source
E	up, disconnecting feedback loop
F	closed, opamp as amplifier not integrator

With the thermistor attached to ground, the common mode voltage is very low and should not disturb the measurement. For a typical 2 KΩ thermistor, the common mode voltage across the thermistor would be only 50 mV. Before temperature of the thermistor is measured, the voltage across the two precision resistors will be measured, allowing for calculation of the resistance to voltage relationship. The voltage drop across the 2.5 KΩ precision resistor will be nominally 125 mV, compared to a 188 mV common mode voltage. With a reasonable common mode rejection ratio of just 60 dB, the common mode voltage would contribute less than 0.2 mV to the output.

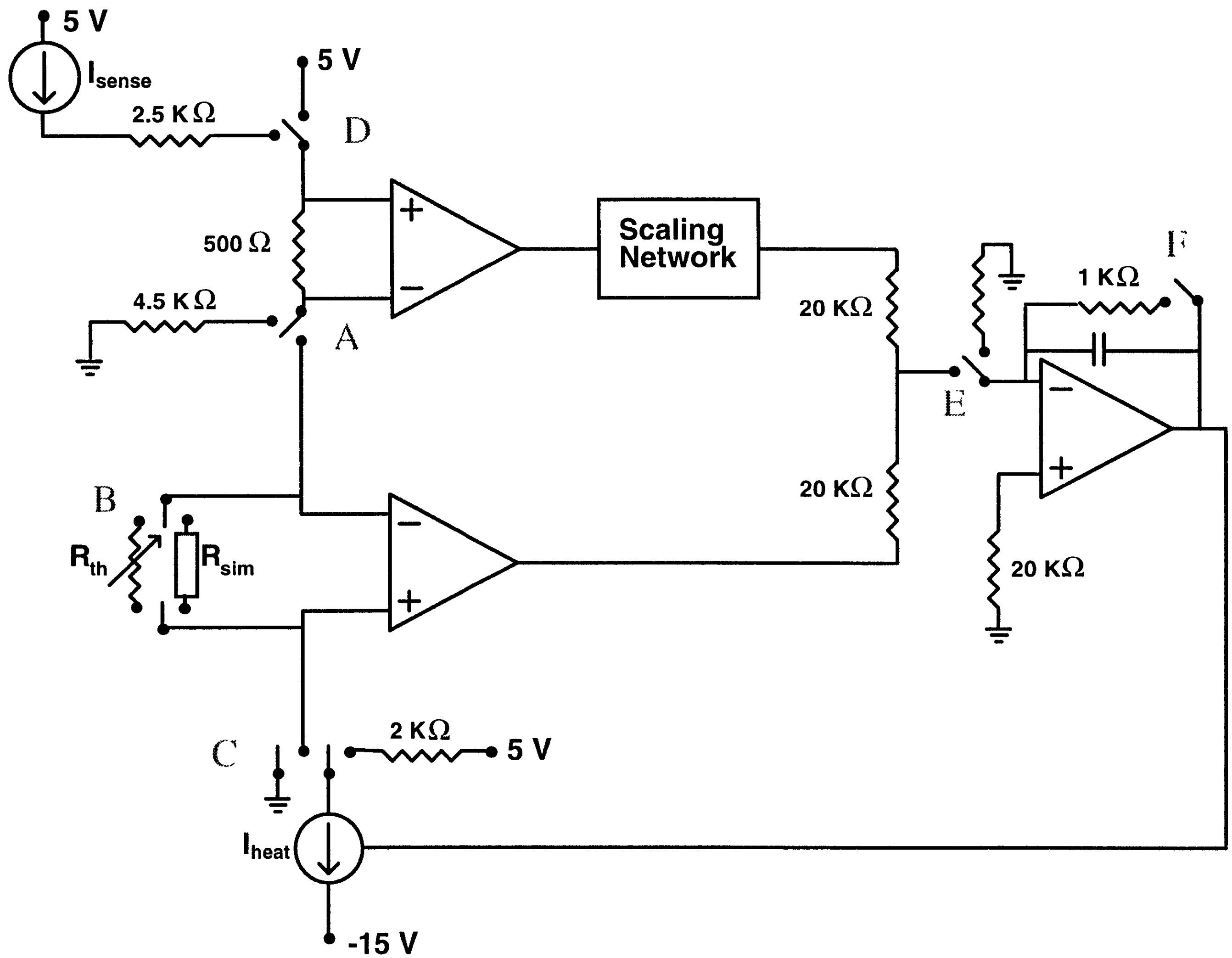


Figure 3-7: Switches to Control Testing and Calibration Modes

The second mode is self-heating using the control loop. For this mode, the fixed resistor and thermistor are connected to 5 V and to the current sink so that the voltage-controlled current sink draws current through both the resistor and the thermistor. Once the scaling network is set for the desired level of amplification, the rest of the control loop is closed using the switches described in Table 3-2.

Table 3-2: Switch Position for Heat Mode

Switch	Position
A	right, connecting fixed resistor to thermistor
B	left, selecting thermistor
C	left, connecting thermistor to current sink
D	right, connecting fixed resistor to 5V
E	down, closing feedback loop
F	open, opamp as integrator

Calibration of this part of the system is slightly more complicated than that for the sense mode. The scaling network must be calibrated before a perfusion measurement to account for any variation in the DAC, the opamp, and the resistors which make up this circuit. To calibrate this network, an appropriate voltage is generated across the 500 Ω fixed resistor. The expected current of 1 mA during normal operation is generated using the 5V source and a 4.5 KΩ resistor to ground. By measuring the voltage across the fixed resistor and the voltage out of the scaling network for two different values of the DAC code, the relationship between the code and the set resistance can be calculated.

As described in Table 3-3, during this calibration, the feedback loop is broken and the current sink is connected to 5 V with a simple 2 KΩ resistor through which it can draw current. The thermistor is disconnected from the 500 Ω resistor and, instead, connected to ground. Since no current flows through the thermistor in this configuration, the switch settings described in Table 3-3 are the default settings used when the probe board is first powered up.

Table 3-3: Switch Position for Heat Calibration

Switch	Position
A	left, connecting fixed resistor to 4.5 KOhm res.
B	left, selecting thermistor
C	right, connecting thermistor to ground
D	right, connecting fixed resistor to 5V
E	up, disconnecting feedback loop
F	closed, opamp as amplifier

The 20 KΩ voltage divider used between the scaling network and the differential amplifier measuring voltage across the thermistor does not require calibration because these resistors are purchased as a matched set. The match is good to 0.01%. Because the

two resistors are packaged together, any drift in temperature should effect both resistors equally.

3.3.2 Testing

Although the measurement control circuit is not particularly complicated, it is composed of several distinct parts: the differential amplifiers, the scaling network, the voltage divider, the integrator, and the two current sources. For testing purposes, it is desirable to have the ability to separate these components from one another, especially those that compose the feedback loop. The switches have been placed to facilitate this testing.

The sense current source, differential amplifiers, and scaling network can be isolated with the normal measurement and calibration modes described above. However, additional configurations are needed to isolate the integrator and the current sink. By setting Switch E to the upper position and closing Switch F, the voltage input to the current sink can be set to approximately zero. When Switch C is moved to the right, roughly 6 mA will be drawn through the 2 KΩ resistor, producing a voltage of approximately 12 V. Test points around this 2 KΩ resistor make it simple to verify the current sink operation. To test the operation of the integrator, Switch F can be opened and the output of the opamp will slowly integrate up. These two simple tests should make diagnosis of control loop problems relatively straight-forward.

3.3.3 Thermistor simulator

Another important testing feature is the inclusion of a thermistor simulator, depicted in Figure 3-8. This thermistor simulator circuit has been used in place of a real probe to test the TDP200 and to measure variation between TDP200 instruments. As an on-board feature, the simulator may be used in place of the thermistor during the calibration procedure, and it allows for basic diagnostic tests without external connections.

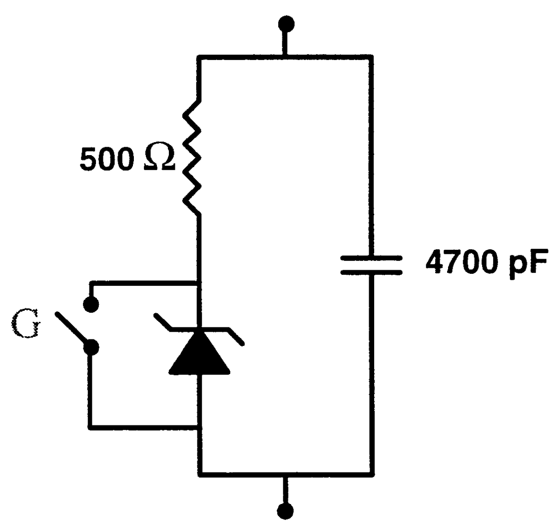


Figure 3-8: Thermistor Simulator

Essentially, the simulator exploits the fact that the effective resistance of a zener diode is inversely proportional to the voltage across it. By combining a zener diode and a resistor in series, a simple circuit can be built with characteristics similar to a thermistor. The effective resistance of the simulator drops as the voltage across it is increased. This feature makes the simulator circuit very useful in testing the control loop.

For testing the circuit in the temperature sensing mode, Switch G is included. With a current of only 50 μA, the voltage across the zener diode is quite high compared to the expected voltage across a thermistor. When Switch G is closed, the zener diode is shorted and the thermistor simulator acts as a simple resistor. For simplicity, Switch G is controlled along with Switch C. Therefore, when the feedback loop is closed, Switch G is automatically opened, and when the thermistor or thermistor simulator is connected to ground, Switch G is closed.

3.3.4 Switches

Most of the switches required for this design can be implemented with CMOS switches. Although the Maxim MAX333A switches have a finite resistance, with a maximum of 35 Ω, this resistance is not critical for most of the switch sites. For example, in sense mode, the resistance of Switches D, A, and C would add at most 105 Ω to the approximately 5 KΩ load for the current source. With differential voltage measurements and calibration of the current source using precision resistors, this extra resistance will have no adverse effect on the accuracy of temperature measurement.

Although 35 Ω is not a problem in most cases, the notable exception is Switch B which controls the selection of either the thermistor or the thermistor simulator. The differential amplifier inputs surround these switches, so the voltage drop across the switches will be included in the voltage measurement used for the feedback loop. If the resistance were absolutely constant and consistent from part to part, the added resistance could simply be accounted for in the calibration procedure. However, a variation as small as 1 - 2 Ω could disrupt the measurement. Therefore, mechanical relays are used for switch B and for the switch to select the thermistor or resistor in the circuit to measure resistance of the sense thermistor. The Aromat TQ2E relays have a nominal resistance of only 50 mΩ, considerably less than the 35 Ω of the CMOS switches.

Aside from the low resistance, mechanical relays are indicated for safety reasons as well. If a thermistor were to become accidentally shorted to the patient, it is absolutely necessary to be able to completely disconnected this thermistor from the rest of the system, particularly for the case of a multi-thermistor probe. CMOS switches can be turned off but do not provide isolation. If the patient potential were to drift more than 15 Volts from the instrument potential, a CMOS switch could break down and allow current to flow. The mechanical separation of the relays provides much better isolation between a disconnected thermistor and the probe board.

The TQ2E relays contain two single-pole double-throw switches controlled by the same solenoid. For this application, the 24 Volt version is ideal. The solenoid is connected to the ± 15 V supplies and draws approximately 8.3 mA in the active mode. With this configuration, the relays send no current into the ground plane. Compared to the 5V version of the TQ2E relay, used in the TDP200, which draws 28.1 mA, the load on the power supply is considerably reduced.

3.4 Measurement circuitry

3.4.1 Analog to Digital Converters

Two Analog Devices Sigma-Delta analog to digital converters are employed for voltage measurement, one dedicated to each thermistor, heat and sense. The AD7712 was chosen for the serial interface, self-calibration, an on-chip programmable gain amplifier, and resolution up to 24 bits. In addition, there are two inputs, selected by a multiplexer, one differential input and one single-ended input. The differential input is used for the sense mode. Using the built-in 2.5 volt precision reference and a 5 V power supply, the differential channel has a range from 0 to + 2.5 V, in unipolar mode, with an absolute maximum voltage of 5.3 V. Given the 50 μA sense current and using a gain of 16, the A/D can measure resistances in the range from 0 to 3125 Ω. This is somewhat larger than the range of the TDP200, with a maximum of 2900 Ω.

To achieve 10 Hz sampling with self-calibration, the first notch of the digital filter must be set to 60 Hz. The self-calibration feature configures the device to measure the zero point and the reference voltage alternately between each voltage measurement, constantly correcting for any drift. Because it takes three cycles to settle between two inputs, the effective sampling rate is reduced by a factor of six. With the first notch at 60 Hz and the gain set to 16, the effective precision is limited to 19 bits according to the Analog Devices data sheet. This precision allows resolution better than 0.01 Ω, compared to the 0.145 Ω resolution of the TDP200. If only 16 bits are collected, the precision is still better than 0.05 Ω.

3.4.2 Sense Mode Measurement Calibration

The differential channel is also used for the calibration of the "sense" current source. The input of the A/D converter is controlled by an analog multiplexer, the Maxim MAX309 2x4 multiplexer. The four channels of the multiplexer are set to the thermistor (or thermistor simulator), the 500 Ω precision resistor, the 2500 Ω precision resistor, and to an open channel. As the voltage across the thermistor during the self-heating mode might exceed the absolute maximum voltage of the A/D converter and cause damage, the fourth channel of the multiplexer is used to connect the differential input to ground during perfusion measurements.

Although the multiplexer has a non-zero on resistance, with a maximum of 100 Ω, this resistance does not adversely affect the measurement. First, the resistance match between channels in guaranteed to 5 Ω. More importantly, the differential channel of the A/D converter has a very high impedance. The specification for the maximum DC input leakage current is only 10 pA. With virtually no current flowing through the multiplexer, any small variation in resistance poses no problem.

3.4.3 Heat Mode Measurement

Calibration of the heat mode also uses the differential channel, for the first part of the measurement. The voltage across the 500 Ω precision resistor is measured by the differential channel. Although the 500 mV across the resistor in calibration is much higher than it is in sense mode (50 mV), the level is still within the range of the A/D, if the gain is reduced. However, the voltage out of the scaling network may be even higher, up to 3 V which is beyond the range of the differential channel, unless an additional opamp is used to attenuate the signal. Therefore, the single- ended channel is used for both this part of the calibration and for measurement of the voltage across the thermistor during self-heating.

With an initial voltage attenuation by a factor of four, the single-ended channel of the A/D converter has a range of 10 V. Using a gain of 1 and the same filter frequency of 60 Hz, the effective resolution is better than 19 bits. This effective precision of 20μV compares favorably to the 100μV precision of the TDP200. The factor of five improvement for the precision of the voltage measurement is especially important because the voltage is squared to determine the power.

Because the thermistor is not connected to ground, an additional differential amplifier is used to generate a voltage to be read by the single-ended channel of the A/D converter. The use of this high impedance differential amplifier also insures that the current through the thermistor is not disturbed by the measurement. The single-ended channel has a relatively low impedance which can be as little as 30 KΩ. Although the output of the differential amplifier has no trouble with a 30 KΩ input, connecting such a low impedance input directly to the thermistor would draw current away from the thermistor and disrupt the feedback loop.

3.4.4 Over-Voltage Protection

Typical voltages across the thermistor during self-heating are on the order of 1 to 2 Volts. However, during the initial transient, the voltage can exceed the 10 V range of the single-ended channel and could even exceed the 11.2 V absolute maximum that the A/D converter can tolerate without damage. A simple circuit, shown in Figure 3-9, serves to protect the converter from brief periods of high voltage. A non-inverting amplifier is used to generate a constant 10 V. The voltage input to the A/D converter is connected to this 10 V with a diode. For inputs less than 10 V, the diode is reverse biased and has no effect on the input to the A/D converter. However, if the input reaches approximately 10.6 V, the diode begins to conduct, maintaining a voltage of less than 11 V at the input to the A/D converter. The 25 Ω resistor is included to limit the current that would flow in an over-voltage condition, protecting the opamp. This resistance is less than 0.1 % of the input impedance of the single-ended channel and should have little effect on the measurements.

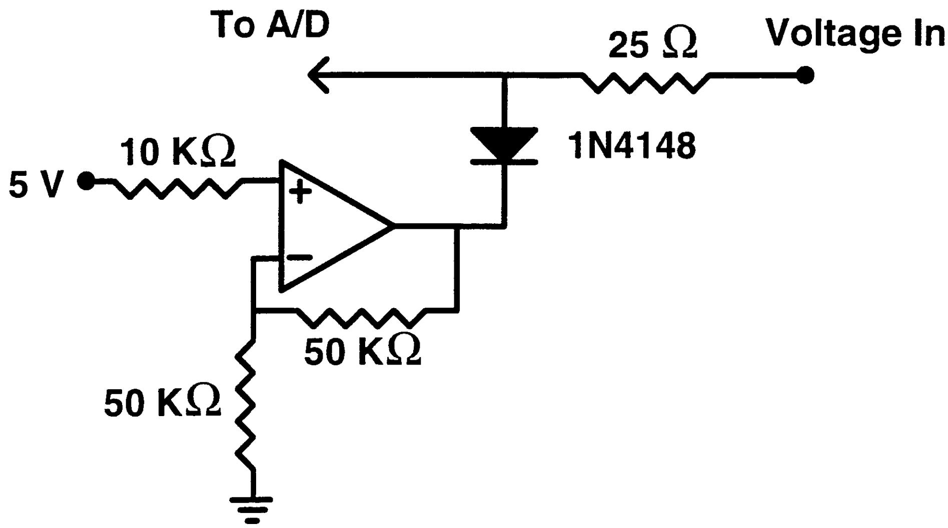


Figure 3-9: Over Voltage Protection Circuit

4. Digital Design

The digital portion of the circuit enables us to coordinate the control of the different modes of the analog circuit, to measure the voltages from the temperature and effective thermal conductivity measurements, and to communicate this information to the host PC. The microprocessor, in conjunction with the analog to digital converters, provides the quantitative information while simple LEDs indicate the status of the board to the user.

4.1 The Microprocessor

A Dallas Semiconductor High Speed Microprocessor is the central part of the digital circuitry. This microprocessor is pin and command compatible with standard 8051 microprocessors which are familiar to many users and for which there are many development tools and a significant amount of support. However, Dallas has improved upon the basic design and created microprocessors that run more efficiently and at higher speeds. Most operations are performed in four clocks cycles as opposed to twelve with the standard 8051. In addition, many of the members of this family of microprocessors include extra features not found on other 8051 microprocessors.

The particular microprocessor chosen for this application, the DS87C520, includes two complete UART serial ports. This configuration is ideal for this project because it allows one serial port to be dedicated to communication with the serial components on the board, the analog to digital converters and the digital to analog converter, while reserving the other serial port for communication with a host PC or master microprocessor.

The DS87C520 has six external interrupts, two data pointers, three timers, a separate watchdog timer, and four ports for I/O allowing up to 32 inputs or outputs. Four lines are reserved for serial communication, but the remaining 28 lines are available to the program because the microprocessor also has a significant amount of onboard memory. It includes 16 KBytes of EPROM which is more than sufficient for the firmware.

Also, aside from 256 Bytes of scratchpad RAM used for special function registers, there is 1 KByte of SRAM available through the MOVX assembly code command. This SRAM can be used to store any necessary values for the firmware as well as buffer data before it is sent to the master microprocessor.

With this application, for each sample of data, there are two bytes of data representing the time in seconds, two bytes of data to represent the milliseconds, two or three bytes to represent the voltage across the sense thermistor, and two or three bytes to represent the voltage across the heat thermistor. A total of up to ten bytes will be saved for each sample, depending on the number of bits read from the A/D converters. At 10 Hz, there will be up to 100 bytes per second. The master microprocessor should collect the data at least once a second. However, if there is a delay, the DS87C520 will have room to store at least five seconds worth of data.

By eliminating the need for external ROM or RAM, we have access to the remaining 28 ports on the microprocessor for input and output rather than requiring 16 of these ports for the address and data interface to memory. This architecture eliminates the need for large and expensive memory chips or extra I/O ports on the board. It not only reduces the real estate requirements, it also greatly simplifies the digital circuitry. There is no need to test or debug the interface between the microprocessor and the memory since it has been taken care of by the manufacturer.

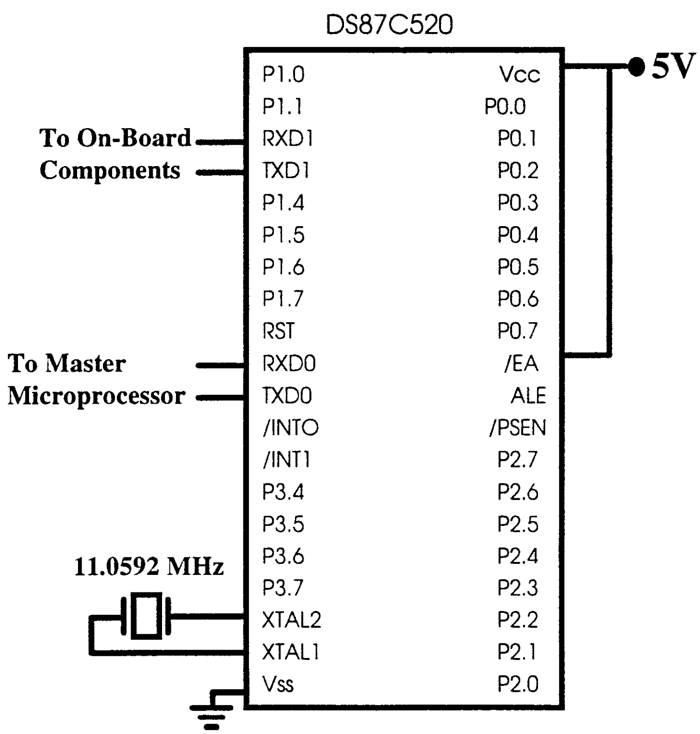


Figure 4-1: Basic Connections for Microprocessor

The microprocessor is powered with a 5 Volt supply, controlled by a separate voltage regulator dedicated to the digital components. As shown in Figure 4-1, the microprocessor is driven with an 11.0592 MHz crystal which allows communication with the master microprocessor at standard baud rates such as 9600 baud. Besides the power connections to V_{CC} and V_{SS}, pin 31 (/EA) is set high to force the microprocessor to use the on-chip EPROM rather than external memory. These three connections and the crystal are all that is required for basic functionality of the DS87C520.

For this application, both serial ports are utilized. Figure 4-1 shows that the first serial port (pins 10 and 11) is dedicated to communication with the master microprocessor or host PC and the second serial port (pins 3 and 4) is connected to the other serial components on the board, the analog to digital converters and the digital to analog converter. Several of the remaining ports are used for controlling the switches and multiplexers necessary for the various measurements.

Table 4-1 lists the connections for each pin of the microprocessor. Port 2 is dedicated to control of the analog circuit and the switches shown in Figure 3.7. Port 2.2 is connected to the "Heat" Relay which is indicated as Switch B in Figure 3.7. This switch selects either the heat thermistor or the thermistor simulator in the heat thermistor circuit. There is a similar relay connected to Port 2.1 which controls selection of the sense thermistor or a fixed resistor in the circuit dedicated to the sense thermistor.

Port 2.0 controls Switch H which controls the input to the single-ended channel of the A/D converter 1 and is used during the calibration of the self-heating circuitry. Ports 0.0 through 0.3 control the multiplexers that select which signal is input to the differential channel of each A/D converter.

