Keywords: thermoelectric cooler, TEC, optical microcontroller, laser, thermal-loop, current-loop, dual loop control, digital signal processing, DSP, H-bridge, PWM, ADC, DAC, C discrete time, z-transform

APPLICATION NOTE 5425

Thermoelectric Cooler Control Using the DS4830A Optical Microcontroller

Abstract: This application note first briefly discusses the basic operation theory of a thermoelectric cooler (TEC) and its application in optical modules. Then it presents a digital approach to TEC control based on the DS4830A optical microcontroller. Mathematical analysis, algorithm implementation, firmware flowcharts, coding tips as well as an example code are included to make this article a step-by-step guide for TEC control using the DS4830A. Accuracy of ±0.1°C is readily achievable with TEC devices used in typical optical modules.

Introduction

This article begins with a brief discussion of the basic operational theory of a thermoelectric cooler (TEC) and its application in optical modules. It then presents a digital approach for TEC control that uses an optical microcontroller. Mathematical analysis, algorithm implementation, firmware flowcharts, coding tips, and an example code are included to make this article a step-by-step guide for TEC control. The appendices include a digital filter coefficient calculator lab results and an example code.

Part I: Background Information

Principle of Operation for a TEC

A thermoelectric cooler (TEC) is a device based on the Peltier effect. It typically comprises two kinds of materials and transfers heat from one side of the device to the other while a DC current is forced through it. The side from which heat is removed becomes cold; the side to which heat is moved becomes hot. When the current reverses its direction, the previously "cold" side becomes hot and the previously "hot" side becomes cold.

A TEC has no moving parts or working fluids, so it is very reliable and can be very small in size. TECs are used in many applications that require precision temperature control, including optical modules.

There are two main reasons why an optical module may need precision temperature control.

1) The laser needs to be cooled or heated to maintain its optical performance.
2) The laser needs to be set at a specific wavelength.

A well-controlled temperature is also required in dense wavelength division multiplexing (DWDM) for accurate channel spacing. Although multiple lasers can drive a fiber at the same time to achieve large multichannel data rates, the laser wavelength needs to be tightly controlled to ensure correct channel spacing. Since the laser wavelength is temperature-dependent, the temperature must be well-controlled. Consequently, temperature control is an important task for many optical applications. TECs are widely used in such applications because of their small sizes and ease of use.

Part II: Mathematics Behind TEC Control

TEC Control Overview

Importance of a Dual-Loop Control

A TEC is typically used as a heating or cooling element to control the temperature of a certain device, such as a laser module. To achieve good performance, dual closed-loop (the thermal loop and the current loop) control is implemented. *Figure 1* is a simplified system block diagram showing the basic idea of TEC control.

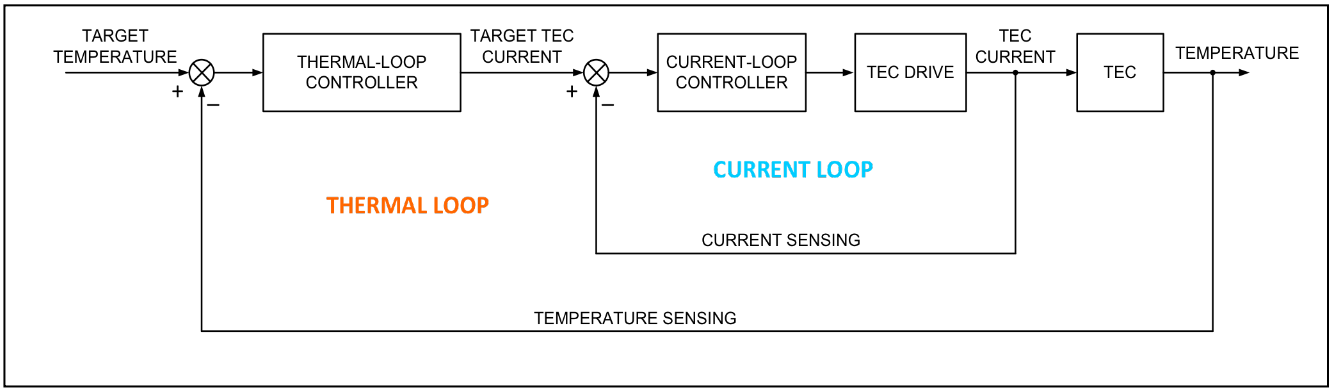


Figure 1. Simplified TEC control block diagram.

