Application Report
SPRA873 - February 2003

Thermoelectric Cooler Control Using a TMS320F2812 DSP and a DRV592 Power Amplifier

David M. Alter, Ph.D. *DSP Applications - Semiconductor Group*

ABSTRACT

This application report presents a thermoelectric cooler system consisting of a Texas Instruments TMS320F2812 digital signal processor (DSP) and DRV592 power amplifier. The DSP implements a digital proportional-integral-derivative feedback controller using an integrated 12-bit analog-to-digital converter to read the thermistor, and direct output of pulse-width-modulated waveforms to the H-bridge DRV592 power amplifier. The system presented provides up to 6.1 watts of heating or cooling to the laser mount, although the DRV592 amplifier is actually capable of delivering up to 15 watts when configured appropriately. The closed-loop TEC system is seen to achieve ±0.0006°C temperature accuracy, depending on the needed operating temperature range, with a step response settling time of 14 to 16 seconds. A complete description of the experimental system, along with software and software operating instructions, are provided.

Contents

List of Figures

List of Tables

1 Introduction

Thermoelectric coolers (TEC) find widespread application in optical networking systems. The wavelength of light emitted by a laser is directly dependent on the laser temperature, typically by a factor of around 0.1 nm/°C. The thermal expansion and contraction of the laser material couples temperature and wavelength, since the physical length of the lasing chamber determines the wavelength of emitted light. As dense-wave-division multiplexed (DWDM) systems drive wavelength spacing on a fiber below 1 nm per channel towards 0.1 nm and beyond, accurate temperature control becomes critical.

In optical networking systems, TEC requirements generally fall into one of two categories. The first category is that of pump lasers, which are used in erbium-doped fiber amplifiers and raman amplifiers. These devices generate from 20 mW to 500 mW of optical power, dissipate between 2 W and 16 W of thermal energy, and require temperature regulation from ±1°C to ±0.1°C. The second category is that of transponder lasers (both fixed wavelength and tunable), which are used as laser light sources in network headend racks, wavelength converters, and elsewhere in the network. These devices generate much less power than pump lasers, typically being less than 20 mW of optical power. Thermal dissipation is generally less than 5 W, but required temperature accuracy ranges from ±0.1°C down to ±0.01°C, and possibly even tighter.

Figure 1 shows a block diagram of the basic temperature feedback control system.

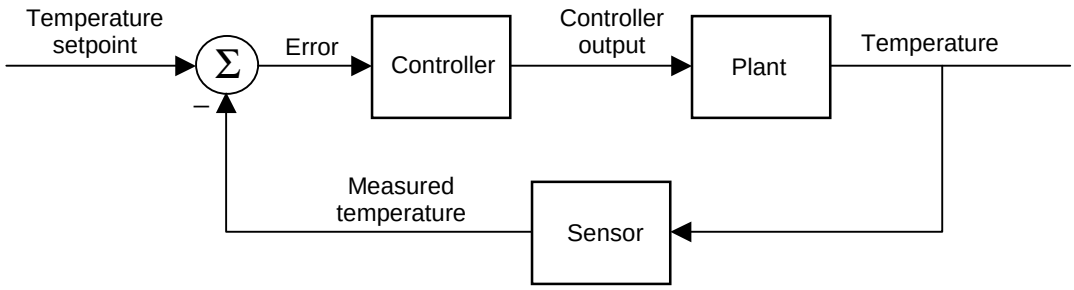


Figure 1. Basic Temperature Feedback Control System

The plant consists of three parts. The first part is the thermal dynamics of the laser, which may be modeled as a first-order differential equation. The relatively large thermal mass of the laser produces a slow open loop time constant on the order of 0.1 to 1 seconds. The second part of the plant is the active heating and cooling element known as a Peltier element. Electrically, a Peltier element looks mostly like a resistor. When current flows through the element in one direction, it sources heat (i.e., it provides heating). Alternately, reversing the current flow causes the element to sink heat (i.e., it provides cooling). This allows a single Peltier element to perform both the heating and cooling functions in a TEC system. Finally, the third part of the plant consists of the power amplifier which drives current through the Peltier element. The power amplifier can be of either the linear type (class A-B), or the switching type (class D). While linear amplifiers offer superior noise emission properties, switching amplifiers have far greater power efficiency and therefore dissipate much less heat. They provide a more compact solution in terms of board space since the large heat sinks needed for linear amplifiers are unnecessary. Indeed, the low heat dissipation seen in TEC applications means that heat sinks can often be avoided all together when using a switching amplifier.

The sensor commonly used in TEC applications is a simple thermistor, a device whose resistance varies with temperature. This variation is typically an exponential function, and hence introduces a nonlinearity into the closed-loop control system. However, given the slow time constants of the open-loop plant and the relatively low bandwidth requirements demanded of the controller, this nonlinearity does not cause any particular difficulties from a control point of view.

Figure 1 shows the controller block in the standard error driven feedback configuration. This configuration is highly suited to regulation control problems where the setpoint is essentially constant (as opposed to tracking control problems where the reference signal varies as a function of time). Due to the slow plant dynamics previously mentioned, TEC control demands a fairly low bandwidth controller. Simple proportional-integral-derivative (PID) control almost always suffices to achieve desired closed-loop performance. The traditional approach for implementing PID control for TEC has been with analog circuitry. While PID control can be effectively implemented using a few op-amps, resistors, and capacitors, digital control offers many well known advantages over analog control. Digital controllers are insensitive to temperature changes, and avoid the issues associated with component variation (e.g., resistor and capacitor tolerances), aging, and drift. Digital control also allows easy and precise controller tuning in software, as opposed to the more time consuming and less repeatable analog control tuning typically done via trim potentiometers. In TEC applications, this also means that a single controller (both software and hardware) can be designed and utilized across multiple laser platforms and products without the need for extensive component changes and redesign.

Low controller bandwidth requirements mean that the processing horsepower of a DSP is not really needed, and one could use a lower performance microcontroller instead. However, the advantage of using a DSP to implement the control is that a DSP is typically already in use in the system for controlling the laser power or amplifier gain. By using that same DSP for TEC control, one essentially gets the TEC controller for free in terms of cost and more importantly, board space. The TMS320C2000™ family of DSPs from Texas Instruments offers highly integrated solutions including on-chip analog-to-digital converters, pulse-width modulated signal generators, and flash EEPROM memory, providing a complete and compact controller solution for multiple aspects of the system control.

2 Discrete-Time PID Controller

The feedback controller used in TEC applications almost always utilizes the PID algorithm. The proportional control mode provides rapid closed-loop transient response to step changes in the temperature setpoint and fast rejection of thermal disturbances. The integral control mode guarantees zero steady-state error with a constant temperature setpoint and constant thermal disturbance (such as heat dissipation from the laser). In other words, integral control guarantees that the final laser temperature will match the temperature setpoint. Finally, the derivative control mode reduces transient overshoot and oscillation in the temperature when the setpoint is changed from one value to another. The PID algorithm is extremely pervasive in control systems textbooks, and will not be extensively reviewed here. Rather, only the formulation of the PID algorithm as implemented in this application report will be presented. The reader may refer to almost any control systems textbook, such as reference [1] or [2], for additional information.

2.1 Mathematical Formulation

The PID algorithm can be expressed in the continuous-time domain as:

$$u(t) = K_p e(t) + K_i \int_{\tau=0}^{t} e(\tau) d\tau + K_d \frac{d}{dt}\left[e(t)\right] \quad (1)$$

TMS320C2000 is a trademark of Texas Instruments.

Where:

u(t) is the control signal
e(t) is the error signal
t is the continuous-time domain time variable
τ is the calculus variable of integration
K_p is the proportional mode control gain
K_i is the integral mode control gain
K_d is the derivative mode control gain

Implementing this algorithm using a DSP requires one to transform it into the discrete-time domain. There are a number of methods available for this task. Perhaps the most straightforward technique uses a trapezoidal sum approximation for the integral term, and a backwards-difference approximation for the derivative term. The proportional term is directly used without approximation.

P term: $$K_p e(t) = K_p e(k) \quad (2a)$$

I term: $$K_i \int_{\tau=0}^{t} e(\tau) d\tau \cong K_i \sum_{i=0}^{k} \frac{h}{2}\left[e(i) + e(i-1)\right] \quad (2b)$$

D term: $$K_d \frac{d}{dt}\left[e(t)\right] \cong K_d \left[\frac{e(k) - e(k-1)}{h}\right] \quad (2c)$$

Time relationship: $$t = k * h \quad (2d)$$

Where:

h is the sampling period
k is the discrete-time index: k = 0, 1, 2, …

For simplification purposes, it is convenient to define new controller gains as:

$$K_i' = K_i \frac{h}{2} \quad (3a)$$

$$K_d' = \frac{K_d}{h} \quad (3b)$$

from which one can construct the discrete-time PID control law as:

$$u(k) = K_p e(k) + K_i' \sum_{i=0}^{k} \left[e(i) + e(i-1)\right] + K_d' \left[e(k) - e(k-1)\right] \quad (4)$$

To eliminate the need to calculate the full summation each time step (which would require an ever increasing amount of computation as time goes on), the summation is expressed as a running sum:

$$sum(k) = sum(k-1) + \left[e(k) + e(k-1)\right] \quad (5a)$$

$$u(k) = K_p e(k) + K_i' sum(k) + K_d' \left[e(k) - e(k-1)\right] \quad (5b)$$

These two equations represent the discrete-time PID control law utilized in this work.

2.2 Implementation Issues

Since the open-loop plant is a slow first-order system, involved analytical system modeling and controller design are not needed. Instead, the controller gains can be easily tuned by hand. However, one must still choose an appropriate sample rate for the discrete-time controller. A rule of thumb for satisfactory closed-loop control performance says that one should sample 5 to 20 times faster than the desired closed-loop system bandwidth. Assuming a desired closed-loop bandwidth of 10 Hz, a 200-Hz sampling rate should be more than sufficient.

Control signal saturation and integral mode anti-windup limiting are easily implemented in software. In this work, the control signal itself takes the form of pulse-width modulated (PWM) outputs from the DSP. Therefore, the control signal is saturated at the value that corresponds to 100% duty cycle for the PWM. An undesirable side-effect of saturating the controller output is integral mode windup. When the control output saturates, the integral mode control term (i.e., the summation) will continue to grow, but will not produce a corresponding increase in controller output (and hence will not produce any additional increase in plant response). The integral can become quite large, and it can take a long time before the controller is able to reduce it once the error signal changes sign. The effect of windup on the closed-loop output is larger transient overshoot and undershoot, and longer settling times. One approach for combating integral-mode windup is to simply limit in software the maximum absolute value allowed for the integral, independent of the controller output saturation. This is the method implemented in this work.

3 Experimental Setup

The TEC system was conveniently constructed using several evaluation boards available from Texas Instruments, as well as a laser diode mount with integrated Peltier and thermistor from Thor Labs, Inc. Figure 2 shows a photograph of the experimental setup, while Figure 3 shows a block schematic. Each major block will now be discussed in greater detail.