Table 4-1: Microprocessor Ports

P0.0	"Sense" Mux 1	P1.0	RFS1
P0.1	"Sense" Mux 2	P1.1	TFS1
P0.2	"Heat" Mux 1	P1.2	SDATA (serial port 1)
P0.3	"Heat" Mux 2	P1.3	SCLK (serial port 1)
P0.4	CTS (serial port 0)	P1.4	A0
P0.5	RTS (serial port 0)	P1.5	SYNC
P0.6	ID 1	P1.6	RFS2
P0.7	ID 2	P1.7	TFS2
P2.0	Switch H	P3.0	RXD (serial port 0)
P2.1	"Sense" Relay	P3.1	TXD (serial port 0)
P2.2	"Heat" Relay	P3.2	INT0 - Safety
P2.3	Switch A	P3.3	INT1 - DRDY2
P2.4	Switch C, Switch G	P3.4	DRDY1
P2.5	Switch D	P3.5	LOAD
P2.6	Switch E	P3.6	ID 3
P2.7	Switch F	P3.7	ID 4

Four ports are dedicated to an ID or numbering system. Using jumper 1, the position of the board in the multi-site system can be indicated. Port 0.6, 0.7, 3.6, and 3.7 are used for the microprocessor to read the jumper and determine a board number from 0 to 15. Port 3.2, interrupt 0, is connected to the safety circuit described in the next chapter. The remaining ports are used for communication with the master microprocessor and the serial components as described below.

4.2 Serial Components on the Probe Board

There are three other components on the board which communicate with the microprocessor through serial communication, the analog to digital converters and the digital to analog converter. A serial interface allows for many bits of data to be transferred using just two lines, requiring fewer traces on the printed circuit board and fewer pins on each component than a parallel interface would have.

4.2.1 Analog to Digital Converters

The Analog Devices AD7712 analog to digital converters can be configured in several different ways. The user must choose the differential or single-ended channel, the setting of the programmable gain amplifier, whether or not to use background calibration, and how many bits will be read from the device (16 or 24). In addition, the frequency of the first notch of the digital filter, which determines the sampling frequency, is also selected. To set these parameters, 24 bits must be sent to the control register. The AD7712 includes a provision to read back from this control register to verify each setting.

Data conversion begins as soon as the device receives power. New data become available at a rate determined by the crystal and the divisor programmed through the control register. When a sample is complete, the data ready indicator (/DRDY) is driven low. After these data are read from the device, the /DRDY signal returns to a high value until a new value is ready. If the data are not collected by the microprocessor, the /DRDY indicator remains low and data collection continues with new data continuously replacing the older data with each new sample. As long as the microprocessor is available to collect data at least as quickly as it is sampled, no data are lost. Because the results of each measurement are saved until the next measurement is complete, there is no need for precise timing between the A/D converter and the microprocessor.

The serial interface between the A/D converters and the DS87C520 is very simple. The serial port is configured for synchronous communication with one line serving as a clock and the other as a bi-directional data line. The microprocessor generates the serial clock at a frequency of the microprocessor crystal frequency divided by 12, 921.6 kHz. At this rate, 24 bits can be transferred in less than 27 µsec. With a sampling rate of 10 Hz, there is ample time between measurements to collect the data.

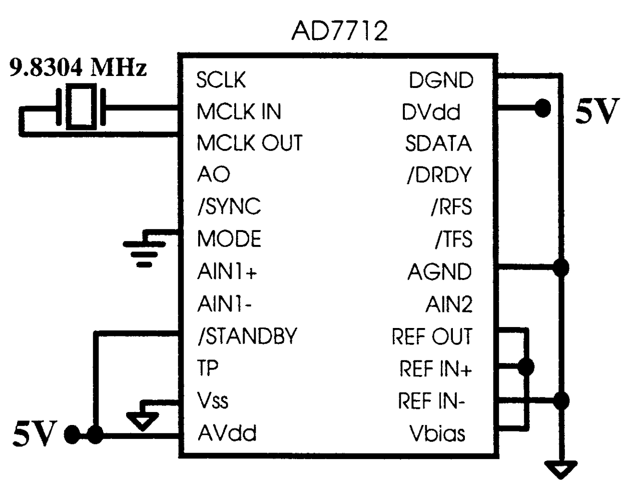


Figure 4-2: Analog to Digital Converter

Figure 4-2 shows the basic connections required for the A/D converters. Since they are used in the unipolar mode, they may be powered from a single 5 V supply. Although there are separate analog and digital voltage supplies on the board, the A/D converters are powered from the analog voltage supply and connected to analog ground. This configuration reduces the noise on the power supply and eliminates the risk that the analog and digital power supplies might drift apart and damage the chip. The exception to the rule about connections to analog ground is the MODE pin which is simply a digital input that is set low for external clocking of the serial interface.

A 9.8304 MHz crystal serves as the clock for each A/D converter. This value can be divided by the appropriate factor to select sampling at exactly 10 Hz. The crystal speed also determines the maximum rate of communication, approximately 1.9 MHz, which is well above the 921.6 kHz actually used.

As shown in Figure 4-2, the serial interface uses pin 1 as the serial clock and pin 22 for the serial data line. These are connected to microprocessor Port 1.3 and Port 1.2, respectively. The serial data line of the A/D converters have tri-state capability allowing both converters to share the same lines. To enable the converters to receive or transmit data on this data line, other inputs are needed. The RFS (receive frame synchronization) must be driven low to read from the device. Likewise, the TFS (transmit frame synchronization) must be driven low for the device to receive data transmitted from the microprocessor. To allow selection of one A/D converter or the other, these lines are kept separate. As shown in Table 4-1, Ports 1.0 and 1.1 are used for the first A/D converter, the one dedicated to the heat channel, and Ports 1.6 and 1.7 control communication to the A/D converter for the sense channel.

The remaining connections to the microprocessor include the address input (A0), the /SYNC control, and the /DRDY (data ready) indicator. The address input, shared by both A/D converters and connected to Port 1.4, allows the user to select whether to read data from the data register or the control register. The /SYNC line, connected to Port 1.5, is used to synchronize the two A/D converters so that they make measurements at the same time. Finally, the two /DRDY lines serve as inputs to the microprocessor. One is connected to microprocessor Interrupt 1. The microprocessor software has been written such that when this indicator goes low, the microprocessor will immediately begin reading the data. The second /DRDY connection is made to Port 3.4. Although this port does not serve as an interrupt, the value of the /DRDY can be polled before data is read to insure that the A/D converter is ready.

Beyond the basic connections, NAND gates are included so that the SCLK signal is only sent to each A/D converter when necessary, not during communication between the microprocessor and the other A/D converter. This use of additional digital circuitry should assure the least amount of noise possible.

4.2.2 Digital to Analog Converter

The interface to the MAX543 digital to analog converter is much simpler than the interface to the A/D converters because data only travel in one direction. Using the same synchronous mode of serial communication and the same baud rate, the microprocessor sends 12 bits to the D/A converter. When the transfer is complete, the LOAD signal, connected to Port 3.5, is used to force the D/A converter to use the new code. There is only one small complication, the 8-bit microprocessor transmits data 8 bits at a time. In order to send 12 bits, at least 16 bits must be sent. Therefore, the software has been written to set the first 4 bits to 0, and the D/A converter uses only the last 12 bits received.

4.3 Serial Communication with the Master Microprocessor

The DS87C520 must be in communication with the rest of the system to receive commands and transmit collected data. The first serial port (pins 10 and 11) is dedicated to this communication. This serial port is configured for asynchronous communication and generates voltages of 0 and 5 V. To meet the specifications of standard asynchronous communication protocols, such as RS-232 or RS-485, serial interface chips must be included. Additionally, the system architecture specifies that each probe board must be electrically isolated from the others and from the master microprocessor. Therefore, the serial connection to the master microprocessor must employ opto-isolators to achieve this electrical isolation.

The probe board uses a serial interface circuit designed by Effective Solutions which can be used for either RS-232 or RS-485 communication. In the short term, RS-232, with the standard tools found on most PC's, will be the preferred communication protocol. However, when the system is complete and several probe boards are used at once, the communication with the master microprocessor would be simplified by using the multi-drop RS-485 protocol.

For either type of communication, high speed opto-isolators (HCPL-2231) are used to provide the necessary isolation of 2500 V. These opto-isolators are connected to the RXD0 and TXD0 pins of the microprocessor for the standard communication. These opto-isolators are also connected to Ports 0.4 and 0.5, which can serve as the CTS (clear to send) and RTS (ready to send) pins for RS-232 communication, and to pin 9 of the microprocessor. This pin is the reset input which forces the microprocessor to restart the software, erase the RAM, and set all of the ports to a high value when the RST input goes high. This RST input is connected, as shown in Figure 4-3, so that it can be accessed by the master microprocessor in case of any difficulty with the DS87C520.

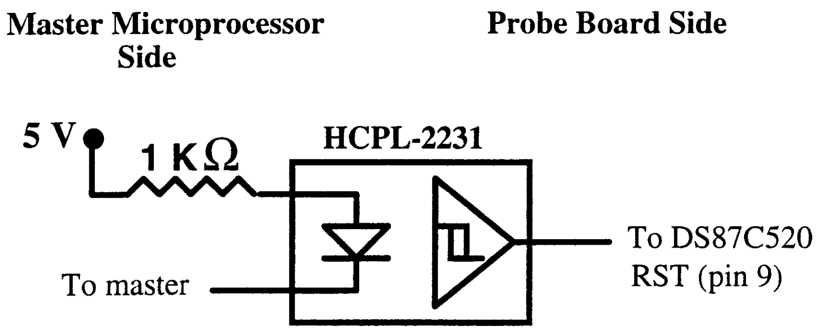


Figure 4-3: Opto-Isolation Circuit

4.4 Indicators

Some portions of the digital circuitry are actually independent of the microprocessor. Although the microprocessor does provide the most informative interface to the probe board, sometimes a simple indicator is more helpful to the user. For this reason, two LEDs are included on the probe board.

The first LED indicates power on and the state of the microprocessor. When power is first applied, the bicolor LED will light red. When the microprocessor comes out of the reset state, the LED will switch to green and the program will begin execution. The second LED indicates that a perfusion measurement is in progress. The yellow LED will light when the board is in the self-heating mode and when the thermistor is in place. Logic gates are configured such that if the thermistor simulator is currently in use, the LED will not light regardless of the current mode.

5. Safety Considerations

Safety is of primary concern with any medical instrument. Although the probes are insulated and should never make electrical contact with the body, there is always some risk when inserting an electrical probe into the body. There are two separate concerns. First, the device might provide a path to ground for another voltage source. Second, the instrument might introduce current to the body.

Obviously, both situations are to be avoided. In a hospital setting, there are many chances for connection to a voltage supply from another monitoring instrument or from any other electrical equipment near the patient. Even small currents can be dangerous when flowing through the body. Aside from the danger of electrical burns, currents in the body can interfere with the normal electrical signals, especially in the heart. Ventricular fibrillation can be caused by low frequency currents as low as 30 mA flowing from one arm to the other. Although fairly high voltages would be required to generate such a current in a healthy person with dry, intact skin, different conditions are likely for a hospital patient. If the skin is wet or pierced by a conductor such as a hypodermic needle, its resistance is greatly reduced. If a patient is instrumented with other electrical apparatus, particularly those which contact the heart, the risk is further increased. Not only do electrical currents pose a threat to the heart and other muscles, but DC currents can cause electrophoresis and metal buildup inside the body. (Oakes)

To protect patients, especially those who may be connected to several devices from different manufacturers at once, national guidelines have been established for safe leakage current levels. Any device contacting the patient must be electrically isolated from all other devices. At frequencies below 1 kHz, at most 10 μA of patient source current is allowed from an instrument. (AAMI)

5.1 Potential Problems

The first step in assessing the risks of a medical device is to imagine all possible fault conditions, no matter how unlikely they are to occur. For each condition, the designer must try to predict the response of the system. If any condition results in an unacceptable response, additional protection circuitry must be included.

5.1.1 Faults on Two Probes

The most basic safety feature of the multi-site perfusion instrument is the isolation of each probe card from the master microprocessor and from each other. Each probe card is powered by its own medical grade DC power supply (Condor MTLL-5W-A) which provides the required isolation. Therefore, the heat thermistor and sense thermistor attached to each probe board share a single ground, but these are isolated from the grounds of other thermistor pairs. This isolation is especially important for the case of multiple single-site probes in use at one time. Such probes, depicted in Figure 5-1, could be placed anywhere in the body, and it is important that no current can pass between them.

With these separate, isolated power supplies, if there is a single failure of the electrical insulation on a probe, the thermistor potential at the site of the break would float to the potential of the patient but would introduce no current or a path to ground for other voltages. If failures were to occur on two separate probes, such as at points A and D in Figure 5-1, each potential would float to the potential of the patient, again without any danger. Since the probes are isolated from one another, there is no circuit for current to flow between them.

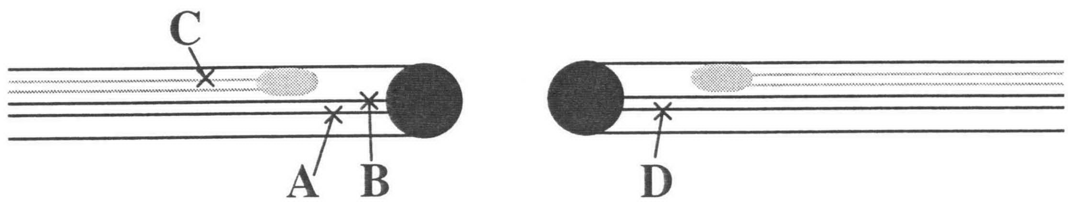


Figure 5-1: Two Single-Site Probes

5.1.2 Two Faults on One Probe

More problematic is the case of two insulation breaks involving a single thermistor pair, such as a thermistor breaking off in the tissue and leaving two lead wires exposed. Since these two faults would be connected to the same power supply, isolation does not help in this case. There are two separate scenarios: two insulation failures on one thermistor (faults at A and B), and one insulation failure on each thermistor (faults at A and C). Each scenario must be considered for both the passive temperature sensing and self-heating modes.

The first case of two faults on one thermistor in the sense mode is fairly straightforward. If the thermistor remains attached but the insulation has been compromised, a small amount of current may flow through the tissue depending on the relative conductance of the tissue and the thermistor. If more than 10 μA flows through the body, this will significantly alter the voltage across the thermistor as the current that is supposed to be flowing through the thermistor is only 50 μA. Any such change could be detected by the software as a sudden jump to an unreasonable temperature. If the thermistor were to become completely detached, this change would only be more dramatic.

The second case is that of two faults on one thermistor that is being self-heated. Unlike the sense mode, a lost of 10 μA would not be immediately obvious when compared to the approximately 1 mA flowing through the thermistor during heat mode. However, it is reasonable to assume that the conductance of the tissue over a short distance (such as between points A and B) would be less than or comparable to that of the thermistor. If such a fault were to occur, the voltage across the thermistor would drop drastically. The feedback loop would then compensate as if the temperature were too high by reducing the current until the circuit was balanced. This balance would most likely not be achieved until there was zero current. However, even if the feedback loop failed to drive the current to exactly zero, the software could easily recognize such a drastic change in power requirements. Therefore, no additional circuitry is required to protect again the case of two faults on one thermistor.

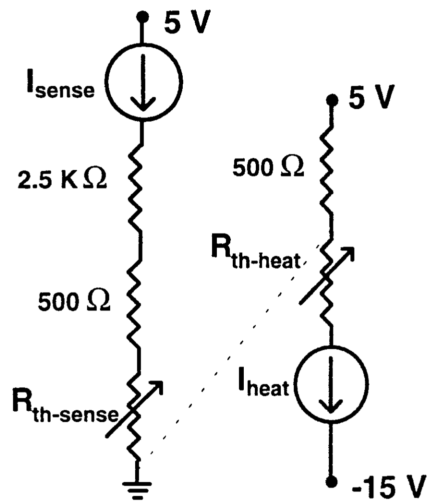


Figure 5-2: Possible Fault between Thermistors

The case of two faults on different thermistors, one on the heat bead and one on the sense bead (e.g. points A and C), is more difficult to handle. Exactly where current would flow would depend on where the faults occurred and the potential of each thermistor. One scenario would be a fault on the ground side of the sense bead and one on the positive side of the heat bead during heating. If the circuitry for the sense thermistor were designed just like that for the heat thermistor, the circuit would look like that in Figure 5-2. In the case of faults at the two sites at either end of the dotted line, the feedback loop would reduce the current that the heat current source was drawing, but there would still be a danger. The 5 Volts at the top of the heat loop would be connected through the body to ground with just 500 Ω in series with it, providing up to 1 mA of current through the body. This scenario necessitates further safety protections.

5.2 Leakage Current Protection Circuitry

With some simple design changes, the circuit can provide protection against all possible scenarios involving two faults. Instead of the sense thermistor being connected directly to ground, the 2500 Ω precision resistor is placed in series with it, as shown in Figure 5-3. Although this placement of the calibration resistor does increase the common mode voltage across the thermistor, this increase can be calibrated out in the temperature to resistance calibration. By measuring the voltage across the 2500 Ω resistor, we can determine the current that is returning from the thermistor. Using a high impedance non-inverting amplifier, the measurement is made without disturbing the basic circuit.

The sense current source provides 50 μA of current. Although it is not built with precision components, using 1% resistors, we can expect that the current source would be accurate to at least 5% and probably be better than that. A deviation of 10 μA would represent a 20% increase or decrease in the voltage. The circuit shown below will detect such a deviation using two comparators. If the current exceeds the acceptable range, Interrupt 0 (INT0) is driven low so that the microprocessor may switch the thermistors out of the circuit to stop the current leakage. This interrupt is the highest priority external interrupt and can be serviced immediately. In addition, jumpers may be set so that the safety circuit automatically disconnects the thermistors. This hardware safety feature is completely independent of the microprocessor and provides an even greater measure of safety.

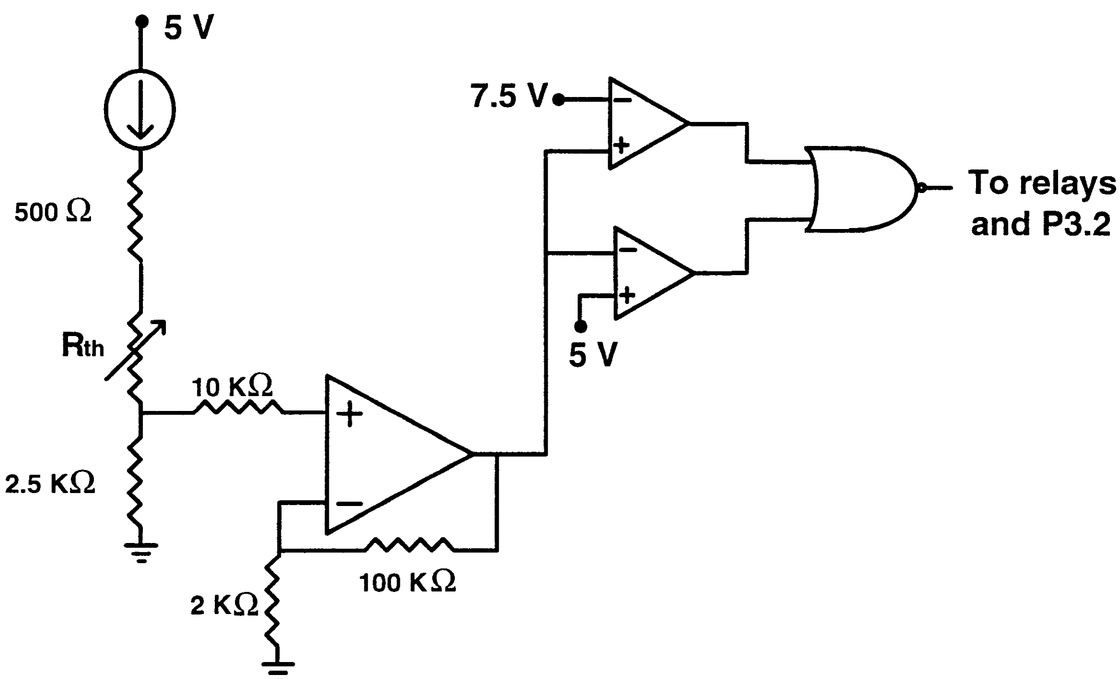


Figure 5-3: Leakage Current Protection Circuitry

This protection circuit, in conjunction with the other features of the basic design, provides all the necessary safeguards against current through the body. If current is leaking from one thermistor and returning back through it, the software will detect the problem or the feedback loop will drive the current to zero. If more than 10 μA of current is flowing between the two thermistors, in either direction, the voltage across the 2500 Ω resistor will slip out of range and trip the safety circuit.

6. Software

The perfusion measurement circuitry is not useful without a way to control the measurements and collect the data. The Dallas DS87C520 microprocessor must be programmed to communicate with the master microprocessor or host PC and the A/D's and to control the switches to change the measurement modes. The software that controls the microprocessor is a critical part of the design of the measurement system.

The Dallas High Speed microprocessors use standard 8051 code and may be programmed with that assembly language. Once assembled, the program may be loaded into the 16 KBytes EPROM resident on the DS87C520. This microprocessor is flexible enough to handle up to 64 KBytes of ROM using an external memory chip, but the on-board ROM is sufficient for this task.

Although it is possible to program directly in assembly language, for the complex task of writing this firmware, a higher level language was desirable. The Systronix BCI51 Basic Compiler proved to be an excellent tool for this project. The firmware is written in a BASIC very similar to QuickBASIC. The advantages of BASIC are clear. It is well known and fairly simple to work with. Most programmers have at least some BASIC experience and would have no trouble upgrading this code if changes were required in the future. However, BASIC interpreters can be very slow. With the BCI51 compiler, speed is not a problem. Like other compiled languages, it runs very quickly and makes efficient use of memory resources.

For this particular application, BCI51 has additional advantages. As a compiled language, interrupts are serviced immediately rather than at 5 msec intervals like the BASIC-52 interpreted BASIC often used with 8051 microprocessors. Also, the Systronix program allows for in-line assembly code to handle any tasks which are not specifically supported by the BASIC commands. When the code is compiled, the BASIC commands are converted to assembly language, then the entire assembly language code is assembled to the hexadecimal executable run by the microprocessor.

6.1 Tasks of the Software

The main duty of the software is to collect data from the A/D converters. The software must not only read from the A/Ds, but must configure the circuit in the proper mode for the measurement and keep track of what type of measurement is currently underway. In addition, the software must facilitate communication with the master microprocessor or a host PC.

The user, through the host, must be able to start and stop measurements and view the data. Before measurements are taken, the software must perform the necessary calibrations. For sense mode measurements, the voltage read by the A/D is calibrated against two known resistances. From these data, the relationship between voltage and resistance is determined. Prior calibration of an individual probe determines the relationship between this resistance and temperature. Heat mode measurements require calibration of the scaling network to account for any variations in the D/A converter and the amplifier between different probe boards. This calibration will insure that the same probe will measure the same conductivity in a given medium regardless of which particular board is used to control that measurement.

The software must also incorporate some simple safety features. Upon startup, the circuit needs to be in a known, safe state with no current running through the thermistors. The software needs to be robust enough to handle input errors from the host due to either an error by the host or communications errors. Finally, as a safeguard against a runaway program, the software must utilize the watchdog timer included on the microprocessor so that the system will be reset if an inordinate amount of time passes without communication between the host and the microprocessor.

6.2 Structure of the Software

The software can be understood as being comprised of three separate sections. The first section handles the initialization of the circuit itself and of the variables which will be used later in the software. The second section prompts the host for inputs and responds to commands. The third part of the software includes the interrupt routines and other subroutines that control the measurements and data collection.

To allow for up to eight probe boards to be controlled by a single host, it is important that data is sent to the host in such a way that the data can be associated with the proper board and that the host is not overwhelmed by data. In addition, the data collection system must allow for the possibility that some boards may be performing measurements while others are not. To simplify the data acquisition tasks of the host, the probe boards do not send data as soon as it is collected. Rather, the host must poll each board and "ask" for any data acquired since the last transmission to the host. To accommodate this method of data transfer, the software must implement a circular buffer. Data buffering is also included in the third part of the software.