The control loops in a typical laser application basically work as follows. First, a target temperature for the laser module is set according to application requirements. A temperature-sensing device, often a thermistor, senses the actual module temperature. The difference of the target temperature and the actual temperature is the temperature error. A thermal-loop controller processes this temperature error. The output of the thermal-loop controller is the target TEC current. Similar to the thermal loop, a current-sensing component senses the TEC current and compares it against the target TEC current. The difference is the current error. Next, the current error is supplied to the current-loop controller. The current-loop controller regulates the TEC drive circuitry to keep the actual TEC current close to the target value. By judiciously designing the controllers and the TEC drive circuitry, high-performance TEC control can be achieved.

Conventional Control Strategies and Implementation

Traditional TEC control strategies available on the market use analog devices, such as analog TEC controller/drivers and operational amplifiers (op amps), to realize the control logic. Although these circuits are well established, they have some drawbacks.

1) The analog implementation usually requires many components, which in turn requires more PCB area. Having more components also renders higher failure rates.

2) In analog approaches, the control thresholds and coefficients are set by discrete components. To achieve high control performance, components with tight tolerances need to be used which, in turn, increases cost. These components are also subject to drift over time which can affect TEC performance.

3) From a development point of view, it is not easy to modify a developed circuit to work for a new application. The component values are interdependent and one must change a number of components to make a modification.

Digital TEC Control Using the DS4830A Optical Controller

The DS4830A is a 16-bit low-power microcontroller with the necessary resources to make high-performance digital TEC control possible. The microcontroller needs to have several integrated capabilities:

- 13-bit analog-to-digital converter (ADC) with 26-input mux
- Eight-channel independent 12-bit digital-to-analog converter (DAC)
- 10 pulse-width modulation (PWM) channels with up to 16-bit resolution
- 32k words of flash memory and 2k words of SRAM
- Single-cycle multiply-accumulate unit (MAC) with a 48-bit accumulator

Unlike analog TEC controllers, the DS4830A optical microcontroller implements the control logic using digital signal processing (DSP) with firmware. This reduces the number of components needed and also makes the control parameters more accurate and repeatable. In addition, it is easier to modify firmware to accommodate new applications than to change discrete components. *Figure 2* is a system block diagram that illustrates the digital TEC control approach.

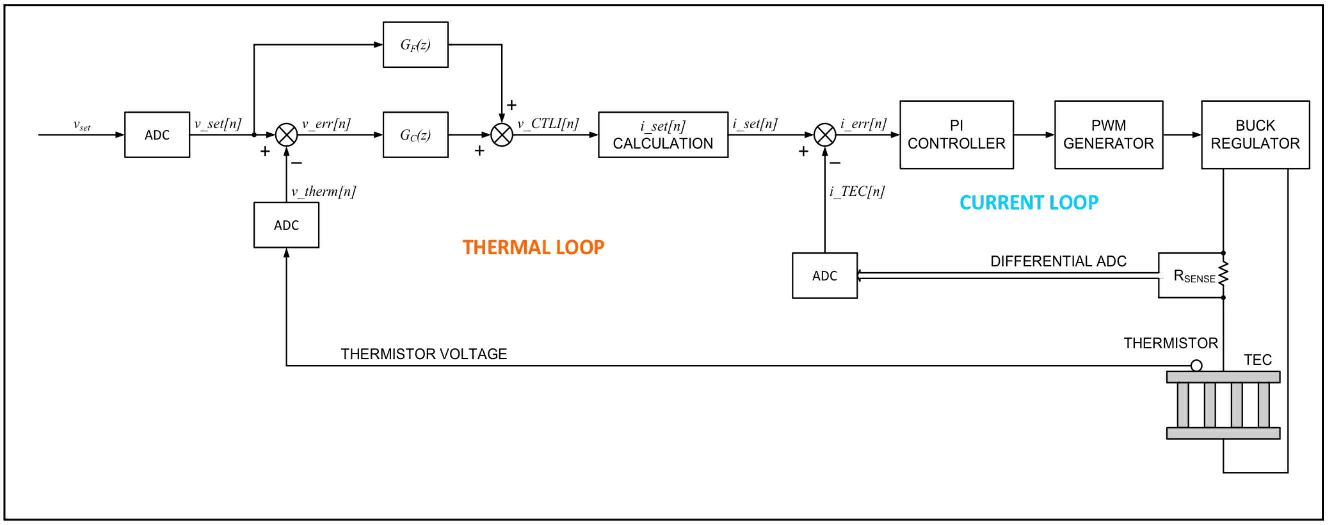


Figure 2. DS4830A TEC control block diagram.

There are some important observations to make from this TEC control approach:

1) Temperature is converted to and represented by a voltage. Controlling the thermistor voltage is virtually controlling the laser module's temperature.