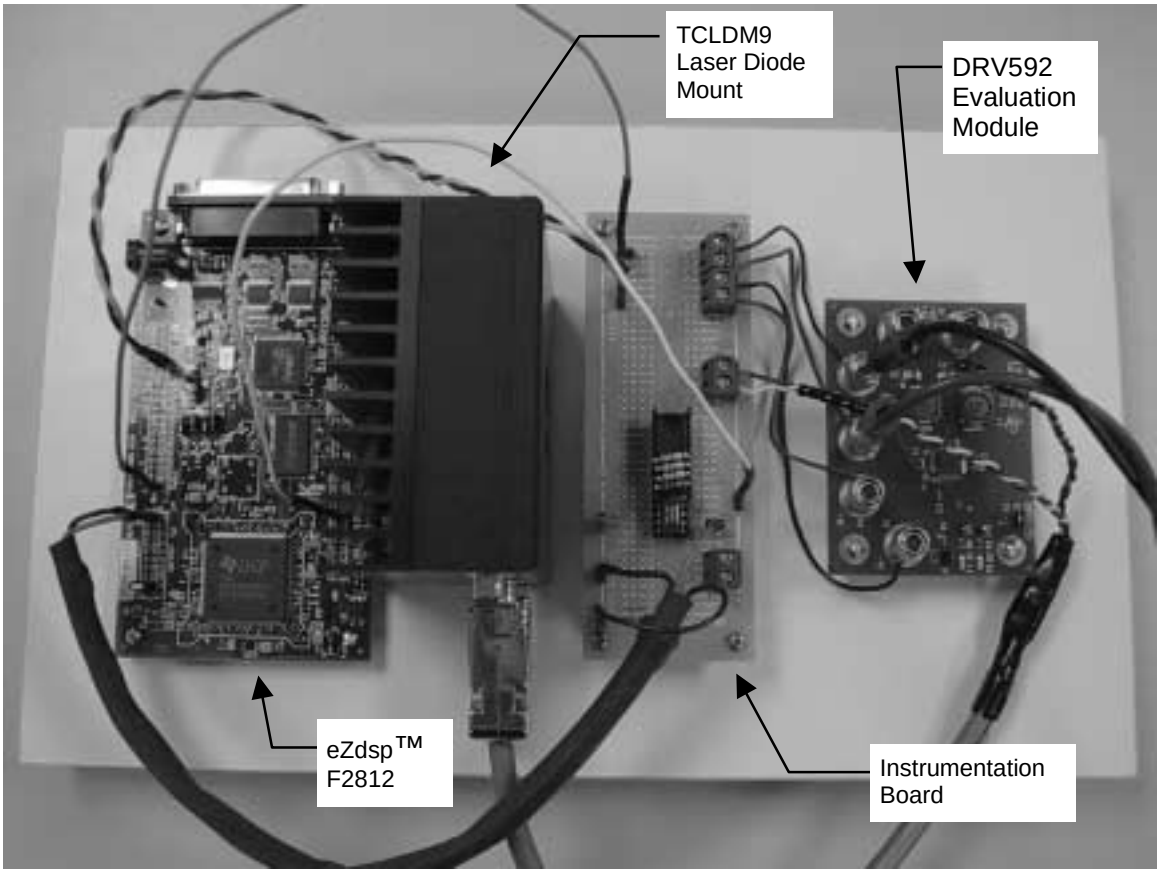


Figure 2. Photograph of Experimental Setup

eZdsp is a trademark of Spectrum Digital Inc.

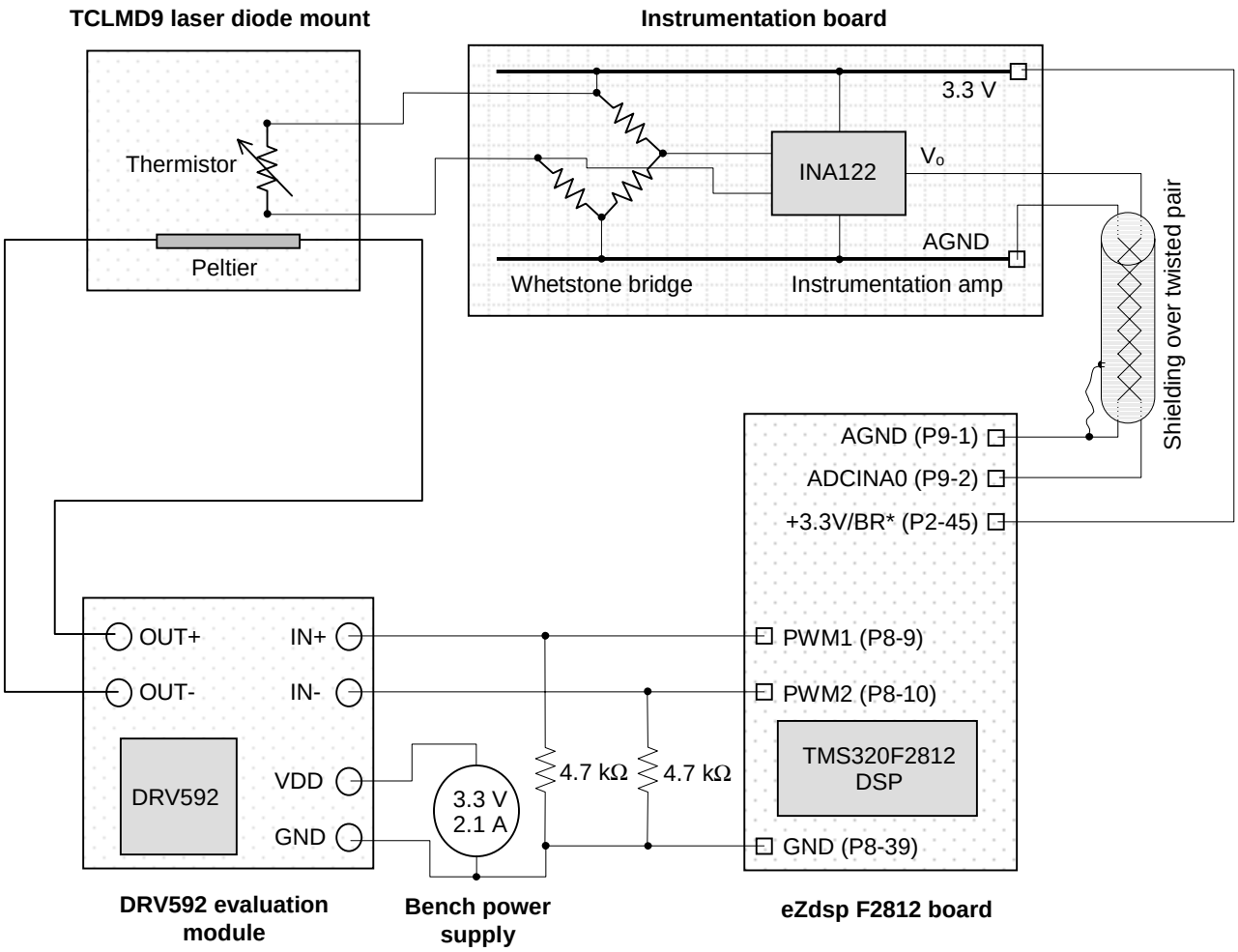


Figure 3. Block Schematic of Experimental Setup

3.1 eZdsp F2812 Board

The eZdsp F2812 board is available from Texas Instruments as a development tool. The eZdsp kit provides a complete development environment, and includes the DSP board, power supply for the board, on-board JTAG compliant emulator, and an eZdsp specific version of the Code Composer Studio™ integrated development environment (full featured, including debugger IDE, and ANSI C and C++ compliant compiler). The DSP board itself has nearly all peripheral signals available on the board headers, making it easy to interface the board with other system hardware.

NOTE: The eZdsp board requires that pins 17 and 18 on header P9 be connected in order for the analog-to-digital converter (ADC) to function properly (i.e., connect ADCLO to analog ground on the DSP). This connection must be manually made on the board.

Code Composer Studio is a trademark of Texas Instruments

3.2 TCLMD9 Laser Diode Mount

The laser diode mount is available as a retail product from Thorlabs Inc., item #TCLDM9:

Thorlabs, Inc.
435 Route 206
Newton, NJ 07860
Tel. 973-579-7227
http://www.thorlabs.com

A laser diode was not actually installed in the laser mount during this project. However, the TCLMD9 incorporates built-in Peltier elements and a thermistor which can be utilized independent of a laser diode.

3.2.1 Peltier Elements

The integrated twin Peltier elements each provide up to 13 W of heating and cooling power with maximum operating current and voltage of 5.6 A and 4 V, respectively. Total power capability is therefore 26 W. Inside the laser mount the Peltier elements are connected in series, although note that Figure 3 depicts only a single Peltier representing the series combination of the two. In this application report, the Peltier elements were driven at 3.3 V (from the DRV592) to illustrate a complete TEC solution operating solely from a 3.3-V supply. With a 3.3-V supply and a logic high signal applied to one of the DRV592 inputs, the Peltier elements are seen to draw current of approximately 2.1 A from the supply, or a supply power of 6.9 W. Of this, roughly 0.8 W is dissipated in the DRV592 (nominal high-side on-resistance of 80 mW, and nominal low-side on-resistance of 90 mW. See reference [3] for more information). This leaves peak power of approximately 6.1 W available for heating and cooling the laser.

3.2.2 Thermistor

The TCLMD9 comes with an integrated 10-kΩ NTC thermistor. Resistance versus temperature data was provided in the TCLMD9 documentation, and is shown in Appendix A of this report. Since the goal of this project is the demonstration of steady temperature regulation (i.e., small temperature deviation around the steady-state value), and not necessarily absolute temperature accuracy (i.e., making the steady-state temperature value match an actual desired value), the actual temperature corresponding to a thermistor reading is not of particular interest. However, it is still useful to develop a mathematical model for the thermistor to provide a basis for measuring temperature stability and closed-loop system response. The classic first-order Steinhart-Hart model for an NTC thermistor is:

$$R(T) = e^{\left(A + \frac{B}{T}\right)} \tag{6}$$

Where:

R(T) is electrical resistance as a function of temperature (Ω)
T is the temperature (*K*)
A is the offset constant ($ln(\Omega)$)
B is the slope constant ($ln(\Omega)\cdot K$)

The temperature range of interest in telecommunication laser applications is often centered between 25 and 30 degrees Celsius. Selecting two data points in this range from Table A-1 in Appendix A (*R = 10 kΩ* at 25°C, and *R = 8.055 kΩ* at 30°C), one can compute the following values for the constants A and B:

$$A = -3.902\ \ln(\Omega) \quad \text{and} \quad B = 3909\ \ln(\Omega) \cdot K \tag{7}$$

Table A-1, Figure A-1, and Figure A-2 in Appendix A show the resistance versus temperature behavior of the model against vendor supplied TCLMD9 data.