6.2.1 Software Section 1: Initialization

The first task of the software is to set the switches and relays to a known state, initialize the serial ports, and initialize the necessary variables. When the microprocessor is reset, either by a reset signal on pin 9, power on, or the by watchdog timer, a standard reset procedure is started and the program in the EPROM begins. Even before the program starts, the reset forces all of the ports high to allow them to be used as inputs. Therefore, the circuitry has be designed so that when all ports are high, the circuit is in the heat calibration state which is a safe resting state in which no current is running through the thermistors. Table 6-1 indicates the position of each switch for high and low inputs.

Table 6-1: Analog Switches and Controls

		Controls:		Position for:		
Switch	Description	1-position	0-position	Sense Mode	Heat Mode	Calibration
A	Ro to 4500 Ohm/thermistor	4500	therm.	0	0	1
B	thermistor/simulator	simulator	therm.	0	0	0/1
C	thermistor to GND/heating current source	GND	current source	1	0	1
D	Ro to 5V/"sense" current source	5 V	current source	0	1	1
E	input to integrator	GND	voltage divider	1	0	1
F	feedback for integrator	amplifier	integrator	1	0	1
G	zener diode in/out	out	in	1	0	1
H	input to A/D	scaling net.	diff. amp.	1	0	1

Once software execution begins, the watchdog timer is disabled briefly to allow enough time for the initialization tasks. BCI51 automatically handles initialization of serial port 0 which is used for communications with the host. The baud rate is set to 4800 baud. While the microprocessor has the capability to go much faster, the communications may be limited by the opto-isolators.

The amount of data and code memory used are sent to the host first. This information was very useful during the development of the firmware and will now serve as a basic check that the microprocessor and the communications are functioning. The software then reads the board position number from jumper 1 connected to Ports 0.6, 0.7, 3.6, 3.7 and reads a serial number which has been loaded to the EPROM.

The second serial port on the microprocessor is not directly supported by the BASIC language. Therefore, the initialization, reading, and writing must be handled with assembly language. The port is set up for synchronous operation (Mode 0), at a baud rate of 920 kbits/second. After the serial port has been initialized, the A/D converters can be configured. By writing to the control registers of each A/D, the gain, mode, and speed of each is set. The software then reads from the control registers to assure that the proper values are in place.

6.2.2 Software Section 2: Inputs from the Host

After initializing a few variables, the program prompts the host for an input. This allows the program to wait until the user is ready to begin. After receiving this go ahead, the real-time clock is started, the watchdog timer is enabled, and an interrupt is set up to read data from the A/D's.

Although no real data is collected before the host requests a measurement, the A/D converters have already begun reading voltages as determined by the configuration information sent during the initialization process. At 100 msec intervals, the A/D's have data ready to be read by the microprocessor. The DRDY (data ready) output of each A/D is set low when data is ready and stays low until the data has been read out. The interrupt routine that is used while waiting for the first command simply reads this data and ignores it. This clears the DRDY low resulting from any old data before a real measurement begins.

Once the first input has been received, the program is forwarded to the general input section which handles all commands thereafter. The program then waits for a two character input from the host. While it is waiting for a new command, interrupts are still serviced. If a measurement is in progress the data is stored to the buffer until the command to print the data to the host or to quit the measurement is received.

Table 6-2 lists the commands that the software recognizes. The first letter of each command indicates the basic category of the command. There are five categories. An "s" command relates to a "sense" measurement. An "h" command controls a "heat" measurement. The "p" commands force the program to print information to the host. The "q" commands are used to quit measurements and to set up special modes for circuit testing. Finally, the "d" commands control the debugging mode.

Table 6-2: Possible Inputs to Software

s0	sense mode - sim (sense channel), sim (heat channel)
s1	sense mode - therm, sim
s2	sense mode - sim, therm
s3	sense mode - therm, therm
h0	heat mode - sim, sim
h1	heat mode - therm, sim
h2	heat mode - sim, therm
h3	heat mode - therm, therm
h8	calibrate heat mode
p0	print out data
p2	print out test line to check communications
q0	stop data - set to cal mode
q2	stop data - leave in present state - turn off watchdog
q4	stop data - set to test mode - watchdog off, set DAC
d0	turn debugging mode off
d1	turn debugging mode on

For the two measurement commands ("s" and "h") the second character generally controls the state of the relays so that either the simulator or the thermistor can be used for each sensor. A "0" indicates that both simulators should be used. For a "1", the sense channel uses the thermistor and actually measures temperature, but the heat channel uses the simulator. For a "2", the reverse is true, the heat channel uses the thermistor and the sense channel uses the simulator. If the user wishes to make measurements with both thermistors, the second input should be a "3".

For the heat mode, there is an additional option for the second input. An "8" indicates that the program should perform a calibration of the scaling network so that it may later be set up for a perfusion measurement. Since this calibration must be performed before the digital to analog converter can be set for a particular set resistance value, if the user requests a measurement before calibration, the calibration is automatically performed. For future measurements, the software proceeds directly to the measurement unless the "h8" command is invoked to force the system to recalibrate.

Once a command is received, the watchdog timer is reset. The watchdog timer needs to be reset at least every 6.07 seconds. During normal operation, the host should be requesting data at least once a second. If the host fails to send any commands to the probe board in 6.07 seconds when the watchdog timer is enabled, it is assumed that there is a problem and the microprocessor is reset.

After acknowledging a command and resetting the watchdog timer, a subroutine reads the command and calls any necessary subroutines and sets up an interrupt routine to handle the data acquisition. These subroutines and interrupt routines make up section three of the software.

6.2.3 Software Section 3: Subroutines for Measurement and Data Collection

Appendix C contains the source code for the software in its entirety. In this section, the basic operation will be discussed, but details can be determined from the source code. The subroutines in section three include the implementation of the data buffer, calibration routines, data output routines, and controls for the measurement mode of the circuit and the software.

The data buffer resides in the 1 KBytes of onboard RAM. Six hundred bytes are set aside, allowing for 10 bytes of data for each of 60 measurements. With data collection at 10 Hz, this circular buffer is sufficient for six seconds of data without any data loss. If the host does not request new data in six seconds, data would be overwritten. However, this also coincides with the time for a watchdog timeout.

Once a calibration or measurement is started, data are written to the buffer. The first two bytes are generally used to store the time in seconds for that particular line of data. The next two bytes generally store the millisecond component, with a precision of 5 msec. Currently, two bytes are used for the voltage across the heat bead and two bytes for the sense bead. In the future, if the A/D's were used in the 24-bit mode rather than the 16-bit mode, the two remaining bytes could be used to store this extra data.

A print command is used to read data from the buffer. The "p0" command sends the latest data to the host. Each measurement is represented on a separate line beginning with the board number originally read from the jumpers. This board identification allows the host to distinguish data from separate probe boards. The remainder of the line includes the time in seconds, the millisecond component of the time, and the A/D readings, all separated by spaces. The first A/D reading is from the "heat" A/D and the second from the "sense" A/D. The A/D data are saved and sent as 16 bit integers. The lowest possible reading is 0 and the highest is 65535. For sense mode measurements, these values are spread over a range of 156.25 mV (2.5 V range and a gain of 16). For heat mode measurements, the entire range of 10 V is used. A typical data record might look like the following:

```
1 35 150 21845 27589
```

Depending on how many measurements have been recorded since the last request for data, as many as 60 lines of data could be sent or as few as zero before the "end of record" statement is sent. Aside from the "p0" command used in standard operation, a "p2" command has been provided to assist in debugging the system should problems arise. When this command is invoked, the software sends a simple test string of

ABCDEFGHIJKLMNOPQRSTUVWXYZ

to allow for diagnosis of possible communications difficulties.

All regular measurement records follow the same protocol. The exception is for the calibration records. As described above, two different types of calibrations are performed. For the sense mode calibration, the circuit is configured for a sense measurement. The analog mutliplexers, which control the inputs to the differential channels of the A/D's, are then set to measure the voltage across the 500 Ω calibration resistor and then to the 2500 Ω calibration resistor. Three measurements are taken for each setting. This process, which is completed in less than one second, is repeated each time a sense mode measurement is started to compensate for any drift in the sense current source over time.

For both the sense mode and heat mode calibrations, the data are saved to the data buffer. To distinguish calibration data from measurement data, the bytes generally used to indicate the time of the measurement are instead used to record the type of calibration information that record includes. For the sense calibration, the first value is set to "0" and the second value is set to "0" to indicate the 500 Ω value and "2' to indicate the 2500 Ω value. An example of the typical sense calibration output follows.

```
1 0 0 10570 10890
1 0 0 10571 10890
1 0 0 10570 10889
1 0 2 52852 53905
1 0 2 52852 53906
1 0 2 52853 53906
1 47 270 21145 23787
1 END
```

The second to last line is an example of the first line of real data from that temperature measurement. The final line includes the board number and an "END" statement which indicates to the host that the data transfer is complete.

The calibration procedure for the heat mode is somewhat more complex. First, the circuit is configured for calibration such that there is 5 V dropped across the 500 Ω precision resistor and the 4500 Ω resistor. The resulting 1 mA of current approximates the typical current drawn during self-heating. The voltage across the precision resistor is measured with the differential channel of the A/D converter so that the input to the scaling network is known. With the DAC code set to 0, the output of the scaling network is measured with the single-ended channel of the A/D converter. Finally, the DAC code is set to 4095, the maximum value for the 12-bit DAC, and the output of the scaling network is again measured. These data are saved to the data buffer with a first value of "10" to indicate a heat mode calibration. The second value is "0" for the input to the scaling network (measured over a range of 0 to 625 mV), "2" for the output with the DAC set to 0, and "4" for the output when the DAC is set to 4095. The output values are measured over the full range from 0 to 10 V. Each measurement is taken and saved three times; however, only the final value for each measurement is saved by the software to be used for calculations. After this stage of the calibration procedure, the data output might look like the following.

```
1 10 0 52444 23790
1 10 0 52443 23788
1 10 0 52443 23786
1 10 2 19665 23786
1 10 2 19666 23786
1 10 2 19665 23784
1 10 4 3277 23785
1 10 4 3278 23784
1 10 4 3278 23783
```

The second voltage value is simply the reading from the second A/D converter. It may be the voltage across the fixed resistor used as the simulator, or it may represent the temperature of the sense thermistor.

After these three measurements are complete, the software prompts the host for a desired "set" resistance for the effective resistance of the scaling network. The value must be input as an integer in tenths of Ohms (1 KΩ would be entered as 10000). Using this information and the information from the calibration, the software computes the appropriate DAC code for this effective resistance. The code is saved to the data buffer with a second value of "6" and then sent to the DAC. The output of the scaling network is measured again to provide assurance that the desired resistance has been selected. The measurement is repeated three times and saved to the buffer with a second value of "8." Finally, the last reading from the scaling network is used to calculate an effective resistance which is stored in the buffer with a second value of "10." For the 1 KΩ setting, the output from the second stage of calibration might be as follows.

```
1 10 6 3277 0
1 10 8 6554 23780
1 10 8 6555 23781
1 10 8 6555 23781
1 10 10 9999 0
1 63 450 0 0
```

The final line indicates the time at which the switches were changed so that the actual heating measurement began. Subsequent data records would include timestamps and non-zero data from the two A/D converters.

Data collection continues until a command to start a new measurement or to quit is received. The basic quit command, “q0,” stops the program from saving data, configures the circuit in the calibration mode with no current running through the thermistors, and returns to wait for another command. The remaining quit commands are designed for use during board testing procedures. The “q2” command stops the data acquisition as does “q0,” but the circuit is left in the present state and also disables the watchdog timer. In this way, the operator place the circuit in either the sense or heat measurement modes and it will remain in that mode to allow for testing until a new command is issued. Without such a feature, the system would be reset in 6.07 seconds. To offer an additional testing state, “q4” stops the watchdog timer as does “q2,” but this command also sets the circuit to a testing mode. The switches are set so that 5 V is dropped across the 500 Ω resistor and the thermistor simulator. The feedback is disconnected, the DAC code is set to the highest possible value, and the output of the scaling network is connected to the input of the A/D.

The final category of commands is the debug commands which are also designed for system testing. For normal operation, the debug mode is off. When the debug mode is enabled with “d1,” measurements continue, but the way in which the data are presented is changed. First, the A/D converters are reconfigured to operate at 2 Hz rather than 10 Hz. The data is then sent to the host after each measurement instead of being saved to the data buffer. To simplify the use of this mode, the data are converted to mV values before printing. This saves the user from having to scale the A/D counts to a voltage that could be measured with an oscilloscope. With data automatically sent to the host, there is no need to use the print commands and inputs might be spaced further in time. Therefore, the watchdog timer is disabled during the debugging mode. When the “d0” command is issued, the watchdog timer is enabled again, and the data logging returns to normal operation. When the next print command is received, the data buffer may contain data from before and after the debugging session; however, the timestamps will clearly indicate which data is which.

7. Performance Evaluation

After completion of the design phase of the project, formal schematics were created with the assistance of Effective Solutions, Inc. (see Appendix A). Using these schematics, the circuit was laid out and a printed circuit fabricated. Once the printed circuit board was populated, tests were conducted to determine the basic functionality of the circuit and to find any errors in the schematic or layout. Subsequent tests evaluated the performance of both the sense mode and the heat mode of operation, using simulators and real perfusion probes.

7.1 Basic System Validation

Before any testing could begin, the board had to be configured properly. For these tests, the transformer was used to generate the isolated voltage for serial communication, and RS-232 communication was selected. For simplicity the safety circuit was disabled. The communication baud rate was set to 19200 with the software. Table 7-1 shows the jumper settings for these tests.

Table 7-1: Jumper Settings

JP1	4th input to MUX 1	pins 1 & 2 connected
JP2	4th input to MUX 2	bottom two connected to top two
JP3,JP4	safety circuit	right two pins to disconnect safety
Z1-Z3	serial interface	right two pins to select RS-232
J1	board ID #	all open, ID # =15

In the final system, J1 will be used to indicate the board number in a multi-site system. When a shunt is connected from the top to the bottom, this indicates a low value. One peculiarity of the layout is that the bits are not in order from left to right. The far left pair of jumpers controls the 4 bit. The next one controls the most significant bit, the 8 bit. Moving on to the right, the next pair controls the 2 bit, and the right-most pair controls the least significant bit.

The first test, when power was first applied and the microprocessor was in the reset state with all ports high, was very encouraging. Each test point voltage was near the expected value shown in Table 7-2. At startup, the circuit is configured for calibration of the heat mode. As shown at TP1, 0.5 V are dropped across the 500 Ω precision resistor, but there is no voltage across the thermistor. The integrator is configured as an amplifier with an input of 0 V. Therefore, the input to the heat mode current source is 0 V, yielding a current of approximately 5.5 mA. For the sense channel, the current source drives current through the precision calibration resistors and through the 2 KΩ resistor which acts as a thermistor simulator.

Table 7-2: Expected Voltages at Key Test Points

Test Point	*Description*	*Expected Value*
TP1	diff. amp. for 500 Ohm res.	0.5 V
TP2	diff. amp. for heat thermistor	0 V
TP3	output of gain of 11 amp	depends on DAC code
TP4	input to gain of 11 amp	TP3/11
U8, pin 17	voltage divider	(TP2+TP3)/2
U5, pin 8	output of integrator	0 V
TP9	bottom of resistor connected to current source	-6 V
TP10	top of resistor connected to current source	5 V
TP5	top of sense thermistor	225 mV
TP6	ground	0 V
TP7	bottom of sense thermistor	125 mV

However, like most printed circuit boards, the first version is not without errors. There were a few layout errors, and a couple of design errors in the schematic. The layout errors were rectified quite easily. The position of diodes D7, D10, and D11 must be reversed. For proper operation, they must be connected opposite the direction of the silk-screen on the printed circuit board. If they are connected as indicated, the LED lights red during standard operation and, more importantly, the polarity of the isolated power generated for serial communication is reversed. There was also an error in the placement of the zener diode, D1, used for the thermistor simulator on the heat channel. The diode must be inserted as shown in Figure 7-1.

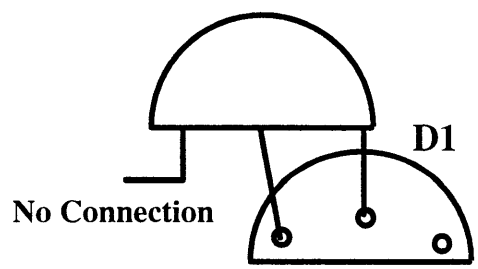


Figure 7-1: Proper Position of Zener Diode

The design errors are somewhat more complicated, but also easy to fix. Port 0 on the microprocessor, which does not require pull-up resistors when used as a memory interface, does require pull-up resistors when used as an I/O port. Because this port was used as a memory interface on the development board used to test the software and communications with the A/D converters, this problem was not discovered until the printed circuit board was complete. Using a resistor pack, 100 KΩ pull-up resistors were added between pins 32-39 and pin 40 on the solder side of the printed circuit board, solving the problem.

The final error is an example of why every circuit should be prototyped. The connection between the microprocessor and the A/D converters was tested using a simple digital inverter on the serial clock line. With this configuration, communication was very reliable. However, there was the concern that with two A/D converters sharing the same serial clock line, the A/D converter would be subject to the noise generated by this line even when the other A/D converter was using it. To try to decrease the electrical noise seen by either A/D converter, a small change was made to the design right before the final schematics were created. Instead of just using an inverter, NAND gates were employed so that the serial clock signal would only propagate through to the A/D converter if its RFS or TFS line indicated that communication was taking place.

Unfortunately, this minor design change had a major, unforeseen effect. Bits are transferred between the microprocessor and the A/D converter on the falling edge of the clock signal. With the NAND gates added, an additional falling edge was generated when RFS or TFS was driven low. Since this falling edge occurred before the data were available, the communication failed to work properly. Therefore, the NAND gates must be removed. If U17 is removed from the board, and jumps are made to connect U17 pins 6 and 8 to U15 pin 12, the A/D converters function properly.

7.1.1 Safety Circuit Validation

Once the basic circuit was validated, the safety circuit was reconnected with jumpers JP3 and JP4. The circuit functions correctly. During normal operation, the input to the comparators is approximately 6.25 V. If the voltage exceeds 7.5 V or drops below 5 V, Port 3.2 of the microprocessor is driven low, and, if the jumpers are set correctly, the safety circuit forces the relays back to the position for the thermistor simulators. The only trouble with this feature is that the software interrupt to handle the safety has not been written. Therefore, if the safety circuit is implemented and somehow tripped, such as by attempting a measurement on a probe that is not connected, the hardware switches the relays back to the simulator position, but the software does nothing. When the relays are back in the simulator position, the input voltage returns to a normal value and the safety circuit allows the relays to switch back to the probe position, repeating the cycle. Without a software interrupt that makes a permanent change in the relay position when the first safety problem is detected, the safety hardware drives the relays back and forth between the two settings until the operator issues a new command that does not involve the probe.

7.2 Testing of Sense Mode

After verifying the basic operation of the circuit, more careful tests were conducted to evaluate the performance of the circuit in sense mode. When a sense mode measurement is started, using either the simulators or a real probe, the voltages across the precision calibration resistors are measured first. The software has been designed to take three measurements across each resistor and send the data to the host. Eventually, software running on the host PC will use this data to determine a relationship between the A/D converter counts and a true resistance. For these tests, the three measurements were averaged and a slope and intercept were calculated for each channel using the average values and the known resistances of the calibration resistors.

The most basic criteria for determining performance of the sense mode is to evaluate these calibration data. Given the 50 μA current and the gain of 16 on the differential channel of the A/D converters, the expected values can be determined. For the 500 Ω calibration resistor, the A/D converter output should be approximately 10486. For the 2500 Ω resistor, a value of 52428 is expected. The actual values are quite close to these expected values.

Initial tests with the on-board simulators yielded typical calibration values. The slope for each channel was 0.046 Ω/Count and the intercept was less than 0.2 Ω. This calibration could be accomplished with a single measurement, but two separate measurements add an extra assurance of linearity. Ideally the intercept would be exactly 0.00 Ω, but, due to small amounts of noise or to imperfect common mode rejection of the differential channel of the A/D converters, there is some variance. An intercept of less than 0.2 Ω, therefore, is quite acceptable, particularly if the measurement is repeatable.

7.2.1 Repeatability Tests with On-Board Simulators

For the sense mode, the most important measurement criteria is that the resistance measurement is repeatable. An accurate resistance measurement is desirable, but it is absolutely critical that when two identical measurements are made at different times, one can be sure that that means that the temperature of the thermistor was the same at each time. To determine the repeatability of the sense mode measurements, a series of ten measurements were performed. For each measurement, calibration data were collected and then resistance data were collected over a period of 2 seconds (20 samples). The calibration constants for each run were used to convert the A/D converter counts to Ohms, and the results were compared over each 2 second window and between the 10 runs. These tests allow comparison of both the calibration coefficients and the resistance readings over the series of measurements.

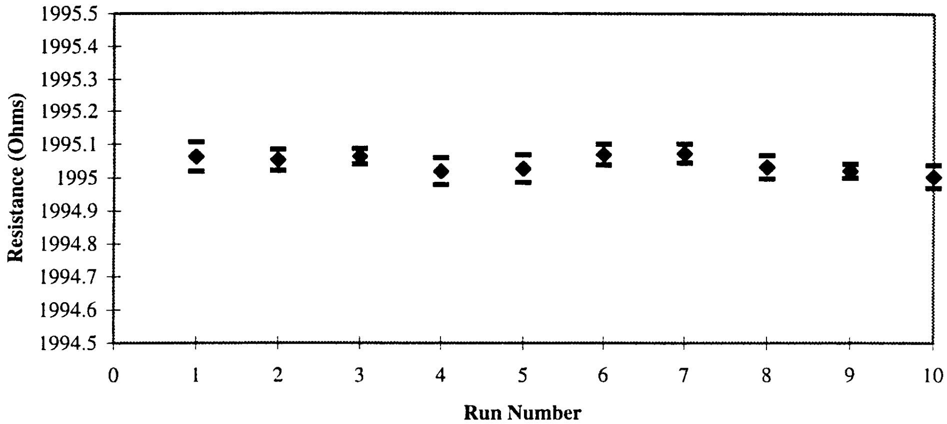


Figure 7-2: Repeatability Tests with On-Board Simulator

The data from the repeatability tests using the on-board thermistor simulators are shown in Figure 7-2 and Table 7-3. For the sense channel, a 2 KΩ 1 % resistor acts as a simulator. For the heat channel, during sense mode measurements, the zener diode is shorted out with an analog switch. Because the switch has non-zero resistance, the effective resistance of the switch and the 500 Ω precision resistor is approximately 600 Ω. The repeatability is quite good. Over the ten runs the calibration coefficients are remarkably close. For the heat channel, the slope is 0.04641 Ω/Count ±0.002 % and the intercept is 0.16 Ω ±0.03 Ω. Similarly, for the sense channel, the slope is 0.04629 Ω/Count ±0.001% with an intercept of -0.00 Ω ±0.02 Ω.

The resistance readings are also quite close. Over each 2 second window, the deviation of the resistances was less than 0.03 Ω and 0.05 Ω for the heat and sense channels, respectively. When the average over each window is compared over the ten runs, the standard deviations are even tighter: 0.02 Ω for the heat channel and 0.03 Ω for the sense channel. Given the expected precision of 0.05 Ω with 16 bit data from the A/D converters, the repeatability is as good as could be expected.

7.2.2 Repeatability Tests with External Resistors

Although these results are very encouraging, the instrument is much more useful when taking measurements using a real probe rather than the on-board simulators. As such, additional repeatability studies were performed using two sets of precision resistors connected in place of a standard probe. A shielded probe cable, approximately six feet in length, was connected to the single probe connector directly on the board, as opposed to the connections made at the back plane. The probe cable was then connected to an unshielded cable with 1000 Ω and 1200 Ω precision resistors. The same repeatability experiments were performed using these "probes."

The results from these tests are also tabulated in Table 7-3. Although there is more noise present when using the probe cable than for on-board simulators, the repeatability is still very good. The deviation between runs is always less than 0.08 Ω and over the two second windows is less than 0.4 Ω. For each test, the variability between runs is less than the variation over the two second window. Clearly, the current source performance is quite repeatable.