2) All the input signals are digitized before being processed. The set-point voltage and the thermistor voltage are converted using single-ended channels; the TEC current and voltage are converted using differential channels for better performance.

3) Both the thermal-loop controller and the current-loop controller in *Figure 2* are implemented digitally, which reduces the number of components used.

4) The thermal loop and current loop have different update periods. The thermal-loop update period is usually a multiple of the current-loop update period. This will be discussed in more detail later.

In the next section, we will discuss the principle and operation of the thermal-loop controller.

DS4830A TEC Control: Operation of the Thermal Loop

The main function of the thermal-loop controller is to process the temperature error, i.e., the difference of the target temperature and the actual temperature of the laser module, and generate a target TEC current. This is illustrated in *Figure 1*. More detail on this is shown in *Figure 2*.

As explained earlier, temperature is converted to and represented by a voltage. In particular, the target temperature of the laser module is represented by the set-point voltage, which is denoted by V_{SET}. The thermistor is typically placed very close to the laser module so that the temperature of the thermistor is virtually the same as that of the laser module. Again, the thermistor temperature is represented by the voltage across the thermistor, V_{THERM}. Controlling the temperature of the laser module is equivalent to controlling V_{THERM}. Since the thermal-loop controller is digitally implemented, both the set-point voltage and the thermistor voltage are digitized by ADC.

Analog Prototype for the Thermal-Loop Controller

A pragmatic way of developing a digital controller is to find the digital equivalent of a working analog controller. There are many analog control circuits available for TEC control applications, and *Figure 3* is one of them. The circuit in *Figure 3* takes V_{SET} and V_{THERM} and generates V_{CTLI}. The circuit is a proportional integral derivative (PID) controller. Its principle of operation is detailed in application note 3318, "HFAN-08.2.0: How to Control and Compensate a Thermoelectric Cooler (TEC)." The strategy here is to first find the continuous time transfer function of the circuit in *Figure 3* and then converting the transfer function to the discrete time.

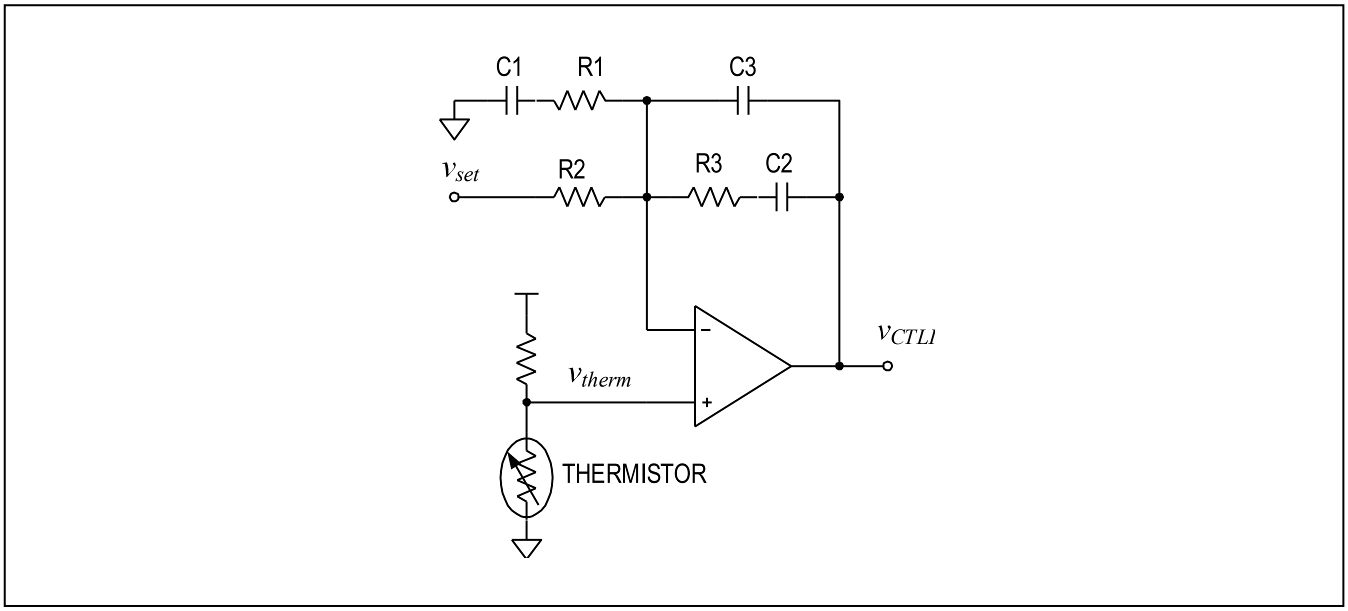


Figure 3. An analog PID controller circuit.