3.3 Instrumentation Board

The instrumentation board converts the resistance of the thermistor into a corresponding voltage that can be read by the ADC of the DSP. There are several common methods for instrumenting the thermistor in order to provide a voltage signal to the ADC. The classical method (which is the one used in this work) uses the thermistor as one of the four resistors in a Whetstone bridge. The bridge output is then feed to an instrumentation amplifier whose output can be read by the ADC. The output voltage produced is proportional to the change in resistance of the thermistor. This method offers simplicity and requires no active components in the bridge itself, but absolute accuracy is sensitive to component variation in both the resistors making up the bridge and also the resistor that determines the gain of a typical instrumentation amp. The second method employs a constant current source to drive a known current through the thermistor. The thermistor resistance is then directly converted into voltage via Ohm's law. This method is more complex than the first in that it needs an active current source, but it does offer relative insensitivity to component variation. Other methods exist as well, such as a series connection of the thermistor with a known resistance, and then applying a constant voltage across the series connection. The voltage drop across the thermistor will be proportional to its resistance. In all cases, an instrumentation or operational amplifier provides the signal conditioning and gain before the voltage is feed into the ADC.

Figure 4 shows the circuit diagram for the instrumentation board. For this project, the instrumentation board was initially breadboarded, and later transferred to a single layer prototyping board with breadboard style solderable copper traces (Datak LKG Industries; Rockford, IL 61109, part no. 12-617).

A number of instrumentation amplifiers from Texas Instruments are suitable for thermistor instrumentation. Among the possibilities is the INA122, which was used in this work. The INA122 is a bipolar, single-supply, rail-to-rail amplifier, with low quiescent current being one of its key features (60 mA). It is available in a DIP package for easy breadboarding. Another choice is the INA326, which is a single-supply, rail-to-rail amplifier built in a complimentary metal oxide semiconductor (CMOS) process. It features extremely low parameter drift over time. A third possibility is the INA330, which was specifically designed for thermoelectric cooler applications. It is optimized for use in 10-kΩ thermistor-based temperature controllers. The INA330 provides thermistor excitation and generates an output voltage proportional to the difference in resistance applied to the inputs. It uses only one precision resistor plus the thermistor, thus providing an alternative to the traditional Whetstone bridge circuit.

Two 0.47-µF tantalum decoupling capacitors were used between the power rails on the board to filter out power supply noise. They were needed because the prototyping board offered little capacitance for a ground or power plane. In a real PCB design, such large decoupling caps would likely not be needed.

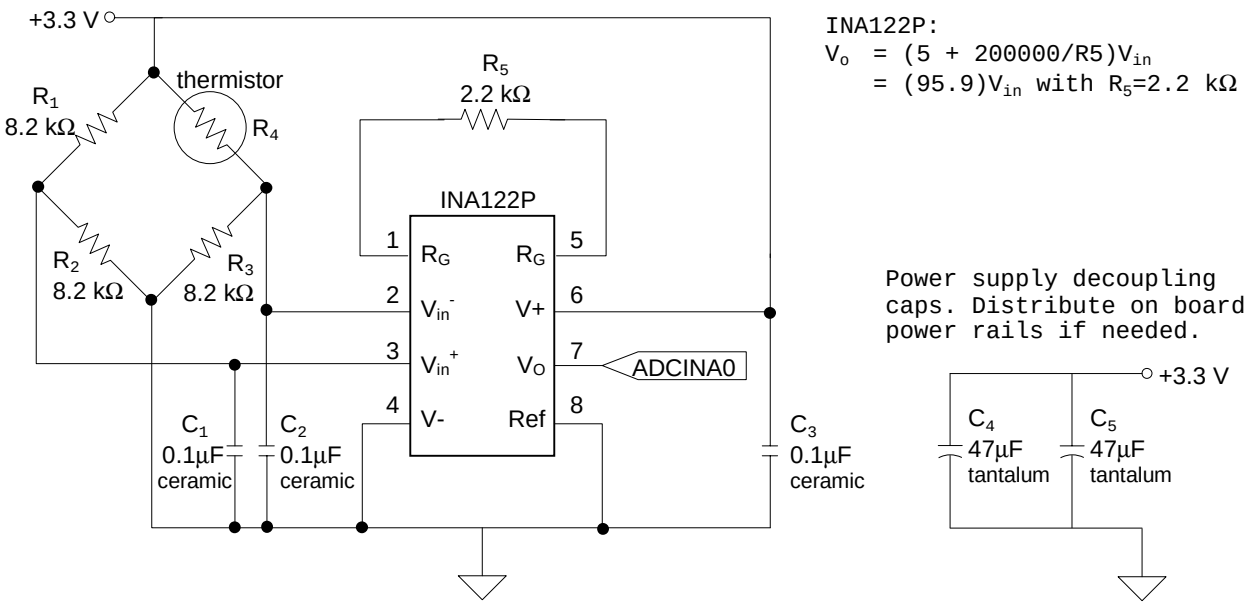


Figure 4. Instrumentation Board Circuit Detail

3.3.1 Whetstone Bridge

Figure 5 shows a detail of the Whetstone bridge. Simple linear circuit analysis of the bridge gives the following input-output equation:

$$V_{out} = \frac{R_2 R_4 - R_1 R_3}{(R_1 + R_2)(R_3 + R_4)} V_{in} \tag{8}$$

The resistor R_4 represents the thermistor. Resistors R_1, R_2, and R_3 are chosen to balance the bridge (i.e., zero output) when R_4 is at some desired value. Let $R_1 = R_2 = R_3 = R$, and $R_4 = (R+\Delta)$. Then, equation (8) reduces to:

$$V_{out} = \frac{\Delta}{4R + 2\Delta} V_{in} \tag{9}$$

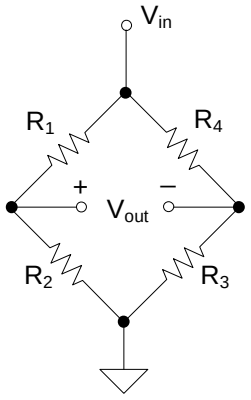


Figure 5. Whetstone Bridge Detail

For purposes of this work, a maximum operating temperature of about 30°C is desired. Using equations (6) and (7), one can determine that the thermistor will have a resistance of 8.2 kΩ at a temperature of 29.55°C. With resistors R_1, R_2, and R_3 chosen as 8.2 kΩ, V_{out} will increase from zero in equation (9) as temperature decreases from 29.55°C. Note that the thermistor introduces a sign inversion into the closed-loop control system: the sensor output decreases as temperature increases. This sign inversion is easy to deal with in software, or alternately one can simply swap the two PWM connections between the DSP and the DRV592 evaluation module and negate the effects of the sign inversion.

3.3.2 INA122P

The gain of the INA122P is determined by a single external resistor, R_5 in Figure 4. The gain equation from reference [5] is:

$$V_o = \left[5 + \frac{200000}{R_5}\right]\left(V_{in}^{+} - V_{in}^{-}\right) \tag{10}$$

Increasing the gain of the amplifier will increase the signal-to-noise ratio with respect to noise entering the signal chain after the instrumentation amplifier. The result of doing this is desirable in terms of noise reduction in the circuit, but reduces the temperature range of the system since the ADC in the F2812 DSP has a fixed full-scale range of 3.0 V. The noise entering the signal chain prior to the amplifier is assumed to be mostly common-mode noise, and therefore filtered out by the common-mode noise rejection capabilities of the instrumentation amplifier. In the experimental system, there was in fact a significant amount of noise entering the signal chain after the amplifier. R_5 was chosen as 2.2 kΩ, which results in a gain of 95.9 V/V. The downside of this choice is that the temperature operating range reduces to only about 1°C (i.e., the INA122P output will swing from 0 to 3 V with a temperature change of only 1°C from the maximum 29.55°C). For demonstration purposes, this was thought acceptable. Most of the noise is related to the fact that the system has been constructed using multiple circuit boards and with mostly unshielded wiring, long wiring paths, etc. In a well designed single-board layout, most of this noise would disappear.

Combining equations (6), (7), (8), and (10), Figure 6 can be constructed showing V_o as a function of temperature. A table of the raw data (including the converted value from the ADC) can be found in Appendix B. Although Figure 6 looks linear to the human eye, keep in mind that it is really exponential in form due to equation (6).