Table 7-3: Sense Mode Repeatability

Condition	Slope Mean - Ω/count	Slope Deviation Ω/count	Intercept Mean (Ω)	Intercept Deviation (Ω)	Resistance Mean (Ω)	Max. Deviation 2 s. (Ω)	Deviation between runs (Ω)
Heat Channel							
Sim.	0.046405	9.64E-07	0.164	0.025	606.090	0.027	0.015
1000	0.046422	1.65E-06	-0.270	0.022	1000.654	0.168	0.028
1200	0.046423	2.41E-06	-0.227	0.042	1200.484	0.321	0.075
Sense Channel							
Sim.	0.046291	3.99E-07	-0.001	0.018	1995.042	0.044	0.025
1000	0.046527	2.69E-06	-2.539	0.028	1003.656	0.215	0.045
1200	0.046530	3.95E-06	-2.546	0.046	1204.482	0.285	0.058

Although these tests confirm that resistance can be measured consistently, there are a few subtle issues which could be addressed with further experiments. The data show more noise when the probe cable is in use. Perhaps this is simply electrical noise due to the fact that the unshielded part of the cable may act as an antenna. Another possibility is that the addition of approximately 12 feet of wire somehow adversely affects the current source, causing some oscillation. However, given the repeatability displayed by these tests, these problems are not significant. Simple averaging of the 10 Hz data may completely eliminate the problem.

Another issue is the intercept of the calibration of the sense channel when the probe cable is attached. While the intercept for the heat channel is changed slightly when the probe cable is used, the sense channel intercept changes from about 0 Ω to approximately 2.5 Ω. This value is significantly above zero and may explain the discrepancy between the expected resistances of 1000 Ω and 1200 Ω and the measured values of 1003.7 Ω and 1204.5 Ω, respectively. The measured resistances on the heat channel are not exactly 1000 Ω and 1200 Ω, but the slight increase in resistance could easily be due to the resistance of the leads used in the measurement. However, lead resistance cannot explain the problems on the sense channel.

The current sources for the heat and sense channels are built identically, but the placement of the calibration resistors is different for each. For the heat channel, the current runs through the two calibration resistors before it goes to the thermistor. For the sense channel, in order to accommodate the safety circuit, the current passes through the low calibration resistor and the thermistor before reaching the high calibration resistor and returning to ground. This configuration may have a subtle effect on the calibration. Because the high resistance calibration resistor is connected directly to ground, any noise in the ground may contaminate the measurement of voltage across it. In addition, because the thermistor is in series between the two calibration resistors, the common mode voltages of the two calibration resistors are further apart for the sense channel than they are for the heat channel. Although the common mode rejection ratio of the A/D converter should be 92 dB, this difference may be contributing to the error.

7.2.3 Long-Term Stability Tests

Although the repeatability of measurements is the most important factor in determining the performance of the system, testing stability over time can also provide information about how the system behaves. The circuit was tested in sense mode using both the internal simulators and external fixed resistors connected with the probe cable. To facilitate the measurements, a modified version of the debugging routine was written. With this version of the software, the "d1" command leaves the sample rate at 10 Hz and simply sends the raw A/D converter counts to the host rather than converting these counts to voltages. The new software should have no effect on the quality of the measurements, but greatly simplifies data collection with standard terminal programs.

Figure 7-3 shows the measured resistance vs. time for the sense channel using the internal simulator. The measurement continued for a period of approximately 2 hours. For short intervals, the measured values vary by only a few bits at most. Although there is some drift over time, this fact just illustrates the need for periodic calibrations. A given sense measurement should last no longer than 20 to 30 minutes. Since the calibration procedure takes less than one second, this calibration should not interrupt measurements significantly.

The long-term tests performed using the external fixed resistors, shown in Figure 7-4, provide more insight into the added noise found when making external measurements. The data from the heat channel and the sense channel were taken simultaneously and are plotted on the same time scale. It is interesting to note the intervals of noisy data versus those of relatively quiet data. Despite the fact that the two resistors are driven by separate current sources and read by separate A/D converters, the noisy and quiet intervals occur at essentially the same time for each channel. This indicates that the noise is most likely due to electrical pickup through the probe cable. Further tests might indicate whether this electrical noise is affecting the A/D converters, the current sources, or both. Although it would be best to reduce the electrical noise, the repeatability tests show that averaging over short time intervals can greatly reduce the final effect on the measurement.

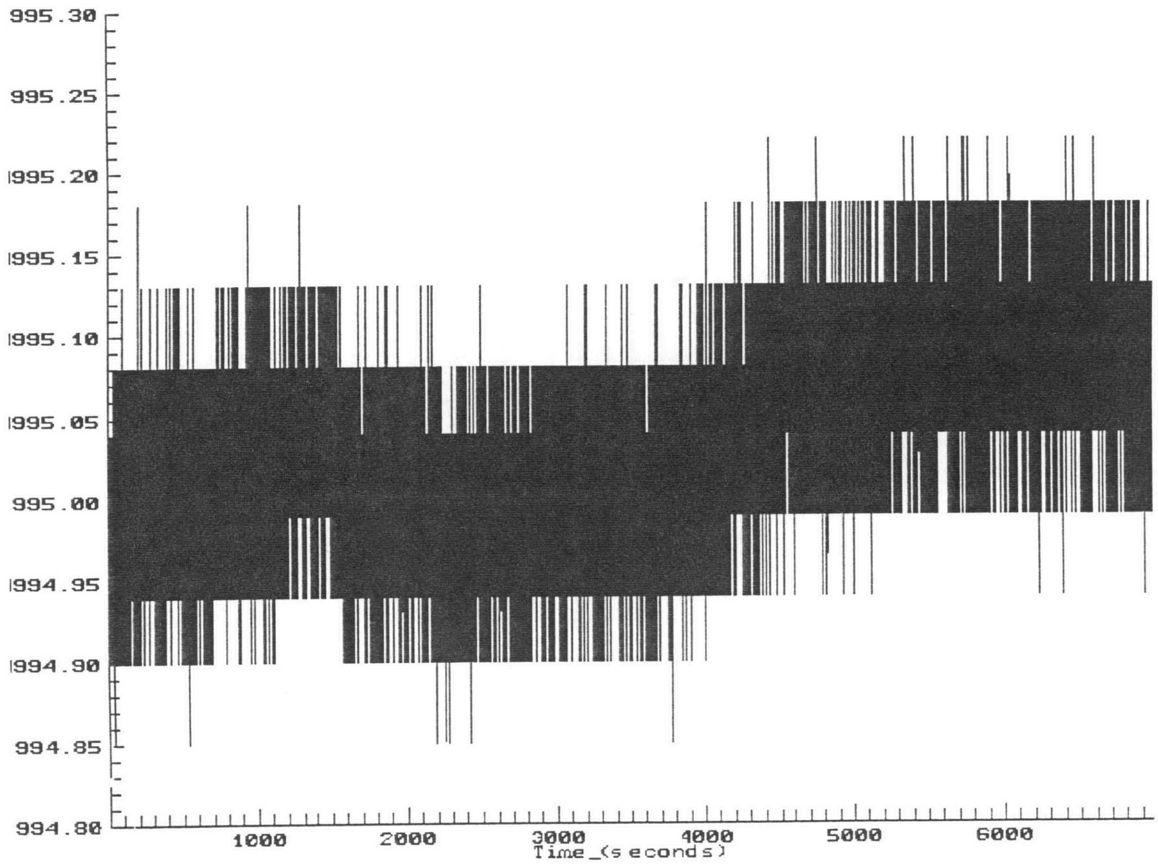


Figure 7-3: Measured Resistance (Ohms) vs. Time with Internal Simulator

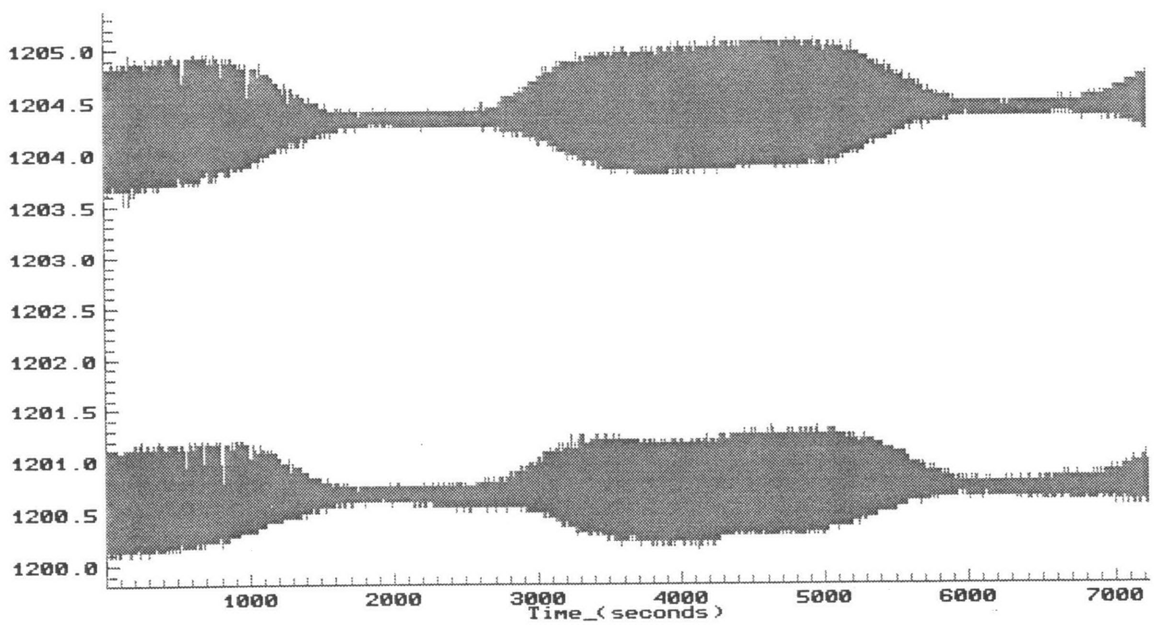


Figure 7-4: Measured Resistance (Ohms) vs. Time with External Resistors

7.3 Testing of Heat Mode

The heat mode of measurement is a little more complicated than the sense mode. First, the scaling network is calibrated. As for the sense mode, the software has been configured to collect three samples for each value measured and send these to the host. After calibration, the desired resistance is requested and the software calculates the appropriate code for the digital to analog converter. The code is entered, the output of the scaling network is double-checked, and the measurement begins.

The basic functionality tests of the circuit had already verified that each component of the control loop was working properly, so testing of the heat mode began with the "h0" command to start heating with the on-board thermistor simulator. The calibration values were very close to the expected values of 52428 for the voltage across the precision resistor, 3277 for the voltage out of the scaling network with the code set to the maximum value, and 19661 for the voltage out of the scaling network with the code set to zero.

Various set resistance values resulted in the appropriate code for the D/A converter, and lower resistance settings produced higher voltages as expected. There were some math overflow problems with the calculation of the "scale" or effective resistance of the scaling network after the proper code had been entered, but a minor software adjustment fixed this bug in the 32-bit math section of the code.

7.3.1 Repeatability Tests for Heat Mode

As with sense mode, the repeatability of the heat mode measurement is key. To determine the repeatability of the calibration and the behavior of the control loop, a series of measurements were taken with the on-board thermistor simulator. Ten measurements were completed at each of three resistance settings: 1000.0 Ω, 1500.0 Ω, and 2000.0 Ω. The measurements were then repeated using an external thermistor simulator at the end of the probe cable.

For each measurement the calibration data were compared using the calculated D/A converter code and a scale value. The software calculates the effective resistance of the scaling network using the last reading from the input to the network and the last reading from the output from the network. This measurement is to serve as a quick check for the user. The host PC software could certainly perform a more exact calculation using all three samples for each value. This improved calculation was used for the scale factors for the data shown below.

Table 7-4 shows the data from these tests. The scale and scale deviation values are given in Ohms. The voltage values indicate the voltage level at which the thermistor simulator settled. To calculate this level, the first five samples were ignored to allow for any settling, and then the next 10 samples were averaged. The next column, voltage deviation, indicates the average deviation over the 1 second window. The final columns, indicate the average power in mW and the deviation over the 10 runs expressed as a percentage of the mean value.

The data show excellent repeatability. Most strikingly, the code calculated for each set resistance is completely constant across all runs with a deviation of zero. The calibration is clearly stable to at least 12 bits, the limit of the D/A converter. The scale factor, while not exactly equal to the desired set resistance, is very consistent. The deviation between runs is less than the 0.625 Ω step size expected for the set resistance with the 12-bit D/A converter. The fact that the scale factor is consistently lower than the desired resistance may be due to a non-linearity in the D/A converter or in one of the opamps in the scaling network. Since the effective resistance is consistent between runs and is calculated after the code is set, the software on the host PC should be able to correct for this minor error.

The voltage value data are also very tight. In general, the deviation across the 1 second window was considerably less than 1 bit. The repeatability between runs is as good as could be expected with 16-bit resolution. As the behavior of the control loop is definitely repeatable, this instrument, along with a probe, can be calibrated for thermal conductivity. The difference between the voltage values for the internal and external thermistor simulators is not significant. The simulators are not built with identical components.

These repeatability tests also allow for evaluation of the calibration procedure. Clearly, three samples for each measurement are sufficient for the D/A converter code calculation which only needs to be accurate to 12 bits. However, the effective resistance calculation can be more accurate than that. Over the 240 calibration data sets collected (4 sets for each of 60 runs), only 11 ranged over two or more bits. Many were consistent to 16 bits, and most of the others had a range of just one bit. Further tests may indicate that more samples are required, but these tests seem to demonstrate that three samples are plenty. Although it would not be difficult to collect more samples, it would increase the time required for calibration.

Table 7-4: Repeatability Data for Heat Mode

Simulator	DAC Code	Scale (Ω)	Scale Deviation	Voltage (Volts)	Voltage Deviation	Power (mW)	Percent Variation
R=1000 Ω							
Internal	3421	998.26	0.04	2.446127	96.4E-06	5.99399	0.0061%
External	3421	998.27	0.04	3.479025	102E-06	12.12463	0.0051%
R=1500 Ω							
Internal	2603	1498.17	0.06	1.833092	75.5E-06	2.242894	0.0049%
External	2603	1498.04	0.04	2.145905	75.6E-06	3.073951	0.0039%
R=2000 Ω							
Internal	1785	1998.73	0.05	1.628668	77.7E-06	1.327126	0.0019%
External	1785	1998.56	0.03	1.800317	77.3E-06	1.621743	0.0015%

7.3.2 Long-Term Stability Tests

For the heat mode, long term tests were performed in a manner similar to those for the sense mode. Measurements were made continuously for approximately two hours with a set resistance of 1000 Ω with both internal and external thermistor simulators. The first five samples of each run were ignored to allow for settling of the control loop, and then the rest of the data were plotted as voltage (in mV) vs. time. Figure 7-5 shows the results with the internal thermistor simulator.

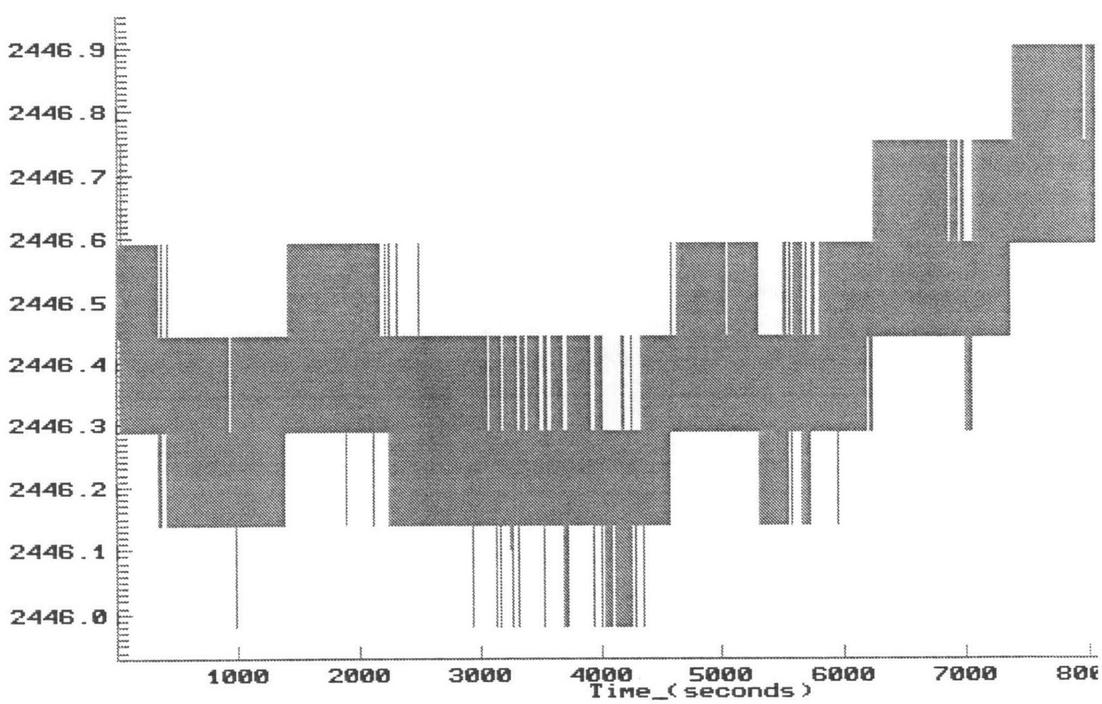


Figure 7-5: Voltage (mV) vs. Time for Internal Thermistor Simulator

Over short intervals, the data vary by only one bit, up or down. However, over the two hour measurement, there is some drift. This drift could be due to several factors including drift in the thermistor simulator itself or in one of the components of the scaling network. Any drift in the circuit can be compensated with more frequent calibration. In practice, it is desirable to restart thermal conductivity measurements at least every 15 minutes to recalculate the intrinsic thermal conductivity.

Figure 7-6 shows the results of the heat mode measurement with the external thermistor simulator. After approximately 1800 seconds, the data are very close to those with the internal simulator, generally varying only one bit, up or down. The initial 1800 seconds shows a steady increase. This measurement was taken right after the instrument was powered up. The increase may be due to some changes in the circuit as it warms up or due to drift in the thermistor simulator itself as it came to thermal equilibrium. Given this result, it is advisable either to allow the instrument to warm up before beginning measurements or to recalibrate frequently during early measurements.

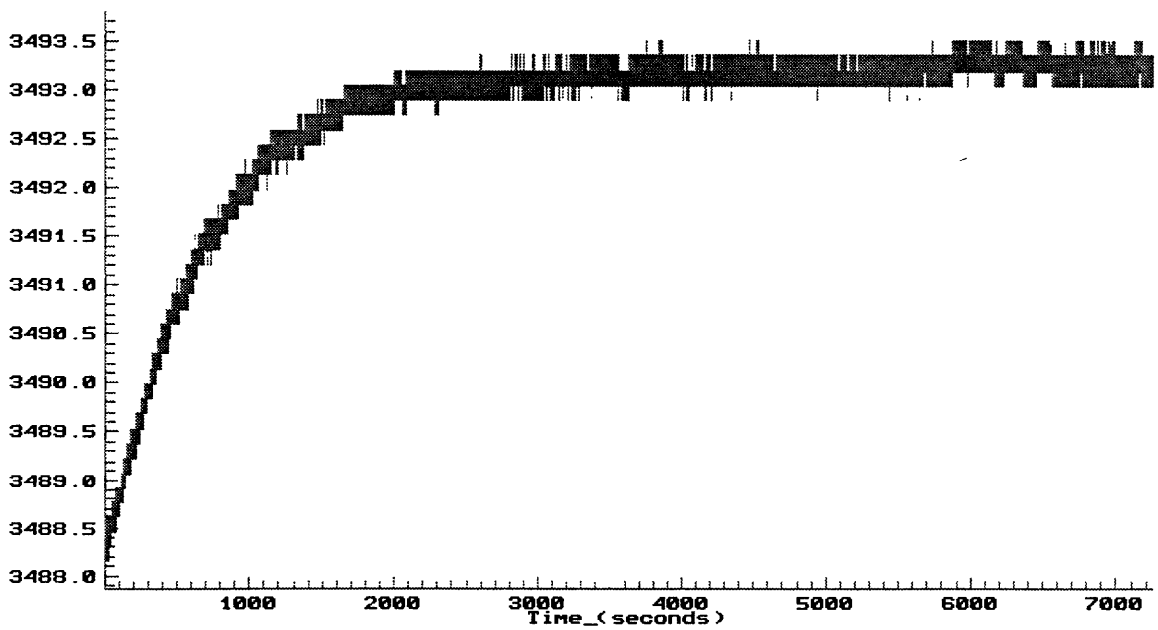


Figure 7-6: Voltage (mV) vs. Time for External Thermistor Simulator

7.4 Improvements to the Circuit

Although the previously described experiments showed that the instrument worked well in most instances, the electrical noise seen during the sense measurements with external resistors was cause for concern. In an effort to reduce the effect of high frequency noise on the circuit, two 0.1 μF capacitors were added to the circuit at the point where the single-probe connector connects the sense thermistor to the probe board. No capacitors were added to the heat thermistor circuit.

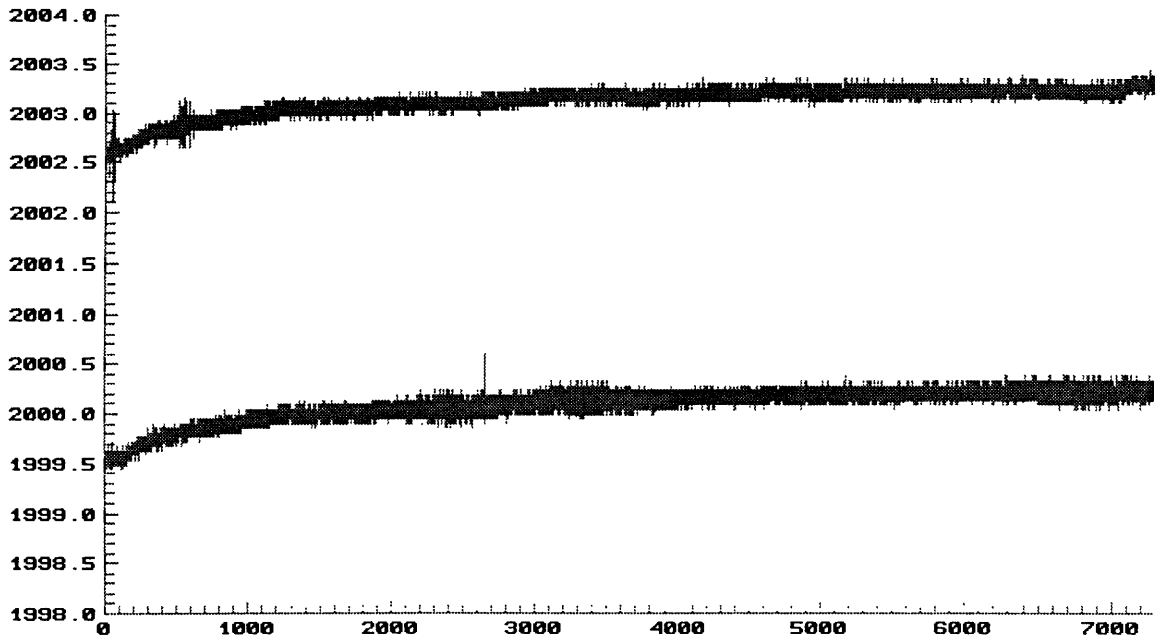


Figure 7-7: Resistance (Ohms) vs. Time for Improved Circuit

The long term external resistor measurement and the repeatability experiments for the sense mode were then repeated. The addition of these capacitors, greatly reduced the noise as can be seen in Figure 7-7. Although there is an initial rise, which is most likely due to warm up of the instrument, the data are significantly tighter than on the previous test without the capacitors. Table 7.5 shows the repeatability results for the improved circuit and the older data for comparison.