Based on the circuit shown in *Figure 3*, V_{CTLI} can be expressed in terms of V_{SET} and V_{THERM}. Since complex impedances are involved, it is convenient to introduce the complex variable *s = j(2πf) = jω*. Thus, the impedance of a capacitor C1 is $\frac{1}{sC1}$. Taking advantage of basic op amp properties and some algebraic manipulations, we have the expression for V_{CTLI} in s-domain as follows:

$$v_{CTLI}(s) = G_C(s)v_{err}(s) + G_F(s)v_{set}(s) \qquad \text{(Eq. 1)}$$

Where:

$$v_{err}(s) = v_{set}(s) - v_{therm}(s) \qquad \text{(Eq. 2)}$$

$$G_C(s) = -\frac{(1+sR3C2)(1+sR2C1+sR1C1)}{s(C2+C3)(1+sR3C3)R2(1+sR1C1)} - 1 \qquad \text{(Eq. 3)}$$

$$G_F(s) = \frac{C1(1+sR3C2)}{(C2+C3)(1+sR3C3)(1+sR1C1)} + 1 \qquad \text{(Eq. 4)}$$

Define:

$$v_1(s) = G_C(s)v_{err}(s) \qquad \text{(Eq. 5)}$$

$$v_2(s) = G_F(s)v_{set}(s) \qquad \text{(Eq. 6)}$$

We can then rewrite $v_{CTLI}(s)$ as:

$$v_{CTLI}(s) = v_1(s) + v_2(s) \qquad \text{(Eq. 7)}$$

Now V_{CTLI}(s)can be seen as the sum of the outputs of two single-input single-output (SISO) systems. Each system can be characterized by its transfer function, i.e., $G_C(s)$ and $G_F(s)$, respectively. The strategy here is to convert these two continuous time transfer functions to ther discrete time, so they can be implemented by firmware. We use $G_C(z)$ and $G_F(z)$ to denote the discrete time domain equivalent of $G_C(s)$ and $G_F(s)$, respectively. This idea is illustrated in *Figure 2*.

Bilinear Transformation

There are several ways to convert analog systems to digital. Among them, bilinear transformation is widely used due to its property of preserving stability.

The bilinear transformation is a stability preserving mapping from the continuous domain to the discrete domain. In the following discussions, the time interval is denoted by the sampling interval *T* (in seconds). The bilinear transformation is implemented by a change of variables which transforms on an s-domain transfer function to a z-domain transfer function. The usual form is:

$$s = \frac{2}{T}\frac{z-1}{z+1} \qquad \text{(Eq. 8)}$$

The *z*-transform transfer functions of the digital filters $G_C(z)$ and $G_F(z)$ are obtained from the corresponding s-domain transfer functions $G_C(s)$ and $G_F(s)$ by substituting for *s* the bilinear form of Equation 8, respectively.

Obtaining Difference Equations for $v_1[n]$ and $v_2[n]$

We obtain the analytical expressions for $G_C(z)$ and $G_F(z)$ by plugging Equation 8 into Equations 3 and 4, respectively. This is a straightforward process, but it requires numerous algebraic operations. Their final analytical expressions in terms of R-C values are given in Appendix A.

Finally, after inverse z-transform, discrete time signals $v_1[n]$ and $v_2[n]$ are obtained as shown below.

$$v_1[n] = -\sum_{i=1}^{3} A_i v_1[n-i] + \sum_{j=0}^{3} B_j v\_err[n-j] \qquad \text{(Eq. 9)}$$

$$v_2[n] = -\sum_{k=1}^{2} C_k v_2[n-k] + \sum_{l=0}^{2} D_l v\_set[n-l] \qquad \text{(Eq. 10)}$$

Since this is a linear system, the superimposition property applies. We have:

$$v\_CTLI[n] = v_1[n] + v_2[n] \qquad \text{(Eq. 11)}$$

The coefficients in Equations 9 and 10 can be calculated using the Excel® digital filter coefficient calculator spreadsheet, Digital_Filter_Coeff_Cal (for more information, see Appendix B). Simply tweak the resistor and capacitor values at the top of the spreadsheet and the updated coefficients will be generated automatically at the bottom (in the yellow and purple cells).