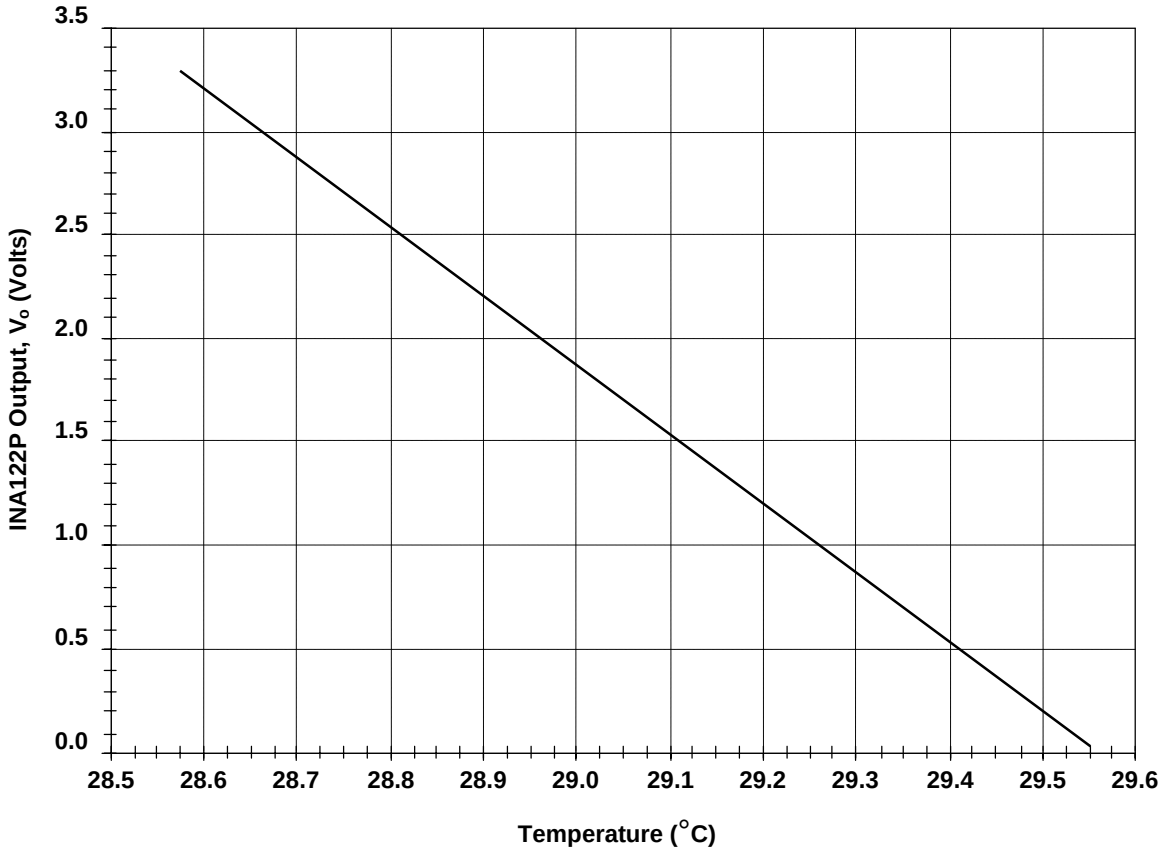
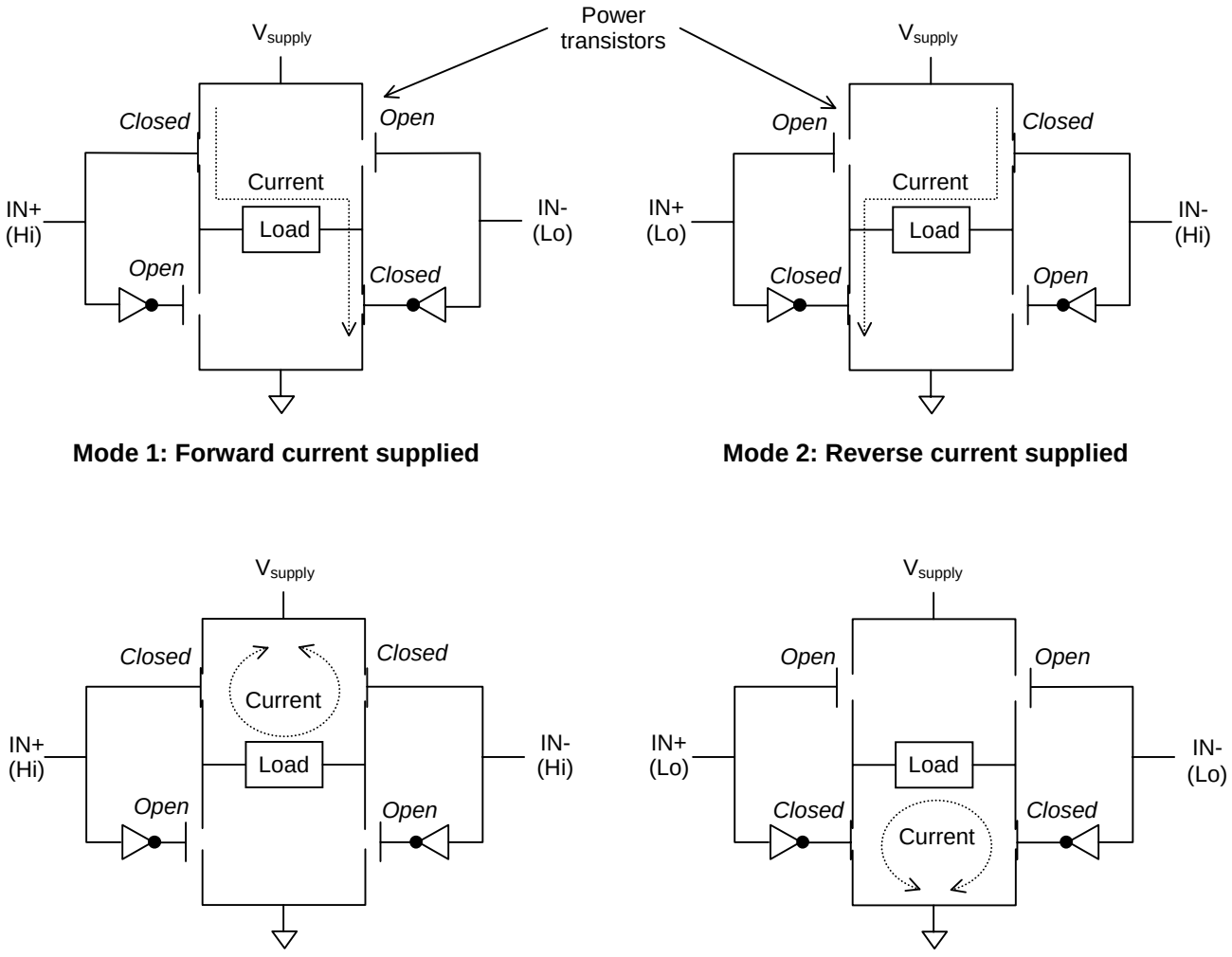


Figure 6. Instrumentation Circuit Output vs. Temperature (Theoretical)

3.4 DRV592 Evaluation Module

This application report employs the DRV592 class D power amplifier from Texas Instruments, conveniently mounted on an evaluation module (EVM) board (the DRV592 evaluation module is available from Texas Instruments). The DRV592 was designed specifically for TEC control, and provides a H-bridge configuration of four power MOSFETS in a single chip. The H-bridge configuration allows bidirectional current flow through the Peltier element using only a single-sided power supply. Directional control is achieved by closing the transistor gates along one of the two diagonals, as shown in Figure 7. In this way, both heating and cooling can be provided with a single chip.



Mode 1: Forward current supplied

Mode 2: Reverse current supplied

Mode 3a: Recirculating high-side current

Mode 3b: Recirculating low-side current

Figure 7. Basic Operating Modes of DRV592 H-Bridge Power Amplifier

The DRV592 inputs (IN+ and IN-) are high-impedance inputs that accept TTL-level signals. The PWM outputs from the F2812 DSP can be directly connected to these inputs. Further, only two PWM outputs are needed, as the two inputs to each half-bridge are connected together in an inverse relationship inside the DRV592.

In modes 3a and 3b, no additional energy is transferred to the load. Rather, current is allowed to recirculate though a closed-loop path that contains the load. Note that the direction of current flow in the recirculating modes is indeterminate. For an inductive load, the current will continue to circulate in the direction it was flowing prior to entering the recirculating mode. For example, suppose current was flowing through the load from left to right via mode 1 when the DRV592 inputs where changed to mode 3b. After entering mode 3b, current will continue to flow through the load from left to right since the inductive load will resist changes in the direction of current flow. Therefore, the recirculating direction will be clockwise for mode 3b in this example. It is easy to present an alternate case that shows counter-clockwise current flow in mode 3b. From the temperature control perspective, the direction of current flow in the recirculating modes is basically inconsequential. Since the load has a resistive component, any energy stored in the load (i.e., inductive or capacitive storage) at the instant a recirculating mode is entered will quickly be dissipated. Such action is all handled by the control system anyway, provided the controller sample rate is fast enough.

The load in Figure 7 is comprised of the Peltier element inside the TCLDM9 laser diode mount, along with an LC-type low-pass filter on the DRV592 EVM. Figure 8 shows a schematic of this load. The LC filter converts the square wave power signal coming from the DRV592 into a low-ripple D.C. (zero frequency) signal. In other words, the square wave is composed of an infinite sum of harmonics (i.e., its Fourier series). The LC filters remove all the harmonics except the D.C. component, and it can be shown that the amplitude of the D.C. component is directly proportional to the duty cycle of the PWM signal (see reference [6]). The filters are required because the thermal stresses generated in the Peltier material when applying fast transients (e.g., square wave power) can shorten its life. Peltier element manufacturers typically recommend no more than 10% ripple.

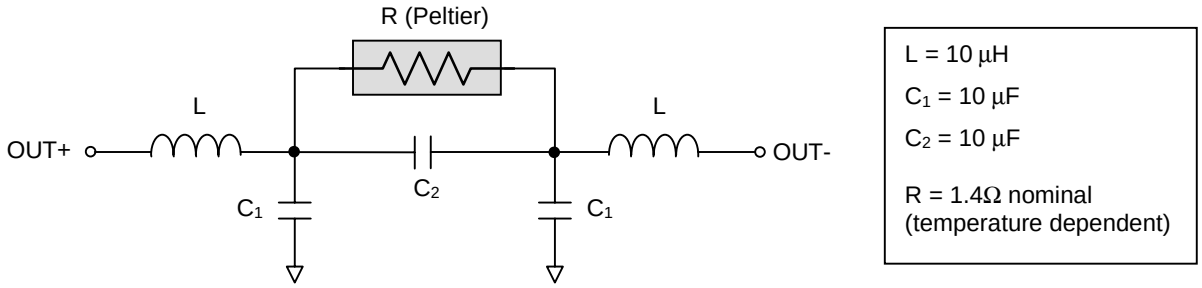


Figure 8. Schematic of LC Filter on the DRV592 EVM

Simple circuit analysis techniques reveal the following transfer function for this system:

$$\frac{V_r(s)}{V_i(s)} = \frac{1}{L(C_1 + 2C_2)s^2 + \frac{2L}{R}s + 1} \tag{11}$$

Where:

$V_r(s)$ is the Laplace transform of the voltage across the resistor (Peltier)
$V_i(s)$ is the Laplace transform of the input voltage, [(OUT+) - (OUT-)]
s is the Laplace Transform operator

Equation 11 can be rearranged into standard form for a second-order linear time-invariant system as follows:

$$\frac{V_r(s)}{V_i(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{12}$$

Where:

$$\omega_n^2 = \frac{1}{LC_1 + 2LC_2} \quad \text{(natural frequency)}$$

$$\zeta = \frac{1}{R}\sqrt{\frac{L}{C_1 + 2C_2}} \quad \text{(damping ratio)}$$

Substituting the parameter values given in Figure 8, the natural frequency and damping ratio of the filter are seen to be:

$\omega_n = 57735 \text{ rad/s} \qquad \zeta = 0.41$

and

$$f_n = \frac{1}{2\pi}\omega_n = 9189 \text{ Hz}$$

Figure 9 shows the (theoretical) frequency response magnitude of the filter. The PWM frequency used in this work was 146 kHz. At this frequency, the filter provides -48.0 dB of attenuation. Simulation of this filter with 146 kHz, 50% duty cycle PWM input (i.e., the worst-case duty cycle) shows 0.49% voltage ripple across the Peltier element.