With the noise reduction capacitors in place, the deviation over the two second windows is generally less than 0.1 Ω and the repeatability between runs is better than 0.03 Ω. Assuming a thermistor with a resistance of 1 KΩ at 37 °C which varies 4%/ °C, the deviation over a two second window would be less than 2.5 mdeg C and the repeatability between runs would be good to at least 1 mdeg C. Calibration between temperature and measured resistance should be very accurate.

The precision of the temperature measurement is also important for the effective thermal conductivity and perfusion calculations which are based on the measurement of both power and temperature. Table 7-4 shows that power can be measured with a precision of approximately 0.006%. If a temperature step of 6 °C is used, the size of the temperature step can be measured with a precision of approximately 2.5 mdeg C/6 °C or 0.04%. The larger error in the temperature measurement dominates the measurement error for effective thermal conductivity and perfusion.

Table 7-5: Repeatability after Noise Reduction

Condition	Slope Mean - Ω/count	Slope Deviation Ω/count	Intercept Mean (Ω)	Intercept Deviation (Ω)	Resistance Mean (Ω)	Max. Deviation 2 s. (Ω)	Deviation between runs (Ω)
Heat Channel							
Sim.	0.046405	9.64E-07	0.164	0.025	606.090	0.027	0.015
1000-imp.	0.046408	1.03E-06	0.008	0.026	1000.723	0.058	0.015
1200-imp.	0.046408	8.26E-07	-0.061	0.033	1200.488	0.103	0.021
1000-old	0.046422	1.65E-06	-0.270	0.022	1000.654	0.168	0.028
1200-old	0.046423	2.41E-06	-0.227	0.042	1200.484	0.321	0.075
Sense Channel							
Sim.	0.046291	3.99E-07	-0.001	0.018	1995.042	0.044	0.025
1000-imp.	0.046835	9.79E-07	-5.712	0.027	1003.22	0.037	0.014
1200-imp.	0.046831	9.32E-07	-5.722	0.023	1203.939	0.070	0.016
1000-old	0.046527	2.69E-06	-2.539	0.028	1003.656	0.215	0.045
1200-old	0.046530	3.95E-06	-2.546	0.046	1204.482	0.285	0.058

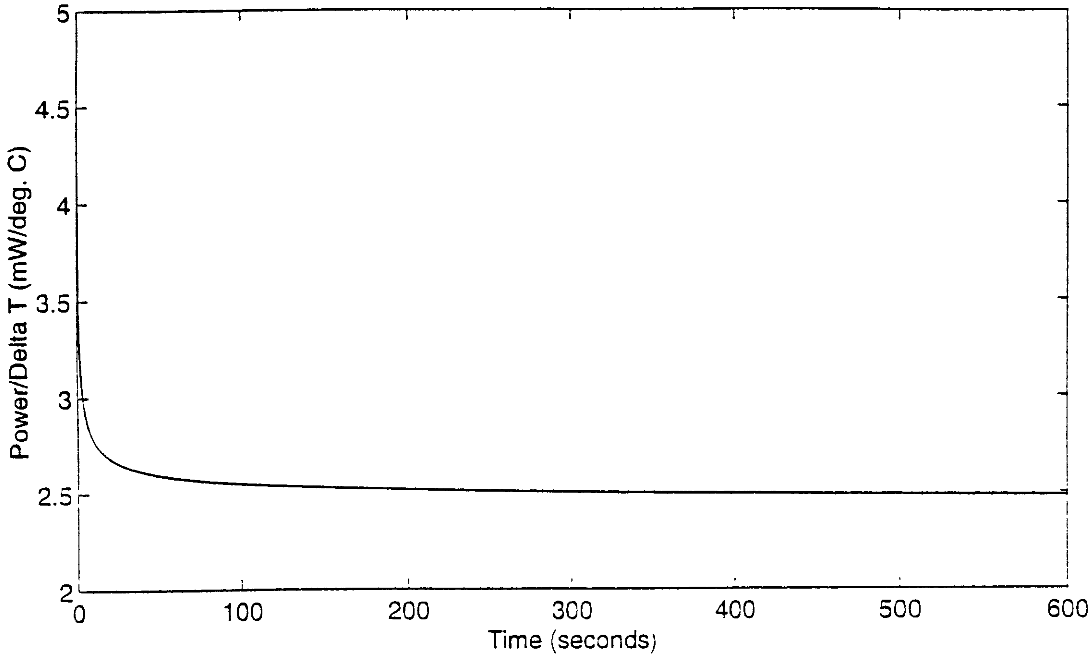
Further improvements to the circuit might reduce the variability of resistance measurements; however, there is another limit to the precision of measurement of the induced temperature step. Temperature is initially measured by both the heat thermistor and the sense thermistor, and the relationship between the two is tracked. When the difference between the two thermistors seems to be stable, heat measurements can be made. During the self-heating, the temperature of the sense thermistor is measured and used as an indication of the baseline temperature of the heat thermistor. If the temperature difference between the heat and sense thermistors is larger than 2.5 mdeg C, improved electronics will not truly improve the accuracy of the measurement.

7.5 Tests with Real Thermistors

Although much can be learned about the electronics with fixed resistors and thermistor simulators, the true test of the instrument is how well it performs with a real probe. To calibrate the probe for thermal conductivity, 10 minute heat mode measurements are made in two different media of known thermal conductivity. Figures 7-8 and 7-9 each show five measurements. The repeatability of the measurements is essential to accurate calibration of the probe.

Measurements were made in high thermal conductivity and low thermal conductivity media placed in a water bath with the temperature held at 37 °C. Before each heat mode measurement, the temperature of each sensor was measured for at least one minute. Using the resistance to temperature calibration previously determined with the TDP200, these resistance measurements were converted to temperature. After the

Figure 7-8: Five Measurements in High Conductivity Medium



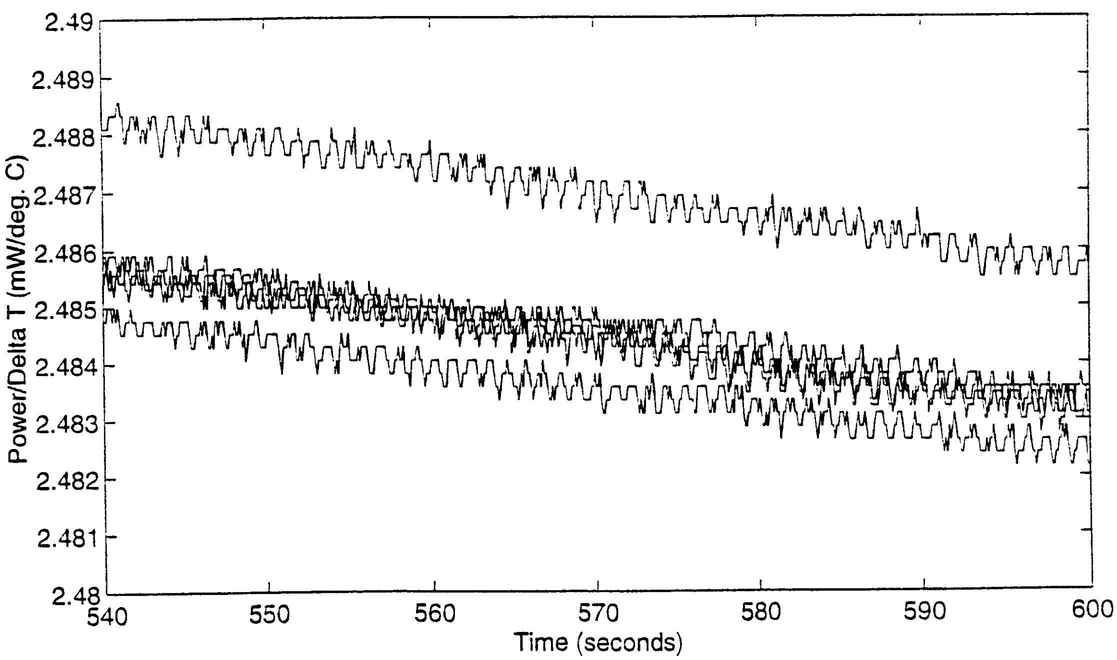
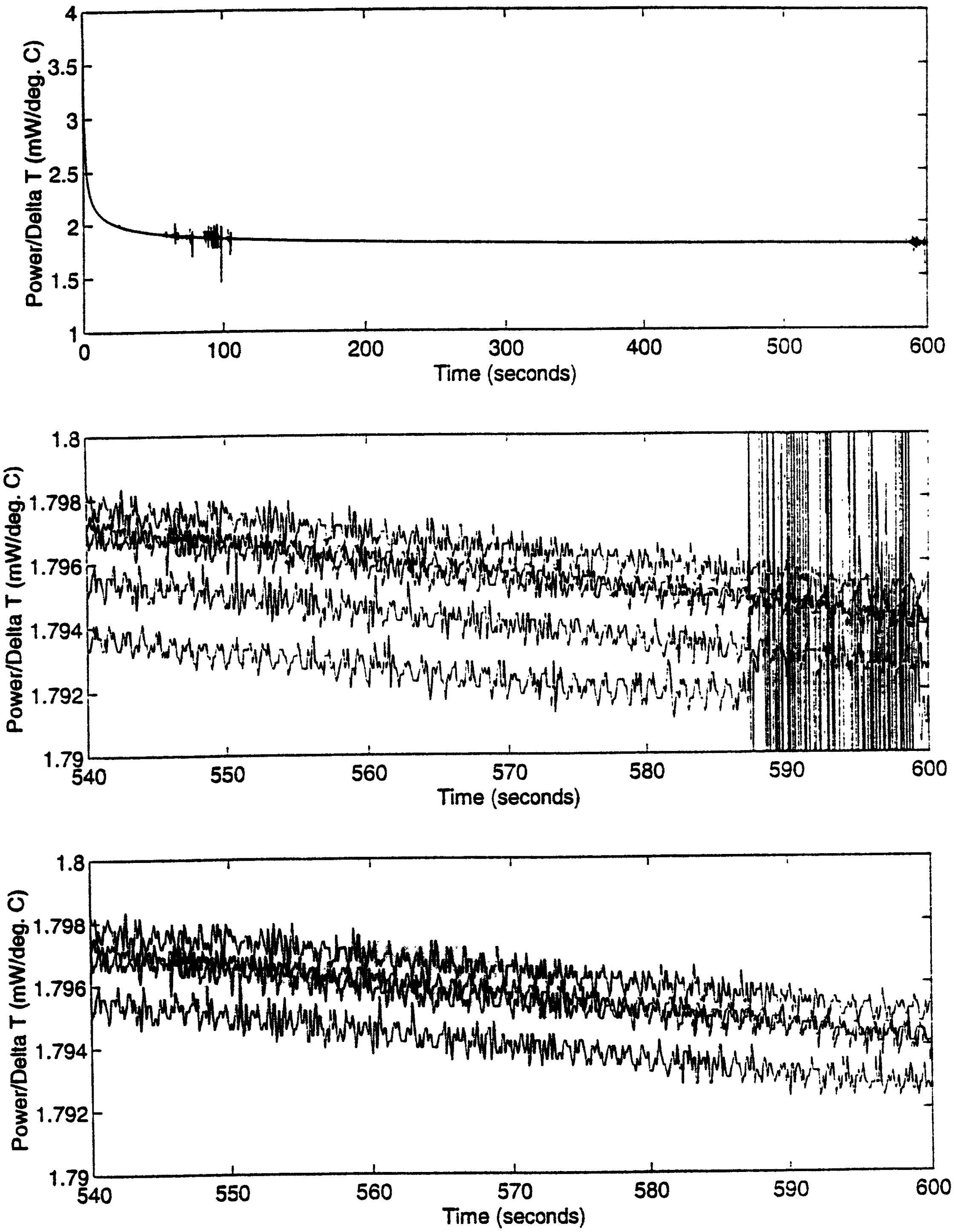


Figure 7-9: Five Measurements in Low Conductivity Medium



baseline temperature had been measured, a heat mode measurement was performed with a set resistance of 707.5 Ω, approximately 43 °C.

To analyze the data, the average temperature prior to the heat mode measurement and the temperature associated with the final scale factor determined in the heat mode calibration were calculated to determine the size of the temperature step. It was assumed that the baseline temperature of the medium was held constant by the fixed temperature water bath. Previous experiments have shown that the temperature is generally constant to at least 10 mdeg C. In these non-perfused media, the thermal field generated by the heat thermistor is very large and encompasses the sense thermistor. Therefore, the sense thermistor temperature cannot be used to track changes in the baseline temperature. After determining the size of the "Delta T," power was calculated using the voltage measured by the A/D converter and the known resistance, 707.5 Ω.

The data from the first measurements are presented in Figure 7-8. The top graph shows all 5 measurements over the 10 minute period. The initial transient has the expected shape. The lower graph shows more detail of the final minute of each measurement. Although the spread in the data is much more evident in this plot, at 600 seconds, the range of Power/Delta T measurements is only 3.1 μW/ °C. Compared to the mean value, this represents a spread of 0.125%. By comparison, a deviation of 10 mdeg C out of the expected 6 °C step would be 0.167%. The data are as repeatable as the measurement apparatus allow.

The data from the low conductivity medium are shown in Figure 7-9. The top graph shows five measurements. For one measurement, there are short bursts of noise in the data, but then the measurement quickly settles down to the correct value. These disturbances may indicate a problem with the A/D converter. Out of 21 such measurements, only two showed any evidence of this type of problem.

The two lower graphs of Figure 7-9 show the final minute in more detail. The bottom plot excludes the noisy data to allow better examination of the remaining measurements. Even including the noisy measurement, at 600 seconds, the range is only 2.6 μW/ °C, or 0.145% of the mean value. The repeatability of the measurements insures that probe calibration will not be a problem.

7.6 Future Work

Although the preliminary tests described in this chapter show that the system is working well, further work could be done to better quantify the behavior of the instrument and possibly improve the measurement capability.

In particular, the effect of the noise-reducing capacitors should be studied further. It is now unclear why these capacitors seem to increase the offset in the voltage to resistance calibration of the sense channel, but further experiments might illuminate this issue. Also, the number, placement, and size of the capacitors should be considered more carefully. Although it is clear that the addition of two capacitors greatly reduced the noise, this solution is most likely not the optimal solution. Perhaps better results could be achieved with three capacitors, or perhaps one capacitor would be sufficient.

Additional work could also be done to study the calibration process for both heat and sense modes. The software now collects three samples for each measurement for both calibration sequences. The precision of some of the calibration calculations might be improved with more samples. Future studies might show whether or not it is reasonable to take more samples and increase the time required for calibration.

The drift seen in the long-term measurement of the heat mode using the external thermistor simulator needs to be studied more carefully. Although the initial transient is most likely due to thermal fluctuations in one or more of the components, this drift deserves further investigation. The occasional noise during real probe measurements in the low conductivity medium may also require further experiments.

Finally, some changes to the software may be in order. Most importantly, an interrupt routine must be written for the safety circuit to shut down probe measurements when there is an indication of a problem. Some additional assembly language code is required. Beyond that, as the probe board is used and interfaced to the rest of the multi-channel perfusion measurement system, changes in the user interface may be desired. Fortunately, most of these types of changes could be made with the BASIC.

Appendix A: Complete Schematics

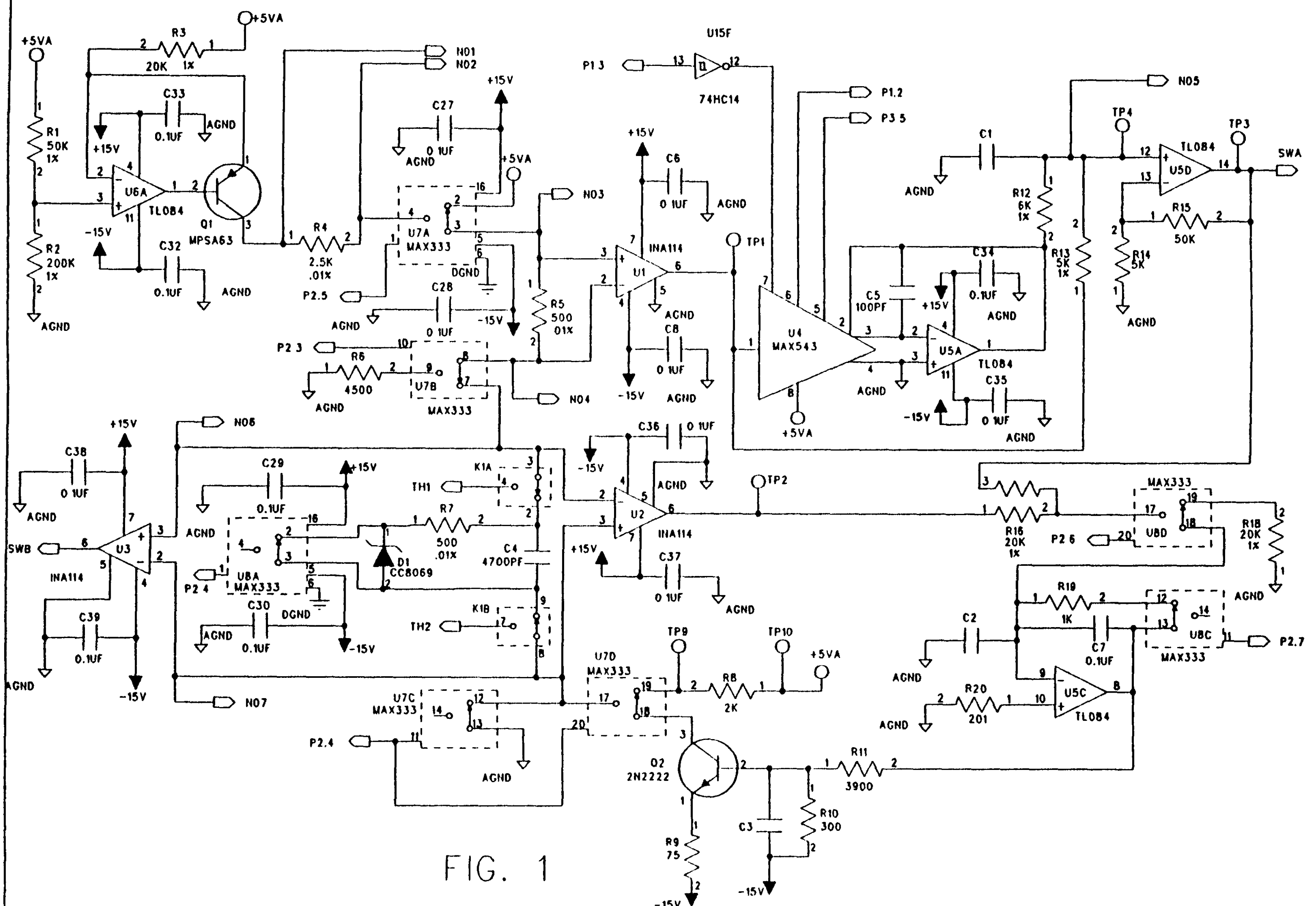
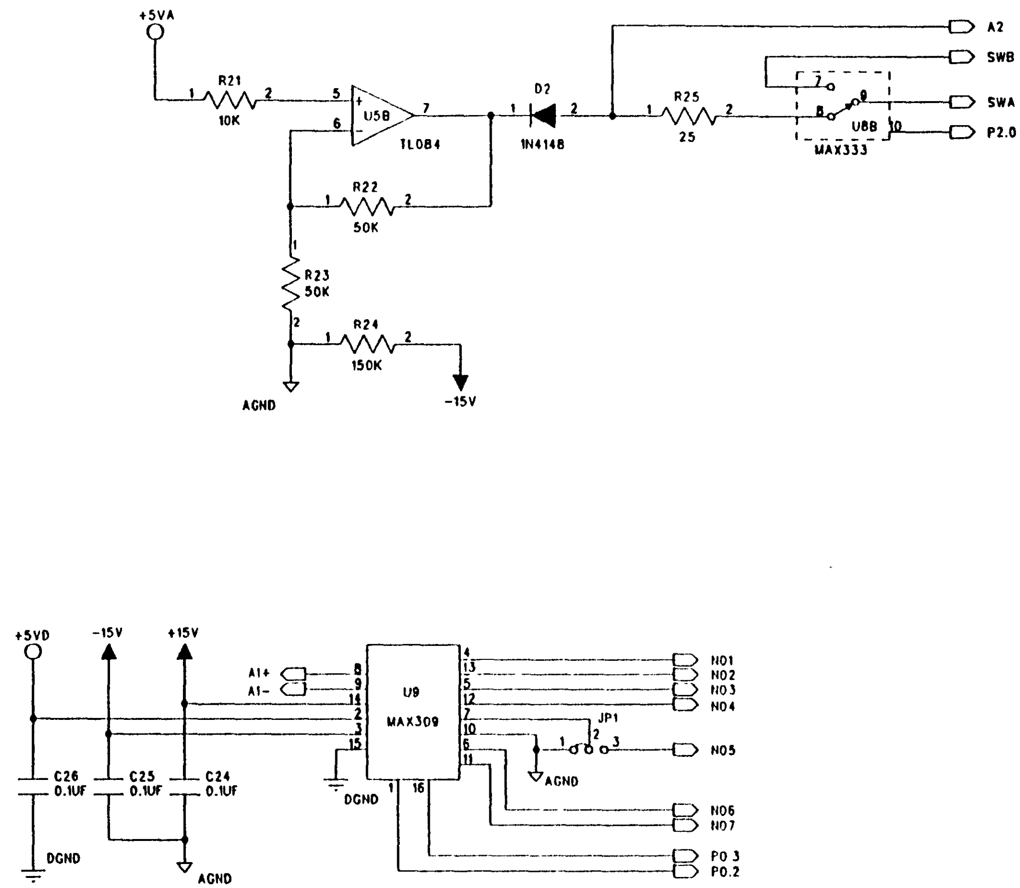
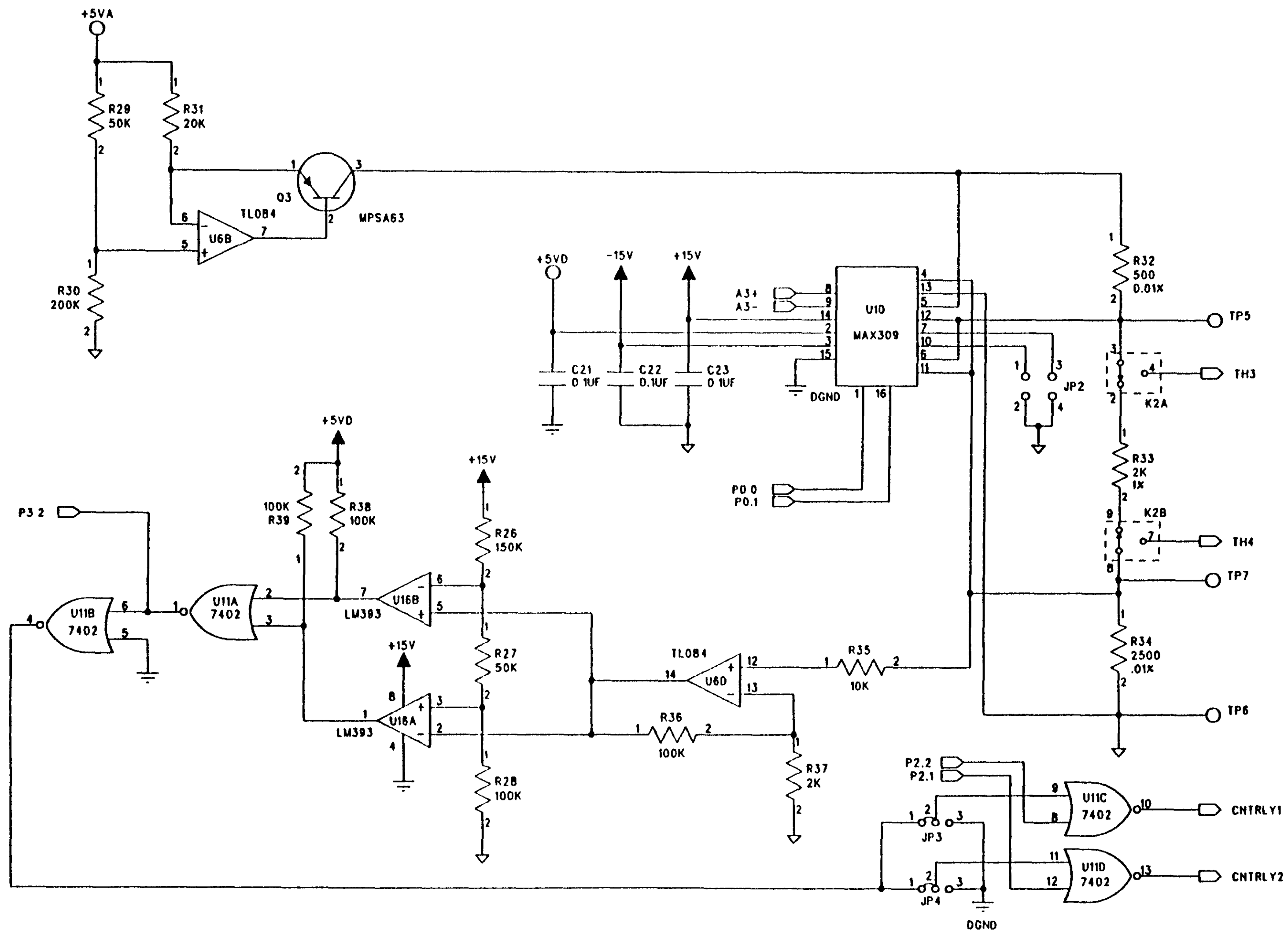