Target TEC Current Calculation

Once we have $v\_CTLI[n]$, we can set the estimated target TEC current using a simple linear approximation. If we use $i\_set[n]$ to denote the target TEC current, we have:

$$i\_set[n] = \frac{v\_CTLI[n] - 1.5}{10 \times R_{SENSE}} \quad \text{(Eq. 12)}$$

R_{SENSE} is the current-sensing resistor shown in ***Figure 4***.

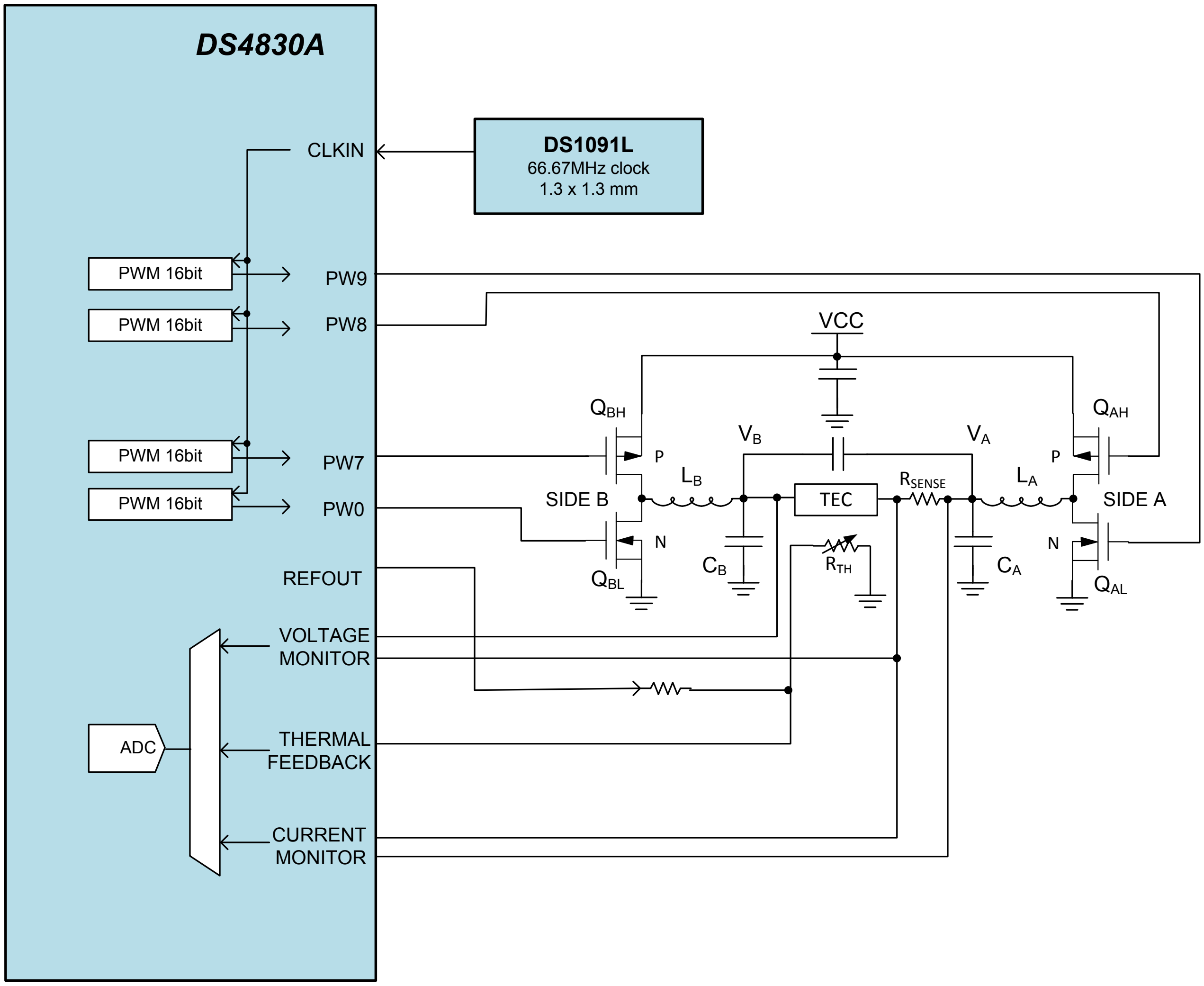


Figure 4. DS4830A TEC H-bridge drive block diagram.

The flow of the thermal-loop control can be summarized as follows. First, the target temperature is set according to system requirements. This temperature is represented by digital voltage, $v\_set[n]$. Then the thermistor furnishes the actual module temperature in the form of voltage, $v\_therm[n]$. Next, the temperature error, $v\_err[n]$, is calculated by finding the difference of $v\_set[n]$ and $v\_therm[n]$. $v\_CTLI[n]$ is then obtained by using Equations 9, 10, and 11. Finally, the target TEC current, $i\_set[n]$, is calculated using Equation 12 and furnished as the input to the current loop.