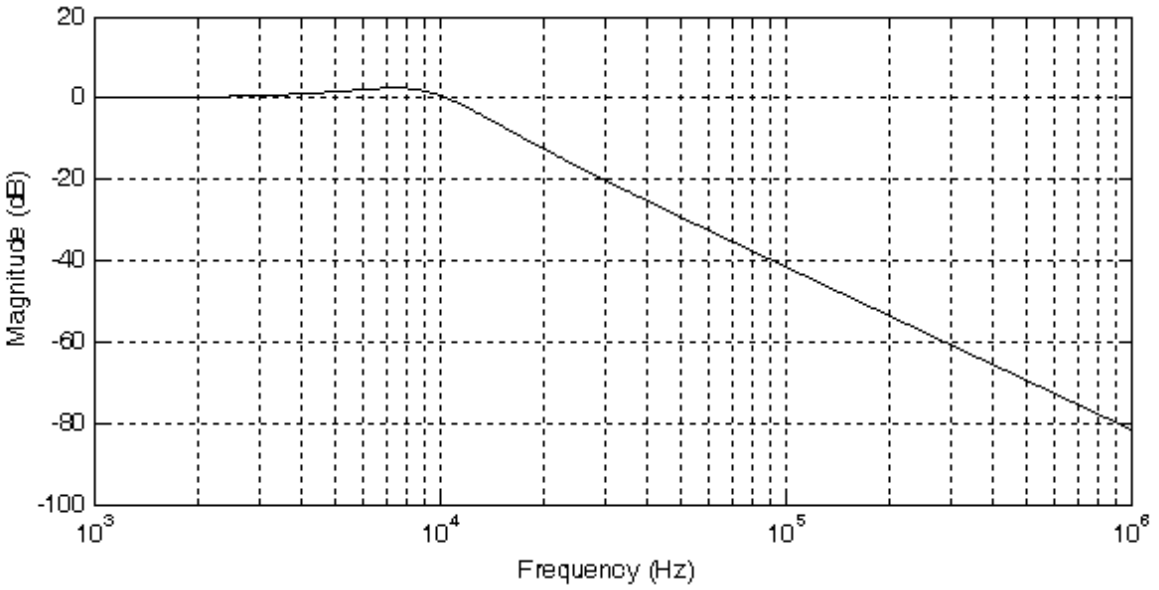


Figure 9. Magnitude Response of the LC Filter on the DRV592 EVM (Theoretical)

3.5 Benchtop Power Supply

The benchtop power supply provides the power to the DRV592 H-bridge, which in turn drives the current through the Peltier element. The system presented here draws about 2.1 Amps from a 3.3-V supply during the largest transients. Any power supply capable of providing this power is suitable. In this work, a *Tektronix PS280 DC Power Supply* was used, with the two 2-Amp outputs configured in parallel, in order to provide up to 4 A at 3.3 V.

4 Software

A code download accompanies this application report. The code is complete and self-contained, and designed to run out-of-the-box on the eZdsp F2812 board. The code makes use of the DSP28 Peripheral Header files for the TMS320F2812 DSP as the programming environment. These header files are available on the Texas Instruments web site. However, the code download contains all needed files, and it is not necessary to separately download the peripheral header files.

4.1 File Descriptions

Root directory:

- *realtime.gel* - Code Composer Studio v2.12 General Extension Language (GEL) file, to simplify use of real time emulation mode
- *tec.pjt* - Code Composer Studio v2.12 project file
- *tec.wks* - Code Composer Studio v2.12 workspace file

/include directory:

NOTE: Most of the include files have been taken directly from the DSP28 peripheral header files v0.58, with no modifications. Many of these files are not needed in this application. However, they are part of the DSP28 header file package, and are all included into the source code by *Device.h*. Such unmodified header files can be readily identified by a filename that begins with *DSP28_*. These files are:

DSP28_Adc.h	DSP28_Gpio.h	DSP28_SWPrioritizedIsrLevels.h
DSP28_CpuTimers.h	DSP28_Mcbsp.h	DSP28_SysCtrl.h
DSP28_DefaultIsr.h	DSP28_PieCtrl.h	DSP28_Xintf.h
DSP28_DevEmu.h	DSP28_PieVect.h	DSP28_Xintrupt.h
DSP28_Ecan.h	DSP28_Sci.h	
DSP28_Ev.h	DSP28_Spi.h	

- *Device.h* - Modified version of DSP28_Device.h from the DSP28 header files v0.58. This file is included in all source files, and includes all other include files into the source file.
- *GlobalPrototypes.h* - Modified version of DSP28_GlobalPrototypes.h from the DSP28 header files v0.58. Contains all function prototypes.
- *Tec.h* - Application-specific variable and constant declarations

/src directory:

- *Adc.c* - ADC initialization routine.
- *DefaultIsr.c* - Contains all interrupt service routines (ISR). The only active ISR is the ADC interrupt service routine. All other ISRs force an emulation halt to trap code bugs during development.
- *DSP28_CodeStartBranch.asm* - Contains the code entry point branched to by the F2812 bootloader
- *Ev.c* - Event manager initialization routine
- *f2812.cmd* - Code Composer Studio linker command file for F2812. The code is linked to internal random access memory (RAM) block H0 (the internal FLASH memory is not used).
- *GlobalVariableDefs.c* - Variable declarations for all global variables, including the DSP28 peripheral header file structures
- *Gpio.c* - General-purpose input/output (GPIO) initialization routine
- *Main.c* - main() function for entire application
- *PieCtrl.c* - PIE control register initialization routine
- *PieVect.c* - Contains all the PIE interrupt vectors and initialization routine
- *SysCtrl.c* - Initialization routine for various system registers
- *TecFunctions.c* - Application-specific functions

4.2 Code Overview

The entire TEC application consists of system initialization, followed by a single interrupt loop (the ADC end-of-conversion interrupt) which implements the PID controller. GP Timer 2 is used to trigger the ADC at a 195.3-Hz sampling rate, while GP Timer 1 is used to clock the PWM output at a 146-kHz switching frequency. The function main() performs CPU and peripheral initialization, and then enters an endless loop that waits for the ADC end-of-conversion interrupt. The controller itself is implemented in the ADC interrupt-service routine in the file *DefaultIsr.c*.

The code is well documented, and should be self-explanatory to the reader. However, a number of key items will be highlighted.

- The DSP is assumed to be running at 150 MHz. All timing calculations (e.g., sample rates, PWM frequencies) have been performed assuming this clock speed.
- The ADC sampling rate can be changed using the constant *sample_period* found in the file *Tec.h*.
- The PWM switching frequency can be changed using the constant *PWM_period* found in the file *Tec.h*. The PWM is of the asymmetric type (as opposed to the symmetric type).
- The thermistor reading obtained from the ADC may be optionally put through a digital low-pass finite impulse response (FIR) filter prior to calling the PID controller. The default configuration of the software uses this filter. To disable the filter, set the constant *FILTER* to 0 in the file *Tec.h*. To enable the filter, set *FILTER* to 1.
- The low-pass digital FIR filter coefficients are defined by the array *aLP[ ]* in the file *GlobalVariableDefs.c*. This is an array of I1Q15 fractional format values. The filter length is defined by the constant NLP in the file *Tec.h*. Both the coefficient values and filter length may be changed by the user, if desired. Note that the low-pass filter should maintain a unity D.C. gain. The default filter is of length 5, with the filter coefficients chosen to have maximum attenuation at the Nyquist frequency.
- The ADC channel used can be selected using the constant *ADC_channel* in the file *Tec.h*. The default configuration uses the ADCINA0 channel.
- The default setpoint for the thermistor voltage is defined by the constant *VREF* in the file *Tec.h*. The default value is 2.0 volts, which corresponds to roughly 28.95°C (see Table B-1 in Appendix B). This can be changed prior to building the code. Alternately, it is generally more convenient to simply change the value of the variable *TEC1.r* in a watch window at runtime using the real time emulation mode of Code Composer Studio. *TEC1.r* is initialized to *VREF* during the initialization portion of the function main().
- All of the variables pertaining to the TEC controller are contained in the structure *TEC1* declared in the file *GlobalVariableDefs.c*. This structure is of type *PID*, as defined in the file *Tec.h*. The PID controller gains, K_p, K'_i, and K'_d, are assigned to the TEC1 structure in the function main() (e.g., *TEC1.Kp*, *TEC1.Ki*, and *TEC1.Kd*). In addition, the total controller output as well as the integral mode controller limit are independently assigned in main(). All of these values may be modified by the user if desired.

- As stated previously, GP Timer 2 is used to trigger ADC conversions of the thermistor reading at a 195.3-Hz rate. However, it was observed during experimentation that sampling the thermistor signal at the same time that the PWM switched either on or off induced noise in the converted thermistor reading. To get around this problem, it is desired to synchronize the triggering of the ADC (i.e., GP Timer 2) with the PWM clock base (i.e., GP Timer 1), and then adjust the sampling instant such that the PWM is not switching at that moment. The code implements this as follows:
  - The PWM switching frequency must be an integer multiple of the ADC sampling rate in order to keep synchronization lock between the two timers. In this case, GP Timer 1 is being clocked at 146 kHz with HSPCLK/1 as the clock source (see the setting for the TPS bit field in the T1CON register, InitPwm() function, file *TecFunctions.c*). This equates to 1024 HSPCLK periods (HSPCLK = high-speed clock = SYSCLKOUT = 150 MHz). GP Timer 2 is being clocked at 195.3 Hz with the HSPCLK/128 as the source (see the setting for the TPS bit field in the T2CON register, InitGpTimer2() function, file *TecFunctions.c*). This equates to 6000 HSPCLK/128 periods, or 6000 * 128 = 768,000 HSPCLK periods. This ratio between 768,000 and 1024 is *exactly* 750. In other words, the ADC will be triggered every 750 PWM periods.
  - The T2CMPR (GP Timer 2 compare) is used to trigger the ADC conversion. This enables the shifting of the ADC trigger point anywhere within the PWM period (as opposed to using the Timer 2 period to trigger the ADC, which would fix the ADC trigger at the end of the Timer 2 period and coincide with the instant when the PWM was switching off). Figure 10 depicts the situation. The compare value of T2CMPR is set to 6 in the InitGpTimer2() function. This triggers the ADC exactly 128 HSPCLK cycles before the end of the active PWM period (128 HSPCLK cycles is equal to 1 Timer 2 tick since Timer 2 is clocked by HSPCLK/128). For small PWM duty cycles (i.e., less than 128/1024 = 12.5%, which occur when the TEC controller has achieved temperature regulation), this compare value triggers the ADC before the PWM switches on. During transients when the PWM duty cycle may be larger than 12.5%, the ADC sample may occur near (or soon after) the switching point, but then noise in the thermistor reading during transients is not really a problem. One really only needs top accuracy in the ADC reading during steady-state regulation. The T2CMPR value of 6 was experimentally determined to give the best results for the system presented under the tested thermal operating conditions. Using a value less than 6 moved the ADC sampling point to the left in Figure 10, and closer to the switch-off point of the previous PWM cycle. This was seen to create more noise in the ADC conversion.