FIG. 1

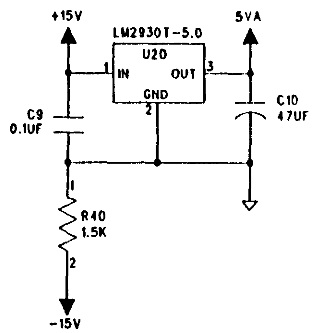
BME Center, MIT MULTI SITE PERFUSION PROFILOMETER	This drawing contains proprietary information which is the confidential property of BME Center, MIT and shall not be reproduced or disclosed to others.	Heat Thermistor Control Circuit	DWN BY: DATE: 10-16-97 PAGE: 1 of 10 REV. 1.0



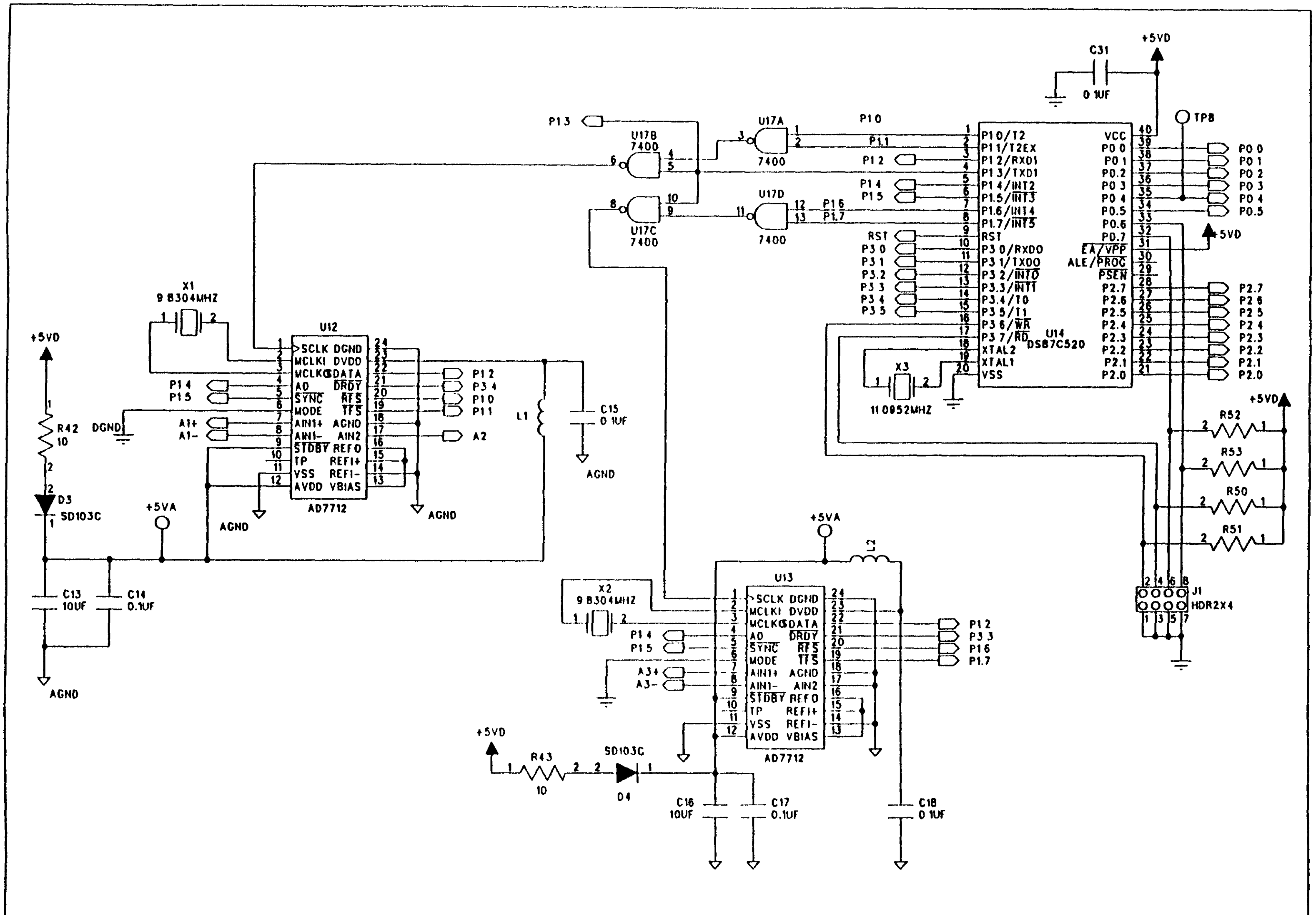
BME Center, MIT MULTI SITE PERFUSION PROFILOMETER	This drawing contains proprietary information which is the confidential property of BME Center, MIT and shall not be reproduced or disclosed to others.	Input to Single Channel of A/D 1 Multiplexer for A/D 1	DWN BY: DATE: 10-16-97 PAGE: 2 of 10 REV. 1.0



BME Center, MIT MULTI SITE PERFUSION PROFILOMETER	This drawing contains proprietary information which is the confidential property of BME Center, MIT and shall not be reproduced or disclosed to others.	Sense Thermistor and Safety Circuit	DWN BY: DATE: 10-16-97 PAGE: 3 of 10 REV. 1.0



BME Center, MIT MULTI SITE PERFUSION PROFILOMETER	This drawing contains proprietary information which is the confidential property of BME Center, MIT and shall not be reproduced or disclosed to others.	+5VD and +5vA Voltage Regulators	DWN BY: DATE. 10-16-97 PAGE: 4 of 10 REV. 1.0



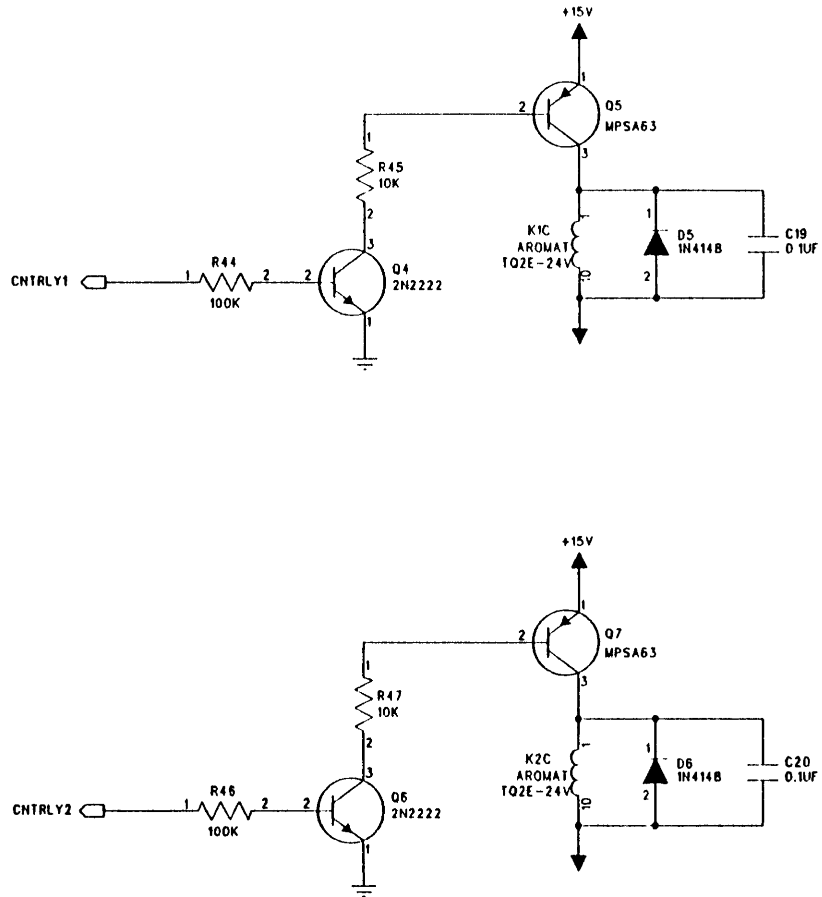
BME Center, MIT

MULTI SITE PERFUSION PROFILOMETER

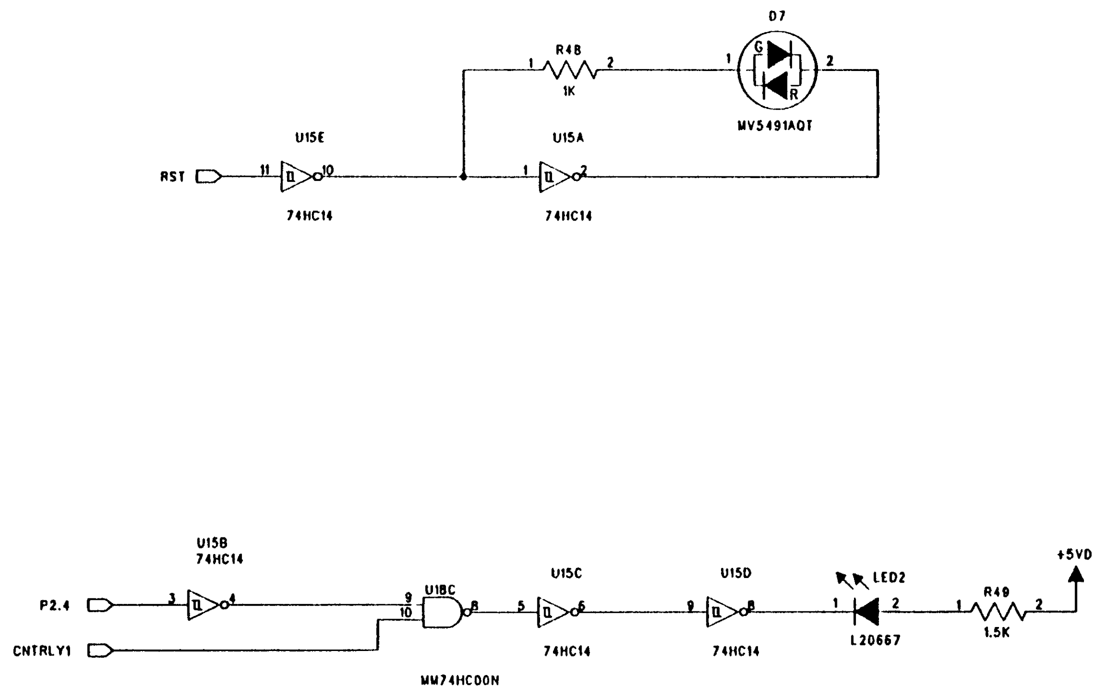
This drawing contains proprietary information which is the confidential property of BME Center, MIT and shall not be reproduced or disclosed to others.

Microprocessor and Digital Circuitry

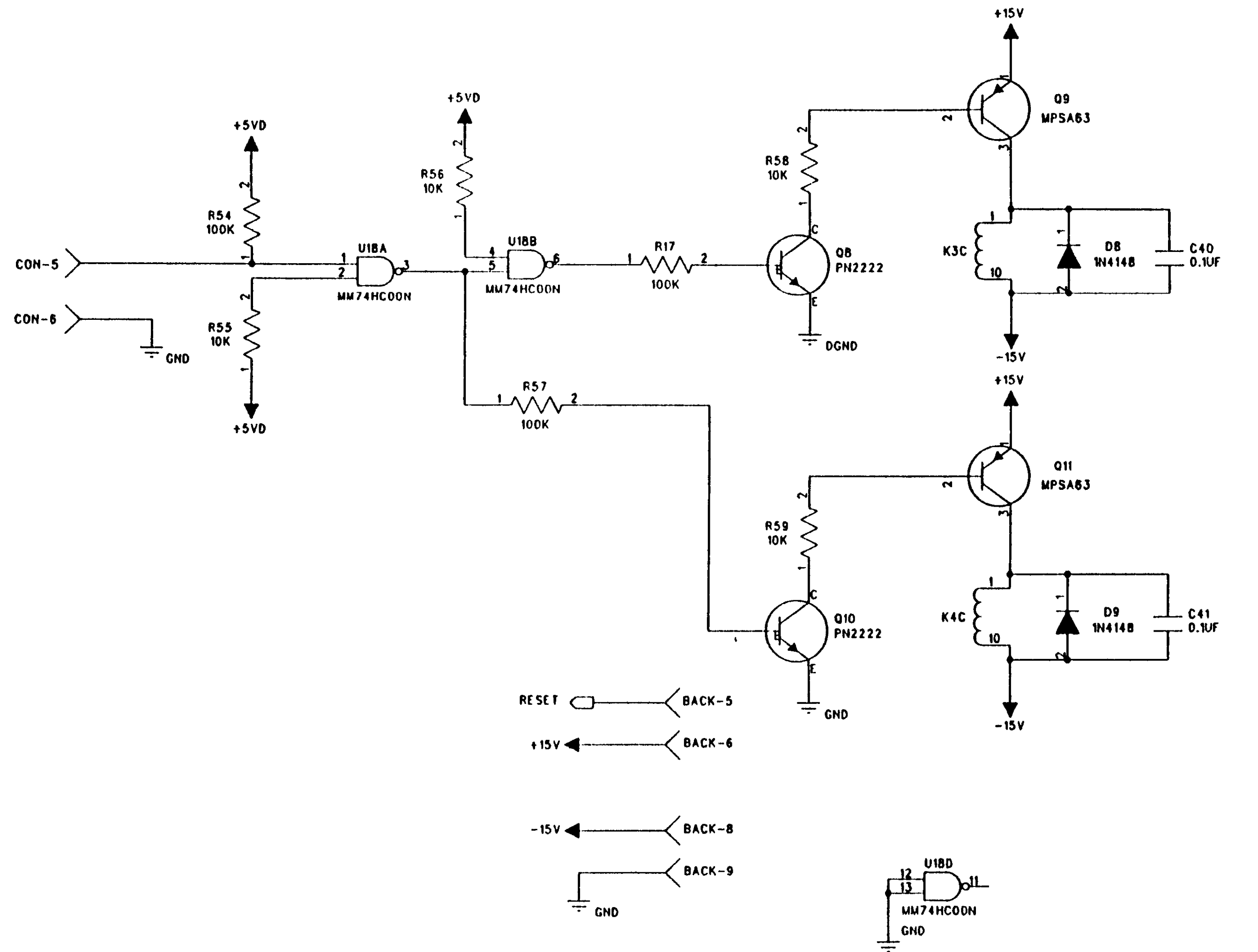
DWN BY:
DATE: 10-16-97
PAGE: 5 of 10
REV. 1.0



BME Center, MIT MULTI SITE PERFUSION PROFILOMETER	This drawing contains proprietary information which is the confidential property of BME Center, MIT and shall not be reproduced or disclosed to others	Thermistor / Simulator Relays	DWN BY: DATE: 10-16-97 PAGE: 6 of 10 REV. 1.0



BME Center, MIT MULTI SITE PERFUSION PROFILOMETER	This drawing contains proprietary information which is the confidential property of BME Center, MIT and shall not be reproduced or disclosed to others.	Reset and Self-Heating Indicator	DWN BY: DATE: 10-16-97 PAGE: 7 of 10 REV. 1.0



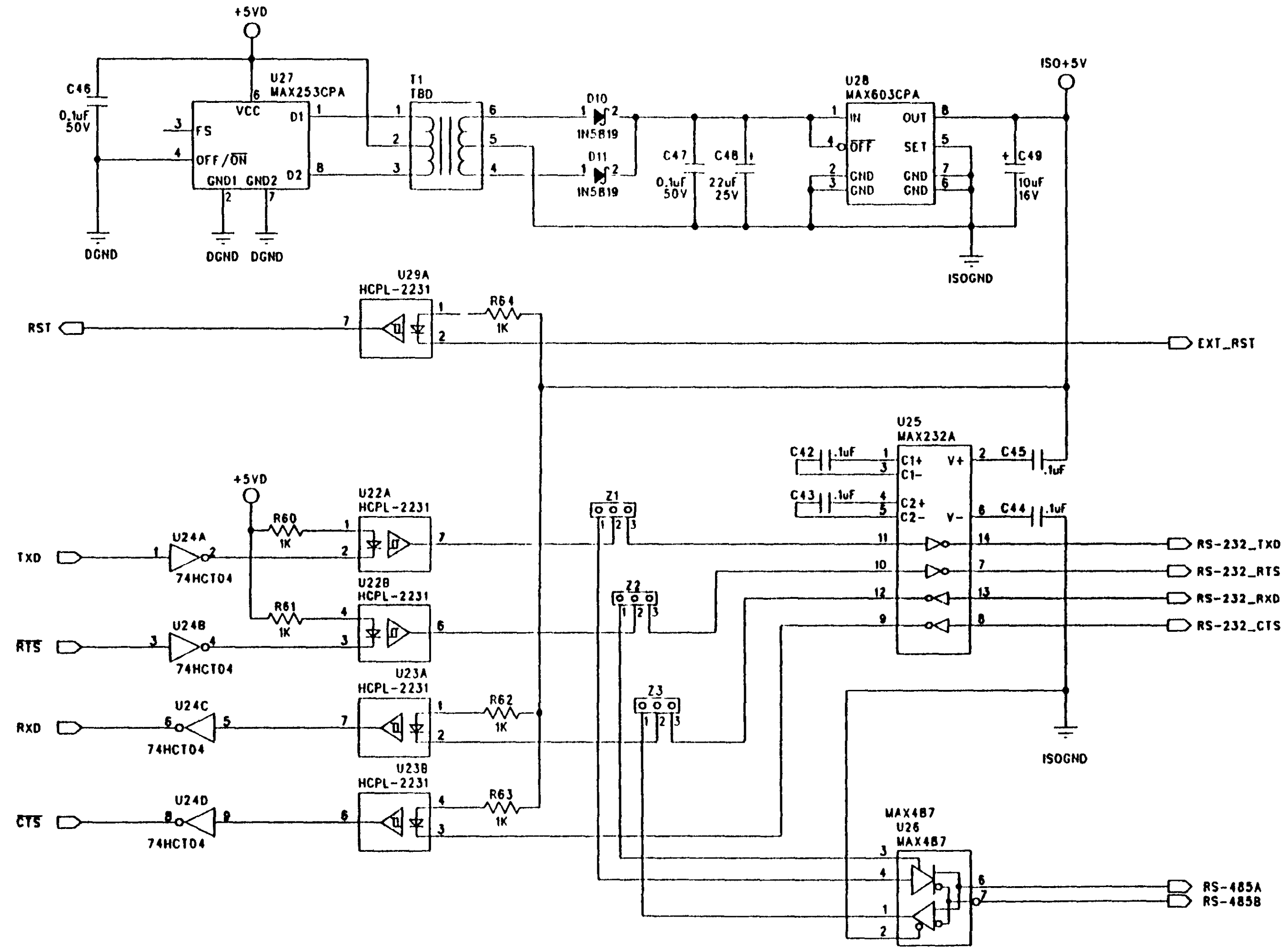
BME Center, MIT

MULTI SITE PERFUSION PROFILOMETER

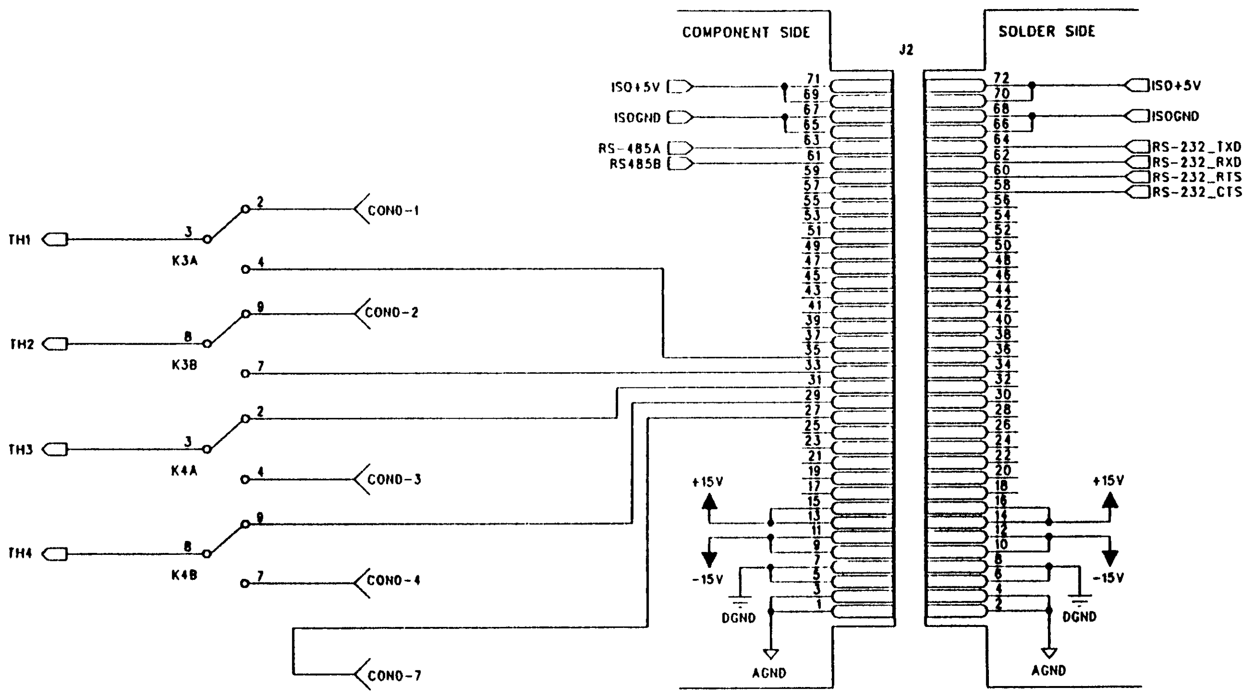
This drawing contains proprietary information which is the confidential property of BME Center, MIT and shall not be reproduced or disclosed to others.