DS4830A TEC Control: Operation of the Current Loop

As we have described, TEC control comprises two loops: the thermal loop and the current loop. The thermal loop is the outer loop, and it generates the input to the current loop. The current loop is the inner loop, which is shown in *Figure 2*. The main function of the current loop is to regulate the TEC current to the target TEC current set by the thermal loop.

TEC Drive Circuitry

To facilitate description, we will review the TEC drive circuitry. *Figure 4* is a circuit diagram implementing the TEC control, shown in *Figure 2* and *Figure 3*.

In many applications, the TEC is required to provide both heating and cooling, but not at the same time, of course. This means that the drive circuit must be able to force current through the TEC in either direction. In addition, only a single supply is available in most applications. An H-bridge circuit can be used to drive a TEC with a single supply.

Figure 4 shows a simplified diagram for the H-bridge that is used to drive the TEC. The H-bridge comprises four MOSFETs, which are driven by four independent PWM signals generated by the DS4830A. PW8 and PW9 drive MOSFETs Q_{AH} and Q_{AL}, respectively, and this side of the bridge is called "Side A." Similarly, PW7 and PW0 drive MOSFETs Q_{BH} and Q_{BL} on Side B, respectively. The MOSFETs on each side of the bridge, along with the corresponding inductor and capacitor, form a buck converter. The two buck converters will be not be active at the same time. Either one buck converter is active and another one is grounded or none of them is active. As the TEC current demand increases/decrease, one output will increase/decrease and the other will be grounded. Since the TEC's volt-amp characteristic is largely resistive under normal operating conditions, we can regulate the TEC current by controlling the operating duty cycle of each buck converter.

Figure 2 shows how the current loop works. First, the target TEC current, $i\_set[n]$, is furnished by the thermal control loop. Then a current-sensing resistor samples the TEC current, $i\_TEC[n]$. The difference of $i\_set[n]$ and $i\_TEC[n]$ is calculated and denoted as $i\_err[n]$. Next, $i\_err[n]$ is furnished to a proportional integral (PI) filter. The output of the PI filter is $e\_PI[n]$ and it controls the buck converters' duty cycles. By controlling the buck converter's duty cycles, the current loop is able to regulate the TEC current to $i\_set[n]$.

PI Controller

A simplified current-loop flowchart is illustrated in *Figure 5*.

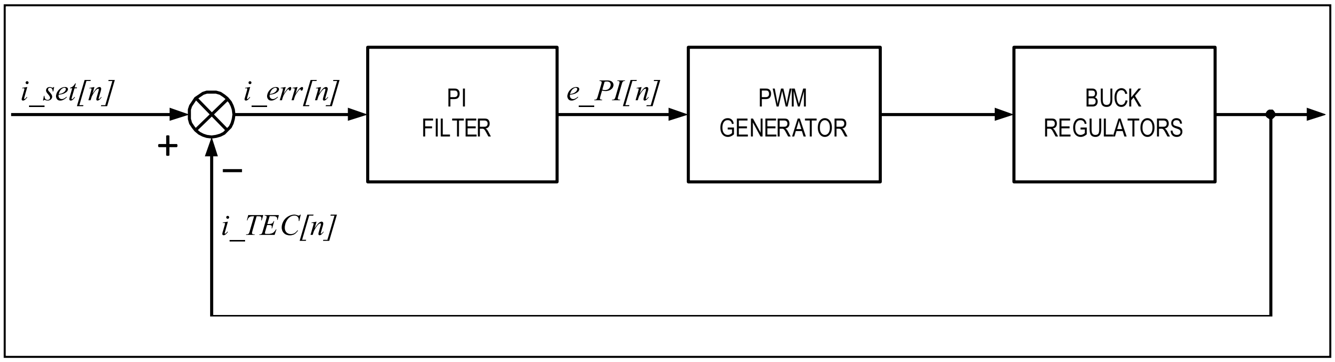


Figure 5. The DS4830A TEC control current-loop diagram.

The basic idea of a generic PI controller is as follows. If $V_i(s)$ denotes the input of a PI controller and $V_o(s)$ denotes the output, the transfer function of the controller can be written as:

$$\frac{V_o(s)}{V_i(s)} = K_P + \frac{K_I}{s} = \frac{K_P s + K_I}{s} \qquad \text{(Eq. 13)}$$

Where K_P is the proportional gain and K_I is the integral gain. The values of K_P and K_I are developed experimentally.