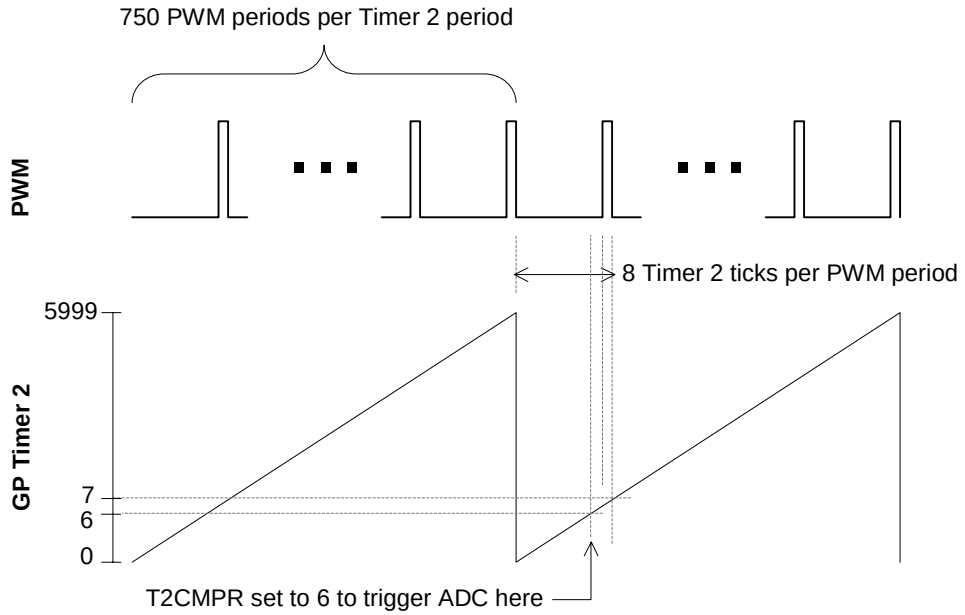


Figure 10. Time Relationship Between PWM Period and ADC Triggering

- The linker command file *f2812.cmd* is set up to link the code to internal RAM block H0. In section 4.3, the needed jumper setting to configure the ROM bootloader for boot-to-H0 mode will be discussed. This allows the code to run straight from a DSP Reset. One could change the linker command file to run the code out of flash and use the boot-to-flash bootloader mode if desired, although this will not be discussed in this application report.
- The code has been tested using C2000 Code Composer Studio v2.12, with the compiler optimizer off, and also with full optimization (level -o3). The default setting in tec.pjt is for the optimizer off. To change this, use the Project→Build_Options menu in Code Composer Studio, select the *Compiler* tab, and change the setting for the *Opt_Level* box.

4.3 Running the Software

The software is designed to run out-of-the-box on the eZdsp F2812 board. In addition, the real time debugger capability of the Code Composer Studio debugger has been enabled in the code, and a Code Composer Studio workspace has been provided. To get started, one could use the procedure outlined below.

1. With the power to the eZdsp F2812 board off, configure the board jumpers as follows (Note: Jumper settings correspond to a revision B board):
   - JP1: 2-3 (XMP/MCn pin, MC mode selected)
   - JP2: 1-2 (flash voltage supplied)
   - JP7: 2-3 (boot mode, boot-to-H0 RAM)
   - JP8: 2-3 (boot mode, boot-to-H0 RAM)

- JP9: 1-2 (PLL enabled)
- JP11: 1-2 (boot mode, boot-to-H0 RAM)
- JP12: 2-3 (boot mode, boot-to-H0 RAM)
- P9.17-18 present (connect VREFLO to Analog GND)
- Note that the last connection, P9.17-18, must be manually made (no jumper actually exists on the board). This connection is critical or the ADC will not function properly.

2. Connect the system as shown in Figure 3 (note that the shielding and twisted pair wires of the ADC input signal are important for noise reduction). Power up the eZdsp board. Leave the bench power supply off.

3. Start Code Composer Studio.

4. Load the workspace file *tec.wks* using File→Workspace→Load_Workspace. Note that you may get a message saying that the workspace was saved using a different driver than what you are currently using. This is because you may not be using the same emulation configuration that the author used when the workspace was created. Just click OK to load as much of the workspace as possible. Then, adjust the graphics windows and watch windows as needed for your PC screen. Figure 11 shows how the screen might look after running the code in the later instruction steps.

5. Open the project file *tec.pjt* using Project→Open. Fix any file load errors that may occur due to path differences between the author's Code Composer Studio installation and yours by removing the problematic file from the project and then adding it back in with the proper directory path for your Code Composer Studio installation. In particular, such a problem may occur with the *rts2800_ml.lib*, and is easy to correct.

6. Build the code using Project→Build.

7. Load the built code using File→Load_Program.

8. Reset the DSP using Debug→Reset_CPU.

9. Disconnect the bench power supply outputs from the DRV592 EVM, and confirm that the voltage is set for 3.3 V (or whatever the desired voltage is). Power off the bench supply, reconnect to the DRV592 EVM, and then turn the supply back on.

   CAUTION:

   It is recommended to skip step 9 the first time through the procedure, and leave the bench power supply off. Complete the remainder of the steps. Get familiar with starting and stopping the code properly, and make sure that the Code Composer Studio screen is updating and no errors are encountered. Then, you can start over from step 8 (resetting the DSP), this time turning the power on to the benchtop power supply. This cautious approach will ensure you do not accidentally overload your Peltier element with current.

10. Enable the real time emulation mode by checking Debug→Real-time_Mode. Click Yes when asked, *Do you want to allow real time mode switching?* At this point, the graphic windows will start updating, although the application is not yet running.

11. Run the code using Debug→Run. The code is now running, and the graphic window outputs are valid (although the thermistor output will be railed until the temperature is brought inside the active range of the instrumentation circuit). Figure 11 shows how the Code Composer Studio screen might look once the temperature regulation settles in. The watch window and two graphs are labeled in the figure. For the graphs, the vertical axes are the variable magnitudes in volts, whereas the horizontal axis is time in seconds.

12. To shut down the code, it is important to reset the DSP immediately upon halting the code. Resetting the DSP puts the PWM outputs into the high-impedance state, thereby shutting off the inputs to the DRV592 H-bridge and stopping current flow to the Peltier element (Note in Figure 3 that the PWM1 and PWM2 outputs from the eZdsp board are pulled to ground through 4.7 kΩ resistors.) The following 3 steps should be performed in relatively quick succession:
    - Halt the CPU: Debug→Halt.
    - Disable Real-time mode: Uncheck Debug→Real-time mode.
    - Reset the DSP: Debug→Reset_CPU.

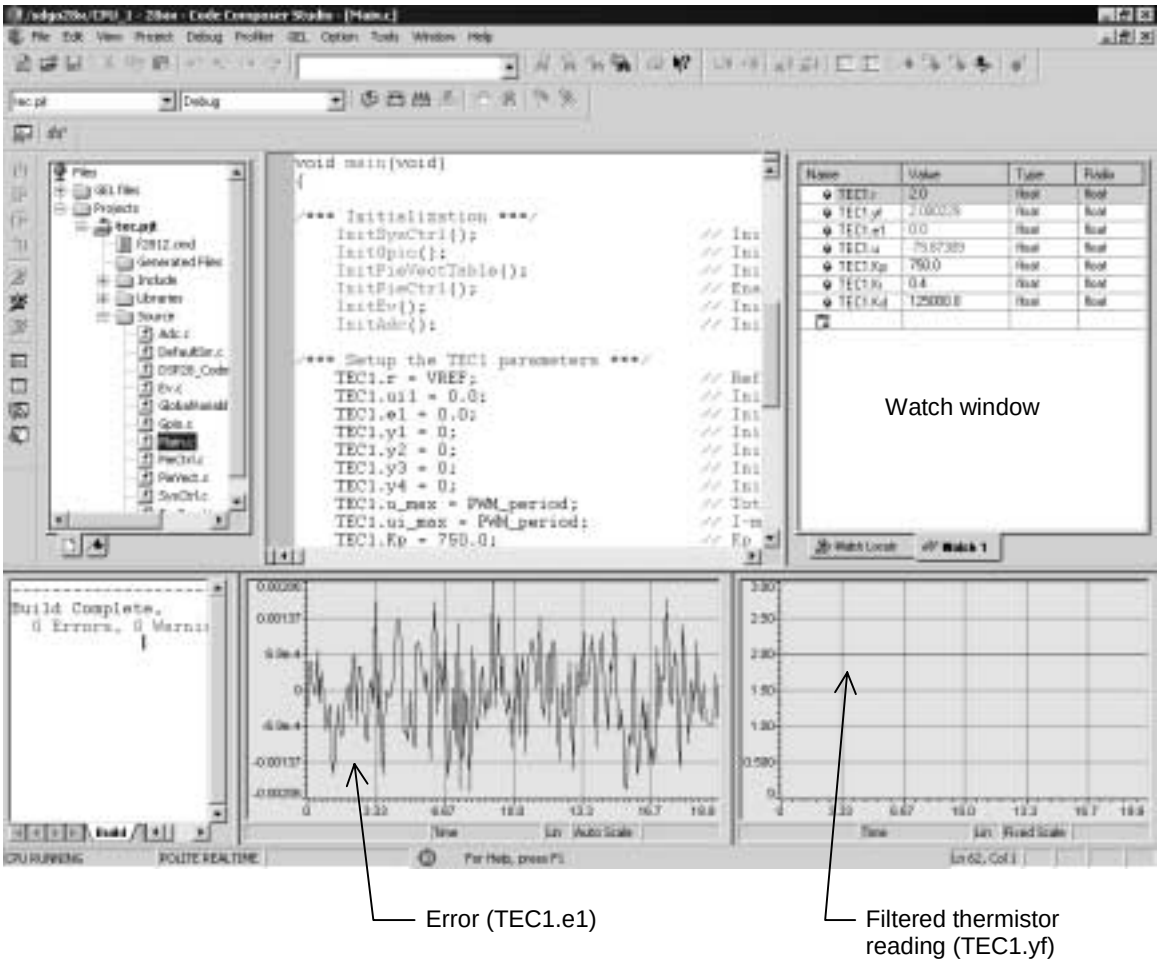


Figure 11. Code Composer Studio Screen After Loading the Workspace *tec.wks* and Running the Code

4.4 Using the GEL File *realtime.gel*

A GEL file has been included to simplify steps 10, 11, and 12 in section 4.3. GEL is a scripting language that allows one to write batch files for Code Composer Studio. Open the file *realtime.gel* with any text editor, and follow the directions given in the notes section of the file header. This will install the GEL file in Code Composer Studio. Once installed, steps 10 and 11 may be replaced by simply clicking the Run_Realtime_with_Reset() function on the GEL menu in Code Composer Studio. Similarly, step 12 may be replaced by clicking the Full_Halt_with_Reset() function on the GEL menu.

5 Controller Performance

The PID controller was tuned by hand to achieve good steady-state regulation and transient step response performance. The controller gains are easily changed while the DSP is running using the real time emulation feature of Code Composer Studio. Results for two different control gain sets will now be presented.

NOTE: All result plots reflect the thermistor output after filtering through the 5-tap, digital low-pass FIR filter, as discussed in section 4.2. Data points for all plots have been taken at a 10-Hz rate using the real time emulation feature of the Code Composer Studio debugger.

5.1 Case 1: $K_p = 750$, $K'_i = 0.4$, $K'_d = 125000$

Figure 12 shows the steady-state regulation error to be approximately ±2 mV. The 12-bit ADC in the F2812 DSP has a 3.0-V full-scale range, which translates into 0.73 mV/LSB. Therefore, the control system is achieving roughly ±3 LSB of accuracy. The reference voltage of 2.0 V is seen from Table B-1 to correspond to approximately 28.96°C. In addition, the local gain of the thermistor instrumentation system at this reference point can be determined from Table B-1 to be -3.36 V/°C (from Table B-1: (2.020 V - 1.936 V)/(28.950°C - 28.975°C) = -3.36 V/°C). Therefore, ±2 mV of error corresponds to ±0.0006°C of regulation accuracy.