Single / Multi-site Probe Relays

DWN BY:
DATE: 10-16-97
PAGE: 8 of 10
REV. 1.0



BME Center, MIT MULTI SITE PERFUSION PROFILOMETER	This drawing contains proprietary information which is the confidential property of BME Center, MIT and shall not be reproduced or disclosed to others.	Serial Communications	DWN BY: DATE. 10-16-97 PAGE: 9 of 10 REV. 1.0



BME Center, MIT MULTI SITE PERFUSION PROFILOMETER	This drawing contains proprietary information which is the confidential property of BME Center, MIT and shall not be reproduced or disclosed to others	Backplane and Probe Connectors	DWN BY: DATE: 10-16-97 PAGE: 10 of 10 REV. 1.0

Appendix B: Parts List

Position	Part Number	Manufacturer	Package Type
U1	INA114	Burr Brown	8-pin DIP
U2	INA114	Burr Brown	8-pin DIP
U3	INA114	Burr Brown	8-pin DIP
U4	MAX543	Maxim	8-pin DIP
U5	TL084	Texas Instruments	14-pin DIP
U6	TL084	Texas Instruments	14-pin DIP
U7	MAX333	Maxim	20-pin DIP
U8	MAX333	Maxim	20-pin DIP
U9	MAX309	Maxim	16-pin DIP
U10	MAX309	Maxim	16-pin DIP
U11	74HC02	Fairchild	14-pin DIP
U12	AD7712	Analog Devices	24-pin DIP
U13	AD7712	Analog Devices	24-pin DIP
U14	DS87C520	Dallas Semiconductor	40-pin DIP
U15	74HC14	Fairchild	14-pin DIP
U16	LM393	National Semiconductor	8-pin DIP
U17	74HC00	Fairchild	14-pin DIP
U18	74HC00	Fairchild	14-pin DIP
U20	LM2930T-5.0	National Semiconductor	TO-220
U21	LM2930T-5.0	National Semiconductor	TO-220
U22	HCPL-2231	Hewlett Packard	8-pin DIP
U23	HCPL-2231	Hewlett Packard	8-pin DIP
U24	74HCT04	Fairchild	14-pin DIP
U25	MAX232A	Maxim	16-pin DIP
U26	MAX487	Maxim	8-pin DIP
U27	MAX253CPA	Maxim	8-pin DIP
U28	MAX603CPA	Maxim	8-pin DIP
U29	HCPL-2231	Hewlett Packard	8-pin DIP
X1	9.8304 M	CTS	HC49
X2	9.8304 M	CTS	HC49
X3	11.0592 M	CTS	HC49
K1	TQ2E	Aromat	see drawing for 2 Form C
K2	TQ2E	Aromat	see drawing for 2 Form C
K3	TQ2E	Aromat	see drawing for 2 Form C
K4	TQ2E	Aromat	see drawing for 2 Form C
Q1	MPSA63		TO-92
Q2	2N2222		T0-18
Q3	MPSA63		TO-92

Q4	2N2222		T0-18
Q5	MPSA63		TO-92
Q6	2N2222		T0-18
Q7	MPSA63		TO-92
Q8	2N2222		T0-18
Q9	MPSA63		TO-92
Q10	2N2222		T0-18
Q11	MPSA63		TO-92
D1	CC8069	Harris	TO-92
D2	1N4148		DO-35
D4	SD103C		SOD-123
D5	1N4148		DO-35
D6	1N4148		DO-35
D7	MV5491RQT		T-1 3/4
D8	1N4148		DO-35
D9	1N4148		DO-35
D10	1N5819		
D11	1N5819		
LED2	L20667		T-1 3/4
L1			
L2			
T1	67117970	Schott	
R1	50K		1/4 W, 1%
R2	200K		1/4 W, 1%
R3	20K		1/4 W, 1%
R4	2.5K 0.01%	Vishay	Vishay S102K
R5	500O 0.01%	Vishay	Vishay S102 K
R6	4.5K		1/4 W, 1%
R7	500O 0.01%	Vishay	Vishay S102K
R8	2K		1/4 W, 1%
R9	75 Ohm		1/4 W, 1%
R10	300 Ohm		1/4 W, 1%
R11	3.9K		1/4 W, 1%
R12	6K		1/4 W, 1%
R13	5K		1/4 W, 1%
R14	5K		1/4 W, 1%
R15	50K		1/4 W, 1%
R16	20K - SP		Vishay 300144

R17	100K		1/4 W, 1%
R18	20K		1/4 W, 1%
R19	1K		1/4 W, 1%
R20	20K		1/4 W, 1%
R21	10K		1/4 W, 1%
R22	50K		1/4 W, 1%
R23	50K		1/4 W, 1%
R24	150K		1/4 W, 1%
R25	25 Ohm		1/4 W, 1%
R26	150K		1/4 W, 1%
R27	50K		1/4 W, 1%
R28	50K		1/4 W, 1%
R29	50K		1/4 W, 1%
R30	200K		1/4 W, 1%
R31	20K		1/4 W, 1%
R32	500 Ohm 0.01%	Vishay	Vishay S102K
R33	2K		1/4 W, 1%
R34	2.5K 0.01%	Vishay	Vishay S102K
R35	10K		1/4 W, 1%
R36	100K		1/4 W, 1%
R37	2K		1/4 W, 1%
R38	100K		1/4 W, 1%
R39	100K		1/4 W, 1%
R40	1.5K		1/4 W, 1%
R41	1.5K		1/4 W, 1%
R43	10 Ohm		1/4 W, 1%
R44	100K		1/4 W, 1%
R45	10K		1/4 W, 1%
R46	100K		1/4 W, 1%
R47	10K		1/4 W, 1%
R48	1K		1/4 W, 1%
R49	1.5K		1/4 W, 1%
R50	100K		1/4 W, 1%
R51	100K		1/4 W, 1%
R52	100K		1/4 W, 1%
R53	100K		1/4 W, 1%
R54	100K		1/4 W, 1%
R55	10K		1/4 W, 1%
R56	10K		1/4 W, 1%
R57	100K		1/4 W, 1%
R58	10K		1/4 W, 1%
R59	10K		1/4 W, 1%
R60	1K		1/4 W, 1%

R61	1K		1/4 W, 1%
R62	1K		1/4 W, 1%
R63	1K		1/4 W, 1%
R64	1K		1/4 W, 1%
C1	?		cer, axial
C2	?		cer, axial
C3	?		cer, axial
C4	4700P		cer, axial
C5	100P		cer, axial
C6	0.1U		cer, axial
C7	0.1U		cer, axial
C8	0.1U		cer, axial
C9	0.1U		cer, axial
C10	47U		elect., radial
C11	0.1U		cer, axial
C12	47U		elect., radial
C13	10U		tant., radial
C14	0.1U		cer, axial
C15	0.1U		cer, axial
C16	10U		tant., radial
C17	0.1U		cer, axial
C18	0.1U		cer, axial
C19	0.1U		cer, axial
C20	0.1U		cer, axial
C21	0.1U		cer, axial
C22	0.1U		cer, axial
C23	0.1U		cer, axial
C24	0.1U		cer, axial
C25	0.1U		cer, axial
C26	0.1U		cer, axial
C27	0.1U		cer, axial
C28	0.1U		cer, axial
C29	0.1U		cer, axial
C30	0.1U		cer, axial
C31	0.1U		cer, axial
C32	0.1U		cer, axial
C33	0.1U		cer, axial
C34	0.1U		cer, axial
C35	0.1U		cer, axial
C36	0.1U		cer, axial
C37	0.1U		cer, axial
C38	0.1U		cer, axial

C39	0.1U		cer, axial
C40	0.1U		cer, axial
C41	0.1U		cer, axial
C42	0.1U		cer, axial
C43	0.1U		cer, axial
C44	0.1U		cer, axial
C45	0.1U		cer, axial
C46	0.1U, 50 V		cer, axial
C47	0.1U, 50 V		cer, axial
C48	22U, 25 V		elect., radial
C49	10U, 16V		tant., radial