Using the bilinear transformation described earlier, Equation 13 is converted to the discrete domain and the corresponding difference equation has the following form:

$$v_o[n] = -A_c v_o[n-1] + B_{c0} v_i[n] + B_{c1} v_i[n-1] \qquad \text{(Eq. 14)}$$

The coefficients can also be calculated using the Digital_Filter_Coeff_Cal spreadsheet. Note that the sampling period of the current loop is shorter than that of the thermal loop, typically by a factor of 8 or 10. In the example code, the value of the current-loop sampling period (T_C in the spreadsheet) is set to 1.5ms as opposed to 9ms sampling period for the thermal loop.

In our application, the input to the PI filter is $i\_err[n]$ and the output is $e\_PI[n]$. We can then rewrite Equation 14 as:

$$e\_PI[n] = -A_c e\_PI[n-1] + B_{c0} i\_err[n] + B_{c1} i\_err[n-1] \qquad \text{(Eq. 15)}$$

Once we obtain $e\_PI[n]$, we can determine the PWM duty cycles. The following section explains the procedure.

Intelligent Drive, One Buck Converter

The DS4830A selects the drive mode shown in *Figure 6* based on the current-loop output.
Below are a few notes:

1) Only one buck converter is active in mode 1 and 2.
2) Neither buck converter is active in mode 3.
3) Switching losses are cut by 50%.

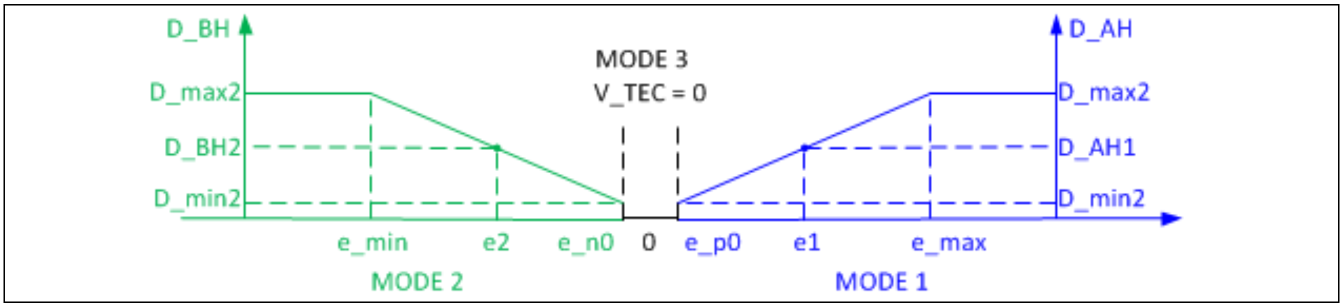


Figure 6. Buck converter mode.

High-Efficiency Drive: Mode 1

From *Figure 7*, we see that:

- Only the buck converter on Side A is active: $V_A = D\_AH$ x Vcc.
- The buck converter on side B ties TEC to GND: V_B = 0.
- The TEC current is regulated by controlling D_AH.
- The switching losses are cut by 50%.

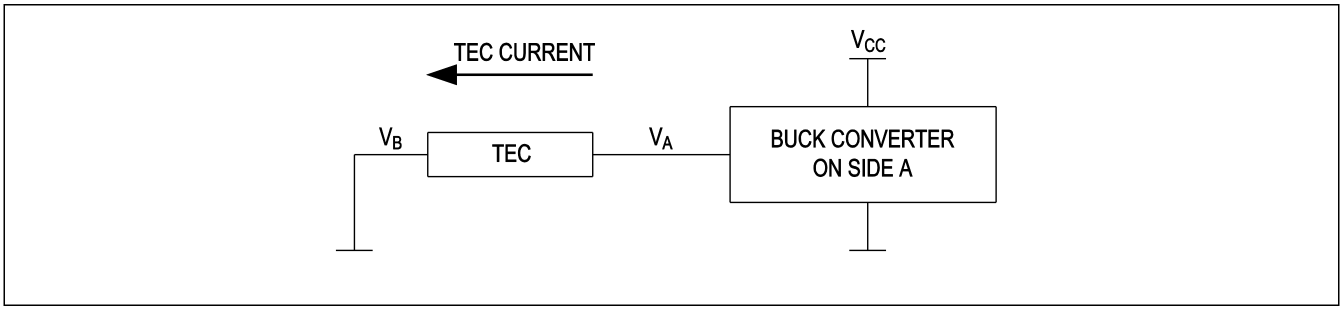


Figure 7. High-efficiency drive Mode 1.