It should be stated that the regulation results presented in Figure 12 do not actually reflect measured temperature error or accuracy. The temperature has not actually been measured to be regulating to within the ±0.0006°C stated above. Rather, the voltage output from the instrumentation amplifier, after digital filtering, has been seen to regulate to within ±2 mV. The actual temperature of the laser mount no doubt fluctuates at a much lower frequency and amplitude than shown in Figure 12 due to thermal mass low-pass filtering effects. In addition, a thermal gradient may exist in the laser mount such that not all locations in the mount are at the same temperature. The TEC system controls the temperature only at the exact location of the thermistor within the laser mount.

It should also be noted that the regulation accuracy in terms of temperature is related to the instrumentation circuit gain, which in turn determines the operating temperature range of the system. In this particular work, the instrumentation amplifier gain is such that the operating temperature range of the system is only 1°C. Reducing the instrumentation amplifier gain will increase the operating range, but will reduce the absolute temperature accuracy of the system. In other words, the accuracy of the system is noise limited. If the instrumentation amplifier gain were reduced by a factor of 10, the operating temperature range would increase to 10°C. However, the regulation accuracy would likely still be ±2 mV, which now would translate into ±0.006°C.

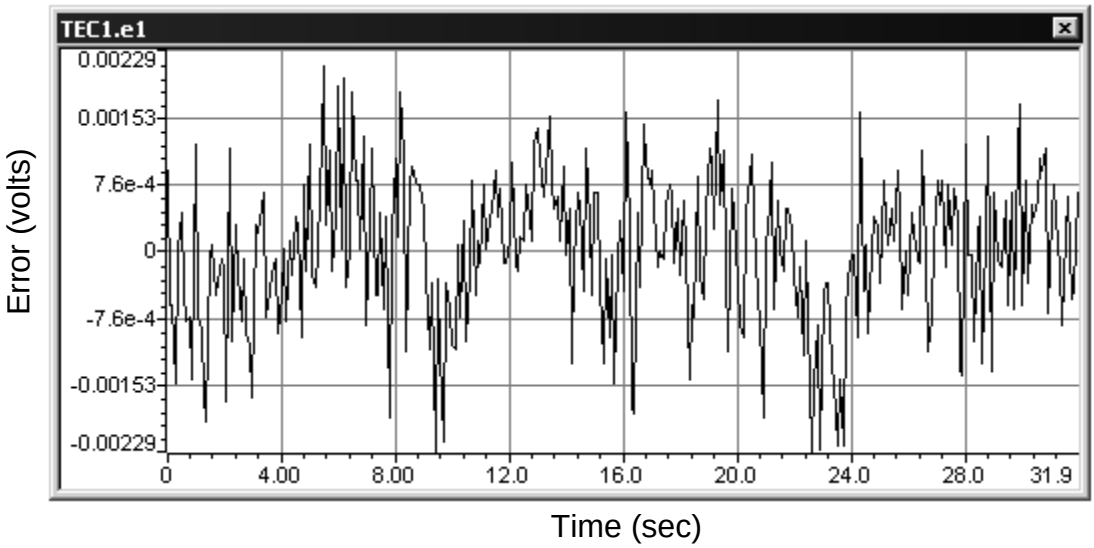


Figure 12. Regulation Error With 2.0-V Reference (K_p = 750, K'_i = 0.4, K'_d = 125000)

Figure 13 shows the error step response to a 0.1-V amplitude step change in the reference voltage (i.e., stepped from 2.0 V to 2.1 V). The response is seen to be underdamped with the PID gains utilized, resulting in an overshoot of 60% of the step size with a rise-time to zero-error crossing of about 3.2 seconds. Settling time to within a ±3-mV window (i.e., ±3% of the step change) is seen to be roughly 14 seconds.

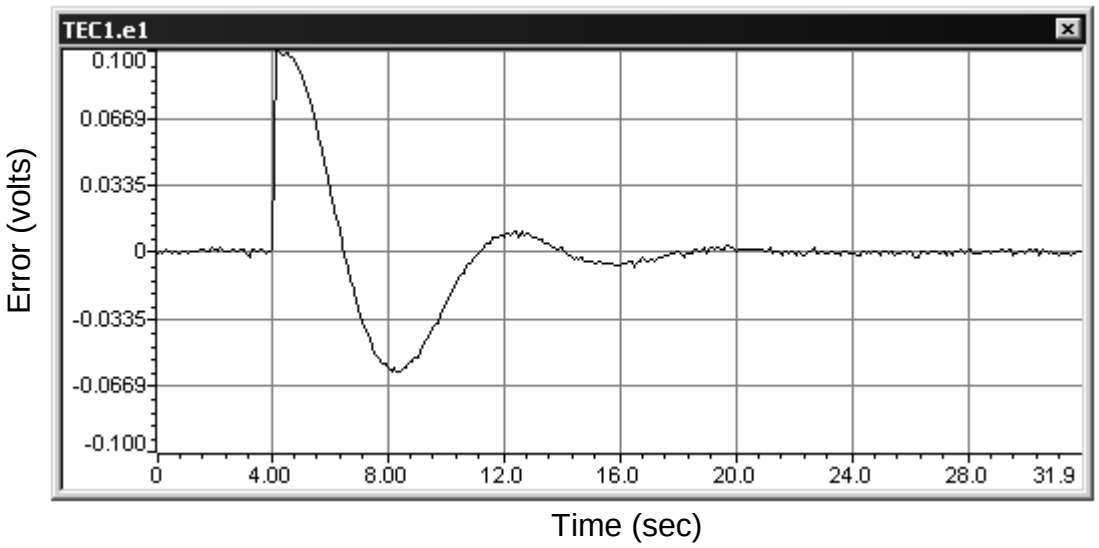


Figure 13. Error Response to a 0.1-V Reference Step at t = 4 seconds (K_p = 750, K'_i = 0.4, K'_d = 125000)

5.2 Case 2: $K_p = 750$, $K'_i = 0.3$, $K'_d = 300000$

Compared with case 1, this case has the I-mode gain reduced by 25%, and the derivative (damping) gain increased by more than two-fold. Based on this, one would expect a more highly damped closed-loop step response and a longer settling time.

Figure 14 shows the steady-state regulation error to fall within approximately a ±2.5 mV window. This is somewhat more error than in case 1, which is not surprising. The derivative mode of the controller is computing a first-difference derivative on the thermistor output signal. This type of computation is fundamentally problematic for noisy signals, such as the thermistor output. Given the higher K_d gain than in case 1, it is to be expected that the steady-state regulation error would contain more noise. In terms of the actual temperature of the laser mount, it is in all likelihood not seeing any of this noise due to the low-pass filtering effects of its thermal mass. Regulation performance is still quite acceptable.

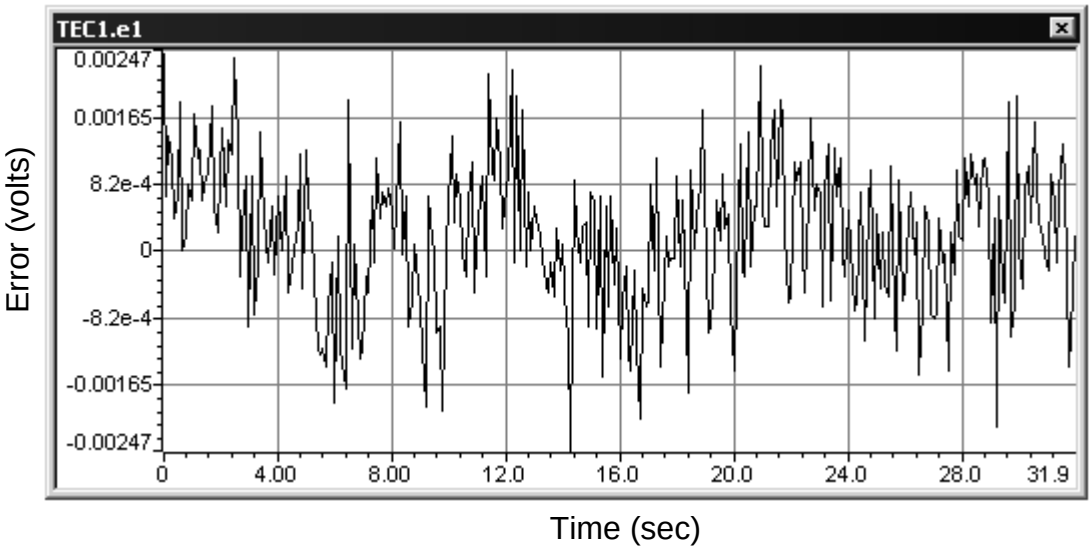


Figure 14. Regulation Error With 2.0-V Reference ($K_p = 750$, $K'_i = 0.3$, $K'_d = 300000$)

Figure 15 shows the error step response to a 0.1-V amplitude step change in the reference voltage (i.e., stepped from 2.0 V to 2.1 V). The response is more highly damped than in case 1. Overshoot has been reduced to 25% of the step size, at the expense of a longer zero-crossing rise-time of about 4.7 seconds and ±3% window settling time of roughly 16.5 seconds.

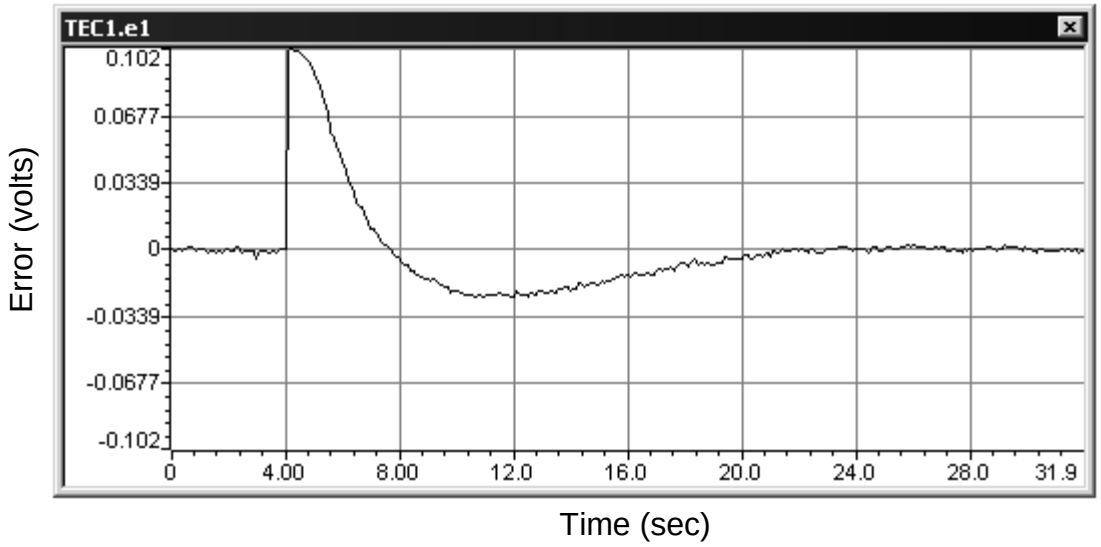


Figure 15. Error Response to a 0.1-V Reference Step at t = 4 seconds (K_p = 750, K'_i = 0.3, K'_d = 300000)

5.3 DSP MIPS Requirements

The controller (i.e., the ADC interrupt service routine) requires about 0.5 MIPS on the F2812 DSP The compiler optimization level made little difference. This value is for the previously stated 195.3-Hz loop rate and includes the 5-tap digital FIR low-pass filtering of the thermistor reading. The 0.5-MIPS value was determined by experimentally measuring the execution time of the ADC interrupt service routine with an oscilloscope, using a pulsed GPIO pin to mark the beginning and end of the routine. The pulse width measured 16 µs, and the routine occurs 195.3 times per second, for a total of 0.00312 seconds of computation per second. Multiplying by the 150-MHz clock rate of the F2812 DSP (this is the actual clock rate used), the 0.5 MIPS value results.