Appendix C: Source Code

```
;Dec. 10, 1997
;PERF_1_0.BAS

;General Idea:
;Do a little setup in very beginning
;Set up an input that will send to various subroutines
;Leave interrupt for data collection
;Can service interrupts while waiting for input
;Have circular buffer for data
        ;Data collection routines write to it
        ;Print routine reads from it
;Watchdog timer set to ~6.07 sec.
        ;looks for input from host - be sure that host is alive
        ;and that program is still looking for inputs

;POSSIBLE INPUTS
;s0     sense mode - sim (sense channel), sim (heat channel)
;s1     sense mode - therm, sim
;s2     sense mode - sim, therm
;s3     sense mode - therm, therm

;h0     heat mode - sim, sim
;h1     heat mode - therm, sim
;h2     heat mode - sim, therm
;h3     heat mode - therm, therm
;h8     calibrate heat mode

;p0     print out data
;p2     print out test line to check communications

;q0     stop data - set to cal mode
;q2     stop data - leave in present state - turn off watchdog
;q4     stop data - set to test mode - watchdog off, set DAC

;d0     turn debugging mode off
;d1     turn debugging mode on
;************************************************************
;SWITCH DESCRIPTIONS
;Switch A - from bottom of Ro to 4500 Ohm cal resistor
;               or to thermistor
;Switch B - relays to thermistor or thermistor simulator
;Switch C - from bottom of thermistor to AGND or
;               to current source for heating
;Switch D - from top of Ro to 5 V or to sense current source
```

```
;Switch E - input to integrator - 0 or output of voltage divider
;Switch F - feedback resistor on integrator
;Switch H - input to single ended channel of A/D
;               scaling network or thermistor voltage
;*************************************************************

;START OF PROGRAM

;*************************************************************

#target 80C52, CMOS
#xtal  11059200

#console timer2, mode=19200          ;19200 baud
#check math on
#print complete off
#control c enable
#istack start 50H        ;necessary for 32-bit math
#error message off       ;just handle with ONERR

unsigned integer T1, M1, XA, XB, TOUT, MOUT, RES
unsigned char CONA2, CONA1, CONA0, CONB2, CONB1, CONB0
unsigned char CRDA2, CRDA1, CRDA0, CRDB2, CRDB1, CRDB0
unsigned char DTA1, DTA0, DTB1, DTB0, DAC0, DAC1, HC
string INP$(2)
unsigned integer DLOW
unsigned char DFIRST, DLAST, DMAX, DUM, DTEMP
unsigned char CHCK, INP2, COUNT, DBUG, BNUM, SAD1, SAD2, CALNUM
unsigned char THI, TLO, MHI, MLO, CAL1, CAL2
unsigned integer DCHI, DCLO, DCIN, DCDIF, DCCODE
unsigned integer TEMPIN, TEMP1, TEMP2, TEMP3, TEMP4, TEMPOUT
unsigned integer SCALE, DUM2, V1, V2, SNUM, BCONV

;**************************************************************
;**************************************************************

main:

#ASM
DS87C520    EQU   1          ;use Port 2 and strings
        orl    pmr, #003h    ;set up internal SRAM
        mov    scon, #050h   ;initialize ser port 0
; Reset watchdog
        mov    ta, #0aah     ;timed access protection stuff
        mov    ta, #055h
```

```
        clr    ewt              ;disable reset on timeout

        orl    ckcon, #0c0h   ;set watchdog for 2^26 clock cycles
        clr    wtrf             ;clear reset flag

        mov    a,p0             ;grab value at Port 0 for board ID#
        mov    dptr,#__DTA1
        movx   @dptr,a                 ;save value

        mov    a,p3                    ;grab value at Port 3 for board ID#
        mov    dptr,#__DTA0
        movx   @dptr,a                 ;save value
#ASM_END

10 PRINT DTOP,SPC(1),CTOP     ;so I know how much memory used
ONERR 900

SNUM=(CBY(16383)*256)+CBY(16382)  ;get serial number of firmware
BNUM=((DTA0/64)*4)+(DTA1/64)      ;calculate board # based on jumpers
BCONV=256: HC=0                   ;constant used to convert bytes

DLOW=400: DMAX=60     ;set up param's for circular buffer
DFIRST=0: DLAST=0     ;initialize circular buffer
DBUG=0: CALNUM=3      ;initially set debug mode off, get 3 cal readings

20 GOSUB 9000         ;initialize serial port for Mode 0 communications

;probably makes sense to start A/D at some known point
;can always adjust this later with a subroutine

CONA0=0B0H: CONA1=11H: CONA2=40H; 10 Hz with cal
CONB0=0B0H: CONB1=11H: CONB2=40H; 10 Hz with cal
GOSUB 2100       ;send control info to A/D1 and A/D2
SAD1=64 :SAD2=64

PRINT SNUM

;wait for an input to continue program
100 INPUT DTA1
DTA1=0:DTA0=0:DTB1=0:DTB0=0

CLOCK0 :TIME=0 :MSEC=0
CLOCK1                ;start the real-time clock

;Set up watchdog timer
```

```
#ASM
        mov    ta, #0aah          ;timed access protection stuff
        mov    ta, #055h
        setb   rwt                ;restart watchdog timer

        mov    ta, #0aah          ;timed access protection stuff
        mov    ta, #055h
        setb   ewt                ;enable reset on timeout
#ASM_END

ONEX1 3100                        ;read from A/D1 and A/D2

;*************************************************************

;INITIALIZATION COMPLETE - WAIT FOR INSTRUCTIONS

;*************************************************************
200 INPUT INP$
; will wait until it gets an input
; meanwhile will be servicing the interrupts from A/Ds

;on input, reset watchdog
#ASM
        mov    ta, #0aah          ;timed access protection stuff
        mov    ta, #055h
        setb   rwt                ;reset watchdog timer
#ASM_END

INP2=ASC(INP$(1))

; check the value of the input against the set values
IF ASC(INP$(0))=ASC(s) THEN GOTO 300      ;start sense mode
IF ASC(INP$(0))=ASC(h) THEN GOTO 400      ;start heat mode
IF ASC(INP$(0))=ASC(p) THEN GOTO 500      ;print out data
IF ASC(INP$(0))=ASC(q) THEN GOTO 600      ;stop measurements
IF ASC(INP$(0))=ASC(d) THEN GOTO 700      ;debug mode control
250 PRINT "Input Err" : GOTO 200

END

#include "MATH32.INC"

;*************************************************************

;SUBROUTINES TO RESPOND TO INPUTS
```

```
;*************************************************************
300 ; Sense Mode Subroutine
;First set ports for sense mode with simulators
;p2.0 1 - switch H
;p2.1 1 - sense bead relay
;p2.2 1 - heat bead relay
;p2.3 0 - switch A
;p2.4 1 - switch C and G
;p2.5 0 - switch D
;p2.6 1 - switch E
;p2.7 1 - switch F

#ASM
        mov    p2,#0d7h
#ASM_END

GOSUB 350                   ;check second input, set simulators or thermistors
IF INP2=99 THEN GOTO 250    ;if error wait for new input

GOSUB 4000                  ;do sense mode calibration

;set up for long term measurement - muxes to thermistor position
;p0.3 1 - heat mux A1
;p0.2 0 - heat mux A0
;p0.1 1 - sense mux A1
;p0.0 0 - sense mux A0

#ASM
        mov    p0,#0cah
#ASM_END

ONEX1 3100             ;clear any old data
IDLE
GOSUB 3600             ;timestamp for start of measurement
ONEX1 3300             ;set up a new ONEX to write data to buffer
GOTO 200               ;return for another input

;*************************************************************
;Subroutine to check 2nd input
;Sets each relay for either thermistor or simulator
350 IF INP2=ASC(0) THEN GOTO 390
IF INP2=ASC(8) THEN GOTO 390 ;calibration mode for heat side
IF INP2=ASC(1) THEN GOTO 360
IF INP2=ASC(2) THEN GOTO 370
```

```
IF INP2=ASC(3) THEN GOTO 380
INP2=99 :GOTO 390

360      ;p2.1 0 - sense channel thermistor
         ;p2.2 1 - heat channel simulator
#ASM
         clr    0a1h
#ASM_END
GOTO 390

370 ;p2.1 1 - sense channel simulator
   ;p2.2 0 - heat channel thermistor
#ASM
         clr    0a2h
#ASM_END
GOTO 390

380 ;p2.1 0 - sense channel thermistor
   ;p2.2 0 - heat channel thermistor
#ASM
         clr    0a2h
         clr    0a1h
#ASM_END

390 RETURN
;******************************************************************
;NOTE: The system must perform a scaling network calibration (heat cal)
;before it can start a heating measurement.  HC is the flag that detects
;whether or not this has been done.
;If a heat mode is invoked and no calibration has been done, this
;subroutine automatically performs a calibration.  If HC=1, the
;calibration is skipped and it just sets up the DAC for the resired
;effective resistance and checks that it is working properly;
;However, the user can force a new calibration using the "h8" command.

;Heat Mode Subroutine
400 ONEX1 3100          ;ignore data for now

;1 - Do calibration for sense channel
#ASM
         mov    p2,#0d7h
#ASM_END

IF INP2=ASC(8) THEN HC=0
GOSUB 350                       ;check second input, set simulators or thermistors
```

```
IF INP2=99 THEN GOTO 200 ;if error wait for new input

IF HC=1 THEN GOTO 410   ;if already cal'ed skip this part
GOSUB 4000              ;do sense mode calibration

#ASM
        mov   ta, #0aah     ;timed access protection stuff
        mov   ta, #055h
        setb  rwt           ;reset watchdog timer
#ASM_END

;2 - Calibrate heat channel
GOSUB 4100              ;calibrate the scaling network
IF INP2=ASC(8) THEN GOTO 200  ;if just calibrating, return for input

410 INPUT "R", RES      ;get desired Rset value from user
GOSUB 4200              ;set DAC to correct value and check it

;3 - Set up heat mode
GOSUB 350               ;check 2nd input, set simulator or thermistor
;set mux to AGND for heat, therm for sense
;p0.3 1
;p0.2 1
;p0.1 1
;p0.0 0
;p2.0 0 - switch H     NOTE order is significant
;p2.5 1 - switch D
;p2.3 0 - switch A
;p2.4 0 - switch C
;p2.6 0 - switch E
;p2.7 0 - switch F
#ASM
                        ;send out pulse on /SYNC
        clr   095h      ;to get A/D's running together
        setb  095h      ;set P3.4 /SYNC

        mov   p0,#0ceh  ;set up mux
        clr   0a0h      ;set switches for heating
        setb  0a5h
        clr   0a3h
        clr   0a4h
        clr   0a6h
        clr   0a7h
#ASM_END
```

```
;4 - Set up new interrupt
GOSUB 3600          ;time stamp for start of heat measurement
ONEX1 3300          ;on interrupt, put data in buffer

499 GOTO 200        ;go back for another input

;***************************************************************
;Print out data subroutine
500 IF INP2 = ASC(0) THEN GOTO 540
IF INP2 = ASC(2) THEN GOTO 560
GOTO 250            ;input error
540 GOSUB 4500
GOTO 200            ;return for new input
560 PRINT "ABCDEFGHIJKLMNOPQRSTUVWXYZ": GOTO 200

;***************************************************************
;Stop data collection subroutine
;can also put board into a known state for circuit testing
600 ONEX1 3100                      ;stop recording data
IF INP2=ASC(0) THEN GOTO 645        ;check for 2nd input
IF INP2=ASC(2) THEN GOTO 630
IF INP2=ASC(4) THEN GOTO 610
GOTO 250                            ;input error

610 DAC1=0fh: DAC0=0ffh             ;set controls for DAC
GOSUB 10200                         ;write to DAC
;Set ports for test mode with simulators, mux to AGND
;p2.0 1 - switch H
;p2.1 1 - sense bead relay
;p2.2 1 - heat bead relay
;p2.3 0 - switch A
;p2.4 1 - switch C and G
;p2.5 1 - switch D
;p2.6 1 - switch E
;p2.7 1 - switch F

#ASM
        mov    p0,#0c5h
        mov    p2,#0f7h
#ASM_END


630 REM leave in present state, turn off watchdog
#ASM
        mov    ta, #0aah
```

```
        mov    ta, #055h
        clr    ewt
#ASM_END
GOTO 200

645 REM stop measurement, go to calibration mode

#ASM
        mov    p2,#0ffh
        mov    p0,#0c5h
#ASM_END

GOTO 200     ;return for new input
;**************************************************************
;Debug Mode Control Subroutine
700 IF INP2=ASC(0) THEN GOTO 710
IF INP2=ASC(1) THEN GOTO 720
GOTO 250                        ;input error

710 DBUG=0                      ;debug mode off, watchdog on
CONA1=11H:CONA2=40H             ;set A/Ds back to 10 Hz
CONB1=11H:CONB2=40H
GOSUB 2100                      ;send new values to A/Ds

#ASM
        mov    ta, #0aah        ;timed access protection stuff
        mov    ta, #055h
        setb   rwt              ;restart watchdog timer

        mov    ta, #0aah        ;timed access protection stuff
        mov    ta, #055h
        setb   ewt              ;enable reset on timeout
#ASM_END

GOTO 200                        ;return for new input

720 DBUG=1                      ;debug mode on, watchdog off

#ASM
        mov    ta, #0aah        ;timed access protection stuff
        mov    ta, #055h
        clr    ewt              ;disable watchdog timeout
#ASM_END

CONA1=16H:CONA2=40H             ;set A/Ds to 2 Hz
```

```
CONB1=16H:CONB2=40H
GOSUB 2100                          ;send new values to A/Ds
GOTO 200                            ;return for new input

;****************************************************************
;Error handler
900 PRINT BNUM,SPC(1),ERRVALUE,SPC(1),ERRLINE
GOTO 200

;****************************************************************

;NEXT LEVEL OF SUBROUTINES

;****************************************************************
;Subroutine to send info to A/D control registers and verify
2100 COUNT=0
2110 GOSUB 10000                    ;write to cont reg
GOSUB 20000                         ;read from cont reg
CHCK=1                              ;check values
IF CRDA2 <> CONA2 THEN CHCK=0
IF CRDA1 <> CONA1 THEN CHCK=0
IF CRDA0 <> CONA0 THEN CHCK=0
IF CRDB2 <> CONB2 THEN CHCK=0
IF CRDB1 <> CONB1 THEN CHCK=0
IF CRDB0 <> CONB0 THEN CHCK=0
IF CHCK=1 THEN GOTO 2199                  ;if values OK, return
COUNT=COUNT+1
IF COUNT < 4 THEN GOTO 2110               ;if not correct, resend
PRINT "A/D ERR"                           ;error after 3 attempts
PH0. CONA2,SPC(1),CONA1,SPC(1),CONA0
PH0. CRDA2,SPC(1),CRDA1,SPC(1),CRDA0
PH0. CONB2,SPC(1),CONB1,SPC(1),CONB0
PH0. CRDB2,SPC(1),CRDB1,SPC(1),CRDB0
2199 RETURN

;****************************************************************
;****************************************************************
3100 GOSUB 23000                    ;read from A/Ds
3110 RETI

;****************************************************************
3200 GOSUB 23000                    ;grab data
T1=CAL1: M1=CAL2                          ;set up cal indicator
IF DBUG=0 THEN GOSUB 3350                 ;write to data buffer
IF DBUG=1 THEN GOSUB 3700                 ;print data to screen
```

```
3299 RETI

;*************************************************************
;Implement circular buffer
3300 GOSUB 23000
IF DBUG=0 THEN GOSUB 3350            ;write data to buffer
IF DBUG=1 THEN GOSUB 3700            ;print data to screen
3349 RETI

;*************************************************************
;Subroutine to write data to buffer
3350 IF DLAST > DMAX THEN DLAST=0          ;test don't exceed buffer size

THI=T1/BCONV: TLO=T1-(THI*BCONV)
MHI=M1/BCONV: MLO=M1-(MHI*BCONV)

XBY(DLOW+(10*DLAST)+0)=THI
XBY(DLOW+(10*DLAST)+1)=TLO
XBY(DLOW+(10*DLAST)+2)=MHI
XBY(DLOW+(10*DLAST)+3)=MLO
XBY(DLOW+(10*DLAST)+4)=DTA0
XBY(DLOW+(10*DLAST)+5)=DTA1
XBY(DLOW+(10*DLAST)+6)=DTB0
XBY(DLOW+(10*DLAST)+7)=DTB1

DLAST=DLAST+1
3399 RETURN

;************************************************************
;Timestamp Subroutine (for start of measurements)
3600 DTA0=0:DTA1=0:DTB0=0:DTB1=0
T1=TIME: M1=MSEC
THI=T1/BCONV: TLO=T1-(THI*BCONV)
MHI=M1/BCONV: MLO=M1-(MHI*BCONV)
GOSUB 3350
3630 RETURN

;*************************************************************
;Subroutine to print data to screen during debug mode
3700 V1=DTA0*BCONV+DTA1:V2=DTB0*BCONV+DTB1
TEMP1=10000: TEMP2=1 : TEMP3=32768:
TEMPIN=V1 : TEMP4=SAD1*2
GOSUB 4400
V1=TEMPOUT
TEMPIN=V2 : TEMP4=SAD2*2
```

```
GOSUB 4400
V2=TEMPOUT
PRINT BNUM,SPC(1),T1,SPC(1),M1,SPC(1),V1,SPC(1),V2
3799 RETURN
;****************************************************************
;Subroutine for grabbing calibration data
3800 ONEX1 3200                    ;set up new ONEX to get data

FOR DUM=1 TO CALNUM
IDLE                               ;wait to get interrupt
DUM2=DTA0*BCONV+DTA1
NEXT DUM

ONEX1 3100
3850 RETURN

;****************************************************************
;Subroutine to calibrate current sources for sense mode
;Sends out data to host, 1st from 500, then from 2500
4000 CAL1=0

ONEX1 3100          ;just grab data and ignore for now


;must first make sure that A/D is in proper mode
CONA0=0B0H                ; G=16, diff. channel
GOSUB 2100                ;check values
SAD1=64: CAL2=0

;set mux to 500 ohm precision resistor
;p0.7 1 - input ID 2
;p0.6 1 - input ID 1
;p0.5 0 - reset test pin
;p0.4 0 - test pin
;p0.3 0 - heat mux A1
;p0.2 1 - heat mux A0
;p0.1 0 - sense mux A1
;p0.0 1 - sense mux A0

#ASM
        mov     p0,#0c5h

        mov     ta, #0aah         ;timed access protection stuff
        mov     ta, #055h
        setb    rwt               ;reset watchdog timer
```

```
#ASM_END

;add a delay?
;read A/D
4020 GOSUB 3800                     ;grab calibration data

;set mux to 2500 ohm precision resistor
;p0.3 0 - heat mux A1
;p0.2 0 - heat mux A0
;p0.1 0 - sense mux A1
;p0.0 0 - sense mux A0

#ASM
        mov     p0,#0c0h
#ASM_END

;add a delay?

4030 CAL2=2 : GOSUB 3800            ;grab calibration data
4049 RETURN

;*************************************************************
;Subroutine to calibrate the scaling network for heat mode
;NOTE: Although the routine grabs CALNUM values for each calibration
;point and sends all of these to the data buffer, only the
;last value is currently used for the calculation of DCCODE and SCALE

4100 CAL1=10
;p2.0 1 - switch H
;p2.1 1 - sense bead relay
;p2.2 1 - heat bead relay
;p2.3 1 - switch A
;p2.4 1 - switch C
;p2.5 1 - switch D
;p2.6 1 - switch E
;p2.7 1 - switch F

;1st read voltage across 500 ohm resistor
;set mux to appropriate setting

#ASM
        mov     p2,#0ffh
        mov     p0,#0c5h
#ASM_END
```

```
;set up A/D in differential mode - gain of 4
CONA0=0A8H                          ;G=4, diff. channel
GOSUB 2100                          ;send and check A/D control values
SAD1=16: CAL2=0

;read data and store
ONEX1 3100                                ;clear any old data
IDLE
GOSUB 3800 : DCIN=DUM2                    ;grab calibration data

#ASM
        mov    ta, #0aah                  ;timed access protection stuff
        mov    ta, #055h
        setb   rwt                        ;reset watchdog timer
#ASM_END

;set DAC to 0
4130 DAC1=00h: DAC0=00h             ;set controls for DAC
GOSUB 10200                         ;write to DAC

;add some delay time for things to settle?
;set up A/D in single-ended mode, gain of 1
CONA0=0A2H                          ; G=1, sing. channel
GOSUB 2100                          ;send and check A/D control values
SAD1=1:CAL2=2

;read data and store
ONEX1 3100                          ;grab and ignore old data
IDLE
GOSUB 3800 : DCHI=DUM2              ;get cal data and store

;set DAC to 1
4160 DAC1=0fh: DAC0=0ffh            ;set controls for DAC
GOSUB 10200                         ;write to DAC
CAL2=4

;add some delay time for things to settle?

;read data and store
ONEX1 3100                          ;grab and ignore old data
IDLE
GOSUB 3800 : DCLO=DUM2              ;get cal data and store

HC=1                                ;set flag, cal complete
4199 RETURN
```

```
;*****************************************************************
;Subroutine to do calculations and test once have initial calibration
;information

;do calculation of code to scaling relationship
4200 DCDIF=DCHI-DCLO
; V1=DCHI*10000/DCDIF
; V2=RES*DCIN/(DCDIF*8)
; V1=V1-V2
; DCCODE=V1*4095/10000

TEMPIN=DCHI: TEMP1=10000: TEMP2=1: TEMP3=DCDIF: TEMP4=1
GOSUB 4400 : V1=TEMPOUT

TEMPIN=RES: TEMP1=DCIN: TEMP4=8
GOSUB 4400 : V2=TEMPOUT

V1=V1-V2

TEMPIN=V1: TEMP1=4095: TEMP3=10000: TEMP4=1
GOSUB 4400: DCCODE=TEMPOUT

;Set up for calibration mode and measurements
;May be unnecessary if just completed cal

#ASM
        mov    p2,#0ffh
        mov    p0,#0c5h
#ASM_END

;set up A/D in single-ended mode, gain of 1
CONA0=0A2H                  ; G=1, sing. channel
GOSUB 2100                  ;send and check A/D control values
SAD1=1

;Now that we have calculated code - must set up DAC
4220 IF DCCODE > 4095 THEN DCCODE=4095  ;don't let it exceed 12 bit max
DAC1=DCCODE/BCONV
DAC0=DCCODE-(DAC1*BCONV)              ;set up code for DAC
GOSUB 10200                           ;write to DAC

;Write code to data buffer
IDLE
T1=10:M1=6:DTA1=DAC0:DTA0=DAC1:DTB1=0:DTB0=0
```

```
GOSUB 3350                                    ;write code to buffer

;Read output just to be sure that it is correct
ONEX1 3100 :CAL2=8                            ;grab and ignore old data
IDLE
GOSUB 3800 : TEMPIN=DUM2                      ;get and store good data

;SCALE=TEMPIN*500*10/(DCIN/16)
4250 TEMP1=5000: TEMP2=16: TEMP3=DCIN/2: TEMP4=2
GOSUB 4400 : SCALE=TEMPOUT

TEMP1=SCALE/BCONV:TEMP2=SCALE-(TEMP1*BCONV)

;Write scale to data buffer
IDLE
T1=10:M1=10:DTB1=0:DTB0=0
DTA0=TEMP1:DTA1=TEMP2
GOSUB 3350          ;write code to buffer

4299 RETURN
;***********************************************************************
;Subroutine to perform 32-bit math
;Will perform 2 multiplications and 2 divisions
;(can set some values to 1 if don't need that many calculations)
4400 REM 32-bit math
#ASM
        mov   dptr, #__TEMPIN
        lcall Load_16X

        mov   dptr, #__TEMP1
        lcall Load_Mul16X
        lcall Mul_16

        mov   dptr, #__TEMP2
        lcall Load_Mul16X
        lcall Mul_16

        mov   dptr, #__TEMP3
        lcall Load_Div16X
        lcall Div_16

        mov   dptr, #__TEMP4
        lcall Load_Div16X
        lcall Div_16
```

```
        mov     dptr, #__TEMPOUT
        lcall  Low_16X
#ASM_END
4410 RETURN

;**********************************************************************
;Subroutine to read from circular data buffer
4500 DTEMP=DLAST                        ;subroutine to print out data
IF DFIRST=DTEMP THEN GOTO 4539      ;no data to send - just return
IF DFIRST>DTEMP THEN GOTO 4520      ;need to use 2 FOR loops

FOR DUM=DFIRST TO (DTEMP-1)
GOSUB 4600                  ;retrieve variables
NEXT DUM

GOTO 4538                   ;have sent data, update DFIRST and return

4520 FOR DUM=DFIRST TO DMAX
GOSUB 4600
NEXT DUM

4530 FOR DUM=0 TO (DTEMP-1)
GOSUB 4600
NEXT DUM

4538 DFIRST=DTEMP           ;keep track of last data sent out
4539 PRINT BNUM,SPC(1),"END"
RETURN                      ;return for another input

;**********************************************************************
;Subroutine to grab data from memory and print it
4600 TOUT=(XBY(DLOW+(10*DUM)+0)*BCONV)+XBY(DLOW+(10*DUM)+1)
4601 MOUT=(XBY(DLOW+(10*DUM)+2)*BCONV)+XBY(DLOW+(10*DUM)+3)
4602 XA=(XBY(DLOW+(10*DUM)+4)*BCONV)+XBY(DLOW+(10*DUM)+5)
4603 XB=(XBY(DLOW+(10*DUM)+6)*BCONV)+XBY(DLOW+(10*DUM)+7)
4604 PRINT BNUM,SPC(1),TOUT,SPC(1),MOUT,SPC(1),XA,SPC(1),XB
4605 RETURN

;*****************************************************************

;ASSEMBLY LANGUAGE ROUTINES START HERE

;*****************************************************************
9000  REM INIT
```

```
#ASM
smod_1 equ    0DFh          ;Ser Port 1 baud rate doubler bit
scon1  equ    0C0h          ;Ser Port 1 control register
scon0  equ    098h          ;Ser Port 0 control register
sbuf0  equ    099h          ;Ser Port 0 buffer
ckcon  equ    08eh          ;clock cont reg, chcon.4=tim
eie    equ    0e8h          ;external interrupts
ti1    equ    0C1h          ;Ser Port 1 transmitter int flag
sbuf1  equ    0c1h          ;Ser Port 1 buffer
ri1    equ    0C0h          ;Ser Port 1 receiver int flag
ren1   equ    0c4h          ;Ser Port 1 receive enable
pmr    equ    0c4h          ;power management register
ewt    equ    0d9h          ;enable watchdog timer reset
rwt    equ    0d8h          ;reset watchdog timer
wtrf   equ    0dah          ;watchdog timer reset flag
ta     equ    0c7h          ;timed access register

       clr   ri1            ;clear receive complete flag
       push  acc            ;save this value just in case
       mov   p1,#0ffh       ;set pins high for I/O, allow input

       clr   94h            ;set A0 low
       setb  smod_1         ;Turn on Ser Mode bit
       mov   p1,#2ch        ;keep DRDY high (only input)
       mov   scon1,#00H     ;Set Ser Port 1 to Mode 0

       setb  90h       ;set P1.0 /RFS1
       setb  91h       ;set P1.1 /TFS1
       setb  94h       ;set P1.4 A0
       setb  96h       ;set P1.6 /RFS2
       setb  97h       ;set P1.7 /TFS2
       setb  0b3h      ;set P3.3 /DRDY2
       setb  095h      ;set P1.5 /SYNC
       clr   0b5h      ;clear timing test bit
       setb  0b4h      ;set P3.4 /DRDY1

       mov   a,#0      ;clear the accumulator
       mov   sbuf1,a   ;flush serial port
       jnb   ti1,$     ;wait till gone
       clr   ti1

       pop   acc       ;recover value saved
#ASM_END
```

```
9100 RETURN
;*******************************************************************

10000 REM Subroutine to write to A/D control registers

#ASM
        push   acc            ;store values for other routines
        push   dpl
        push   dph
        push   b
        mov    a,r1
        push   acc


        mov    r1,#3h                ;# of bytes to write
        mov    dptr,#__CONA0         ;first location in data block
        clr    94h                   ;bring A0 low
        clr    91h                   ;bring /TFS1 low (active)

        lcall  send_chs              ;Send 3 bytes

        setb   91h                   ;set /TFS high
        setb   94h                   ;set A0 high

                                     ;now write to A/D 2
        mov    r1,#3h                ;# bytes to send
        mov    dptr,#__CONB0         ;first location in data block

        clr    94h                   ;bring A0 low
        clr    97h                   ;bring /TFS2 low (active)

        lcall  send_chs              ;Send 3 bytes

        setb   97h                   ;set /TFS high
        setb   94h                   ;set A0 high



        pop    b                     ;restore previous values for
        pop    acc                   ;the rest of the program to use
        mov    r1,a
        pop    dph
        pop    dpl
        pop    acc
```

```
                                        ;send out pulse on /SYNC
        clr    095h                     ;to get A/D's running together
        setb   095h                     ;set P1.5 /SYNC
#ASM_END

10100 RETURN
;*****************************************************************

;*****************************************************************
10200 REM Subroutine to write to DAC

#ASM
        push   acc             ;store values for other routines
        push   dpl
        push   dph
        push   b
        mov    a,r1
        push   acc


        clr    ti1
        mov    r1,#2h                   ;# of bytes to write
        mov    dptr,#__DAC1             ;first location in data block

        setb   0b5h                     ;bring LOAD high

        lcall  send_chs

        clr    0b5h                     ;clear LOAD
        nop
        nop
        nop
        setb   0b5h                     ;bring LOAD high again

        pop    b                        ;restore previous values for
        pop    acc                      ;the rest of the program to use
        mov    r1,a
        pop    dph
        pop    dpl
        pop    acc

#ASM_END

10210 RETURN
```

```
#ASM
;*****************************************************************
;Subroutine to write (r1) bytes of data to either A/D1, A/D2 or DAC
;
send_chs:    movx   a,@dptr        ;get byte
        acall  rotate
        mov    sbuf1,a             ;send  char to transmit buf
        jnb    ti1,$               ;wait until transfer complete
        clr    ti1                 ;clear flag
        inc    dptr                ;next location for next pass
        dec    r1
        mov    a,r1                ;check how many bytes left
        jnz    send_chs            ;repeat until all bytes sent
        ret

;*****************************************************************
;Subroutine to read data from either A/D1 and A/D2 depending on mask
;read_st is a separate entry point for reading control regs (not data)
;Mask = 10h for AD1 or 08h for AD2

read_prt:    mov    a,P3           ;read port one (AD1 or 2)
        anl    a,r6                ;mask /DRDY
        jnz    read_prt            ;wait for data ready

             ;Entry pt for reading control registers only
read_st:     clr    ri1            ;clear receive flag
get_cha:     mov    a,r6           ;recognize AD1 or 2 by its mask
        anl    a,#10h              ;AD1 drdy bit
        jz     doport2
        clr    90h                 ;set /RFS (1) low
        ajmp        readch
doport2:     clr    96h            ;set /RFS (2) low
readch:      setb   ren1           ;start read
        jnb    ri1,$               ;wait until read complete
        clr    ren1
        clr    ri1

        mov    a,r6                ;recognize AD1 or 2 by its mask
        anl    a,#10h              ;AD1 drdy bit
        jz     dopt2
        setb   90h                 ;set /RFS 1 hi
        ajmp   readbuf
dopt2:       setb   96h            ;set /RFS 2 hi

readbuf:     MOV    a,sbuf1        ;get char
```

```
        acall   rotate

        movx    @dptr,a             ;store char
        inc     dptr                ;next location for next pass
            dec     r1
            mov     a,r1            ;check how many left
            ret
;*************************************************************
;Assembly code subroutine to rotate bits for communications

rotate:   rlc     a                 ;rearange bits, A/D sends MSB 1st
        mov     b.0,c
        rlc     a
        mov     b.1,c
        rlc     a
        mov     b.2,c
        rlc     a
        mov     b.3,c
        rlc     a
        mov     b.4,c
        rlc     a
        mov     b.5,c
        rlc     a
        mov     b.6,c
        rlc     a
        mov     b.7,c
        mov     a,b                 ;rearrange complete
        ret
 #ASM_END


;*************************************************************
20000 REM Subroutine to read from A/D control registers

#ASM

        push    acc         ;save the acc, etc. for other
        push    dpl         ;parts of the code
        push    dph
        push    b
        mov     a,r1
        push    acc
        mov     a,r6
        push    acc
```

```
          setb   90h          ;set P1.0 /RFS1
          setb   91h          ;set P1.1 /TFS1

          clr    94h          ;set A0 low/control reg

          mov    r6,#10h              ;tells read_st to read AD1
          mov    r1,#3h               ;# of bytes to read
          mov    dptr,#__CRDA0        ;first location in data block

          clr    ri1                  ;clear receive complete flga
get_ad1:     acall   read_st          ;read a char
          jnz    get_ad1              ;repeat until done
          setb   90h                  ;Set /RFS back high

                                      ;start reading from A/D2
          setb   96h                  ;set P1.6 /RFS2
          setb   97h                  ;set P1.7 /TFS2

          clr    94h                  ;set A0 low/control reg

          mov    r1,#3h               ;# of bytes to write
          mov    dptr,#__CRDB0        ;first location in data block

          clr    ri1                  ;clear receive complete flag
          mov    r6,#8h               ;tells read_st to read AD2
get_ad2:     acall   read_st          ;read a char
          jnz    get_ad2              ;repeat as necessary
          setb   96h                  ;Set /RFS2 back high


          pop    acc                  ;restore values that may be
          mov    r6,a                 ;used by other sections of
          pop    acc                  ;the code
          mov    r1,a
          pop    dph
          pop    b
          pop    dpl
          pop    acc

#ASM_END

20100 RETURN

;*******************************************************************
```

```
23000 T1=TIME: M1=MSEC

#ASM

        push   acc          ;save values for use later
        push   dpl
        push   dph
        push   b
        mov    a,r1
        push   acc
        mov    a,r6
        push   acc

                            ;start reading from A/D1
        setb   90h          ;set P1.0 /RFS
        setb   91h          ;set P1.1 /TFS
        setb   94h          ;set P1.4 A0  high

        mov    r1,#2h              ;# of bytes to write
        mov    dptr,#__DTA0        ;first location in data block
        mov    r6,#010h            ;Using P3.4 as /DRDY1
read_ad1:    lcall   read_prt      ;read data
        jnz    read_ad1            ;repeat as necessary
        setb   90h                 ;Set /RFS back high


                                   ;Now read from A/D2
        setb   96h                 ;set P1.6 /RFS2
        setb   97h                 ;set P1.7 /TFS2
        setb   94h                 ;set P1.4 A0  high

        mov    r1,#2h              ;# of bytes to write
        mov    dptr,#__DTB0        ;first location in data block
        mov    r6,#08h             ;Using P3.3 as /DRDY2
read_ad2:    lcall   read_prt      ;read AD1 or 2 depending on mask
        jnz    read_ad2
        setb   96h                 ;Set /RFS2 back high


        pop    acc                 ;recover values saved at start
        mov    r6,a                ;of subroutine
        pop    acc
        mov    r1,a
        pop    dph
        pop    b
```

```
        pop     dpl
        pop     acc

#ASM_END

23010 RETURN
```

References

American National Standard, Safe Current Limits for Electromedical Apparatus, Association for the Advancement of Medical Instrumentation, 1985.

Balasubramaniam, T. A. and Bowman, H. F., "Temperature Field Due to a Time Dependent Heat Source of Spherical Geometry in an Infinite Medium," *Journal of Heat Transfer*, August 1974.

Balasubramaniam, T. A. and Bowman, H. F., "Thermal Conductivity and Thermal Diffusivity of Biomaterials: A Simultaneous Measurement Technique," *J. Biomech. Eng.*, 99:148, 1977.

Bowman, H. F. and Balasubramaniam, T. A., "An Instrument for Monitoring Tissue Perfusion Rates in Isolated Organs," Tenth Annual Meeting, Society for Cryobiology, 1973.

Bowman, H. Frederick, Cravalho, Ernest G., and Woods, Monty, "Theory, Measurement, and Application of Thermal Properties of Biomaterials," *Annual Review of Biophysics and Bioengineering*, Vol. 4, 1975, pp 43-80.

Bowman, H. Frederick and Balasubramaniam, T. A., "A New Technique Utilizing Thermistor Probes for the Measurement of Thermal Properties of Biomaterials," *Cryobiology*, Vol. 13, Academic Press, 1976, pp. 572-580.

Bowman, H. F., Balasubramaniam, T. A., and Woods, Monty, "Determination of Tissue Perfusion from *In Vivo* Thermal Conductivity Measurements," Winter Annual Meeting, Heat Transfer Division of The American Society of Mechanical Engineers, November-December 1977.

Bowman, H. Frederick, "The Bio-Heat Transfer Equation and Discrimination of Thermally Significant Vessels," *Annals of the New York Academy of Sciences*, Vol. 335, 1980, pp 155-160.

Bowman, H. F., "Estimation of Tissue Blood Flow," *Heat Transfer in Medicine and Biology*, Plenum Press, 1984, pp 193-230.

Bowman and W. H. Newman, "Method to Quantify Thermal Dissipative Mechanisms in Biomaterials," US Patent No. 4,852,027, Assigned to Thermal Technologies, Inc., 1989.

Klar, E., Kraus, T., Bleyl, J., Newman, W., Bowman, F., Kummer, R. V., Ottos, G., and Herfarth, C., "Thermodiffusion as a Novel Method for Continuous Monitoring of the Hepatic Microcirculation After Liver Transplantation," *Transplantation Proceedings*, Vol. 27, No. 5 (October), 1995, pp. 2610-2612.

Klar, E., Kraus, T., Bredt, M., Osswald, B., Senninger, N., Herfarth, C., and Otto, G. "First Clinical Realization of Continuous Monitoring of Liver Microcirculation After Transplantation by Thermodiffusion," *Transpl. Int.* (1996) s140-s143.

Klar, E., Kraus, T., Osswald, B., Mehrabi, A., Bleyl, J., Herfarth, C., and Otto, G., "Necessity of a Recovery Phase After In Situ Liver Preparation to Improve Hepatic Microcirculation Prior to Organ Preservation," *Transplantation Proceedings*, Vol. 28, No. 3 (June), 1996, pp 1867-1868.

Klar, E., Bredt, M., Kraus, T., Angelescu, M., Mehrabi, A., Sennigner, N., Otto, G., and Herfarth, C., "Early Assessment of Reperfusion Injury by Intraoperative Quantification of Hepatic Microcirculation in Patients," *Transplantation Proceedings*, Vol. 29, pp. 362-363, 1997.

Newman, William H., Bowman, H. Frederick, Orgill, Dennis, P., and Klar, Ernst, "A Methodology for *In Vivo* Measurement of Blood Flow in Small Tissue Volumes," HTD-Vol. 322/BED-Vol. 32, *Advances in Heat and Mass Transfer in Biotechnology*, ASME 1995.

Oakes, J.B., "Electrical Safety," *CRC Handbook of Clinical Engineering, Volume I, General Principles*, Barry N. Feinberf, editor, CRC Press, Boca Raton, Florida, pp 23-33.

Patera, A. T., Eden, G., Mikic, B. B., and Bowman, H. F., "Prediction of Tissue Perfusion from Measurement of the Phase Shift between Heat Flux and Temperature," Winter Annual Meeting, Heat Transfer Division of The American Society of Mechanical Engineers, December 1979.

Szajda, Kenneth Stanley, *A Data Acquisition System for Temperature and Perfusion Measurement*, Master's Thesis, Massachusetts Institute of Technology, June 1989.

Tsai, Andy, *Investigation of Variability in Skin Tissue Instrinsic Thermal Conductivity Measurements*, Master's Thesis, Massachusetts Institute of Technology, May 1995.

Valvano, J. W., *The Use of Thermal Diffusivity to Quantify Tissue Perfusion, Ph. D. Thesis*, Harvard-Massachusetts Institute of Technology Division of Health Sciences and Technology, August 1981.

Valvano, J. W., Allen, J. T., and Bowman, H. F., "The Simultaneous Measurement of Thermal Conductivity, Thermal Diffusivity, and Perfusion in Small Volumes of Tissue," Winter Annual Meeting, Heat Transfer Division of The American Society of Mechanical Engineers, November 1981.

Valvano, J. W., Allen, J. T., Walsh, J. T., Hnatowich, D. J., Tomera, J. F., Brunengraber, H., and Bowman, H. F., "An Isolated Rat Liver Model for Evaluation of Thermal

Techniques to Quantify Perfusion," *Journal of Biomechanical Engineering*, Vol. 106, August 1984.