Recall that the PID controller function itself has been written using floating-point C code. Since the F2812 is a fixed-point processor, it executes floating-point C operations using calls to functions in the runtime support library (e.g., *rts2800_ml.lib*). The computational burden of these support functions is considerable, and the MIPS requirement number could be reduced to practically zero if one were to rewrite the floating-point controller code using fixed-point techniques. However, a 0.5 MIPS requirement is quite small on an absolute scale, compared to the 150 MIPS available on an F2812 DSP. One still has nearly all of the processing power of the F2812 DSP available for other parts of an application. Given this reality, there appears to be little advantage at this point in rewriting the controller using fixed-point techniques.

6 Conclusion

A thermoelectric cooler system has been successfully implemented using a TMS320F2812 DSP and a DRV592 power amplifier. The system provides temperature regulation to within ±0.0006°C, and a ±3% settling time of approximately 14 to 16 seconds to step changes in the temperature setpoint, depending on the controller gains. The controller loop was seen to require only 0.5 MIPS out of the 150 MIPS available on the F2812 DSP, leaving nearly all of the DSP computing horsepower available for other application tasks. In addition, the system is nearly all digital, with the exception of the instrumentation circuit needed to read the thermistor. This offers an advantage in noisy environments over traditional analog component TEC systems.

7 References

1. Astrom, K. J. , and Wittenmark, B., “Computer Controlled Systems - Theory and Design,” Prentice-Hall, Inc., Englewood Cliffs, NJ., 1984, ISBN 0-13-164319-3, pp.180-187.
2. Kuo, B. C., “Automatic Control Systems,” Prentice-Hall, Inc., Englewood Cliffs, N.J., 1987, 5th ed., ISBN 0-13-054842-1, pp. 467-479.
3. *DRV592 ±3-A High-Efficiency H-Bridge Data Sheet* (SLOS390).
4. *DRV592 H-Bridge Evaluation Module User’s Guide* (SLOU122).
5. *INA122 Single Supply, MicroPower Instrumentation Amplifier Data Sheet* (SBOS069).
6. *Using PWM Output as a Digital-to-Analog Converter on a TMS320C240 DSP* (SPRA490).

Appendix A TCLMD9 Thermistor Technical Data

Table A-1. TCLMD9 Thermistor and Thermistor Model Data

Temperature (°C)	Documented Resistance (Ω)	Resistance from Steinhart-Hart Model (Ω)	Model Error (%)
-50	692700	818612	18.18
-45	485500	557638	14.86
-40	344700	386169	12.03
-35	247800	271583	9.60
-30	180100	193783	7.60
-25	132400	140164	5.86
-20	98320	102687	4.44
-15	73720	76142	3.29
-10	55790	57105	2.36
-5	42580	43289	1.67
0	32770	33151	1.16
5	25460	25631	0.67
10	19930	19998	0.34
15	15730	15738	0.05
20	12500	12487	-0.10
25	10000	9985	-0.15
30	8055	8043	-0.14
35	6528	6525	-0.05
40	5323	5329	0.10
45	4365	4379	0.33
50	3599	3621	0.62
55	2983	3012	0.96
60	2486	2519	1.31
65	2082	2117	1.70
70	1753	1789	2.06
75	1482	1519	2.51
80	1258	1296	3.01
85	1073	1110	3.48
90	918.9	955	3.97
95	789.9	825	4.50
100	681.6	716	5.05
105	590.6	623	5.55
110	513.4	545	6.10
115	447.9	478	6.64
120	392.0	420	7.20
125	344.1	371	7.79
130	303.0	328	8.37
135	267.6	292	8.97
140	236.9	260	9.62
145	210.4	232	10.22
150	187.3	208	10.87

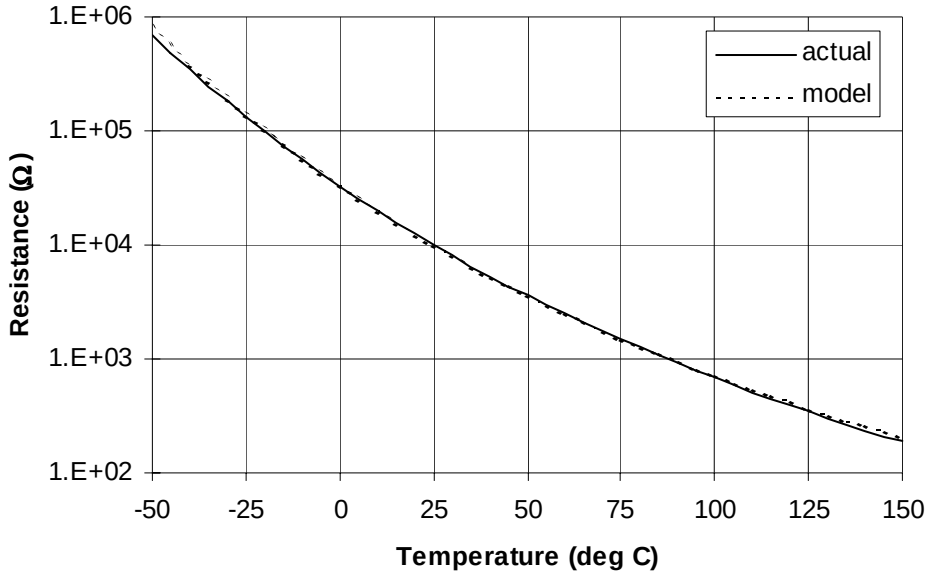


Figure A-1. Thermistor Resistance vs. Temperature

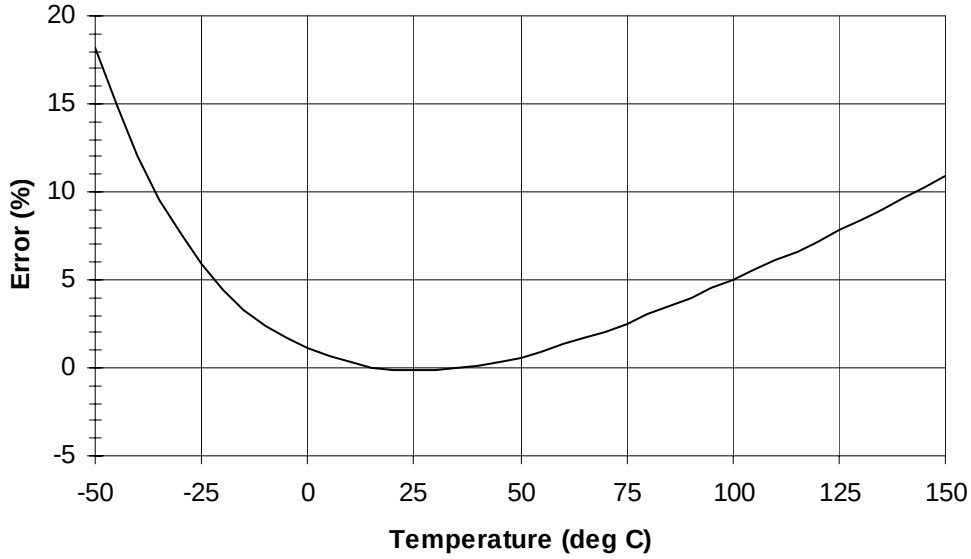


Figure A-2. Thermistor Model Error vs. Temperature

Appendix B Instrumentation Circuit Raw Data

Table B-1. Instrumentation Circuit Input-Output Data (Theoretical)

Temperature (°C)	Temperature (K)	R4 (Ω)	Bridge Vout (Volts)	INA122P Vo (Volts)	Digital ADC Value
28.575	301.73	8548	0.0343	3.292	4096
28.600	301.75	8539	0.0334	3.208	4096
28.625	301.78	8530	0.0326	3.123	4096
28.650	301.80	8521	0.0317	3.038	4096
28.675	301.83	8512	0.0308	2.953	4032
28.700	301.85	8503	0.0299	2.868	3916
28.725	301.88	8494	0.0290	2.783	3800
28.750	301.90	8484	0.0281	2.698	3684
28.775	301.93	8475	0.0273	2.614	3568
28.800	301.95	8466	0.0264	2.529	3453
28.825	301.98	8457	0.0255	2.444	3337
28.850	302.00	8448	0.0246	2.359	3221
28.875	302.03	8439	0.0237	2.275	3105
28.900	302.05	8430	0.0228	2.190	2990
28.925	302.08	8421	0.0219	2.105	2874
28.950	302.10	8412	0.0211	2.020	2758
28.975	302.13	8403	0.0202	1.936	2643
29.000	302.15	8394	0.0193	1.851	2527
29.025	302.18	8385	0.0184	1.766	2412
29.050	302.20	8376	0.0175	1.682	2296
29.075	302.23	8367	0.0167	1.597	2180
29.100	302.25	8358	0.0158	1.512	2065
29.125	302.28	8349	0.0149	1.428	1949
29.150	302.30	8340	0.0140	1.343	1834
29.175	302.33	8331	0.0131	1.258	1718
29.200	302.35	8323	0.0122	1.174	1603
29.225	302.38	8314	0.0114	1.089	1487
29.250	302.40	8305	0.0105	1.005	1372
29.275	302.43	8296	0.0096	0.920	1256
29.300	302.45	8287	0.0087	0.836	1141
29.325	302.48	8278	0.0078	0.751	1026
29.350	302.50	8269	0.0070	0.667	910
29.375	302.53	8261	0.0061	0.582	795
29.400	302.55	8252	0.0052	0.498	679
29.425	302.58	8243	0.0043	0.413	564
29.450	302.60	8234	0.0034	0.329	449
29.475	302.63	8225	0.0025	0.244	333
29.500	302.65	8217	0.0017	0.160	218
29.525	302.68	8208	0.0008	0.075	103
29.550	302.70	8199	0.0000	0.000	0
29.575	302.73	8190	0.0000	0.000	0
29.600	302.75	8182	0.0000	0.000	0

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third–party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Mailing Address:

Texas Instruments
Post Office Box 655303
Dallas, Texas 75265