High-Efficiency Drive: Mode 2

From *Figure 8*, we see that:

- Only the buck converter on Side B is active: $V_B = D\_BH$ x Vcc.
- The buck converter on side A ties TEC to GND: V_A = 0V.
- The TEC current is regulated by controlling D_BH.
- The switching losses are cut by 50%.

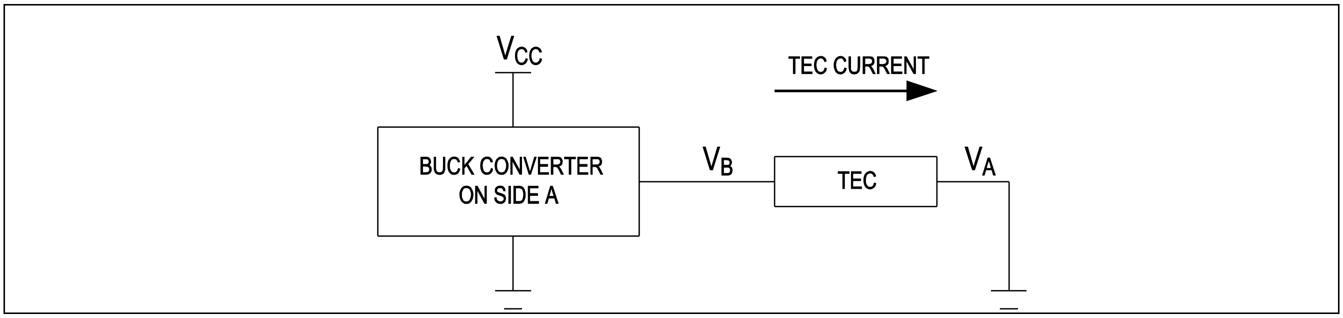


Figure 8. High-efficiency drive Mode 2.

Determine PWM Duty Cycles In Mode 1

We take a closer look at the H-bridge shown in *Figure 4*. The output voltage of the buck converter on Side A, V_A, can be approximately expressed as:

$$V_A = \frac{T_{QAH\_ON}}{T_{SW}} V_{CC} \qquad \text{(Eq. 16)}$$

Where T_{SW} is MOSFET Q_{AH}'s switching period and T_{QAH_ON} is the on-time for Q_{AH}. Here we neglect the "on" drop of Q_{AH} and Q_{AL}. From Equation 16 we can see that at a certain switching frequency, the longer on-time for Q_{AH}, the higher V_A.

The voltage across the TEC is $V_A - V_{RSENSE}$, where V_{RSENSE} is the voltage across the current-sensing resistor, R_{SENSE}. Therefore, by manipulating the "on" times of Q_{AH}, we can control the TEC voltage and thus the TEC current.

As shown in *Figure 4*, there are four MOSFETs and each of them is driven by a PWM signal from the DS4830A. **Table 1** summarizes the PWM signals and the MOSFETs they drive. Note that any PWM channel can be used for either A or B side.

Table 1. PWM Signals Used for TEC Drive

MOSFETs Driven			PWM Drive Signals		MOSFET On-Time/Period
Designator	Topological Location	Type	Signal Name	Duty Cycle	
Q_{AH}	Side A high	P channel	PWM_AH (e.g.: PW8)	D_AH	1 - D_AH
Q_{AL}	Side A low	N channel	PWM_AL (PW9)	D_AL	D_AL
Q_{BH}	Side B high	P channel	PWM_BH (PW7)1	D_BH	1 - D_BH
Q_{BL}	Side B low	N channel	PWM_BL (PW0)	D_BL	D_BL

Note we define duty cycle as the high time over the period. For n-channel MOSFETs Q_{AL} and Q_{BL}, their on-time is equal to the high time of the PWM drive signals. Conversely, for p-channel MOSFETs, their on-time is equal to the low time of the PWM signals. These relationships are summarized in the MOSFET On-Time/Period column in Table 1.

Figure 9 shows the phases and duty cycles of the four PWM signals in Mode 1. The following are a few notes:

1) Both PWM_AH and PWM_AL have the same frequency.
2) PWM_BL is on (turn on nMOS and ground side B) and PWM_BH off.
3) To further prevent shoot-through, a short dead-time is inserted between the high-side "on" stage and the low-side "on" stage.

4) According to *Figure 9*, D_AL can be expressed in terms of D_AH. If we let DT denote the ratio between the dead-time and the PWM period, we have:

$$D\_AL = D\_AH - 2\times DT \qquad \text{(Eq. 17)}$$

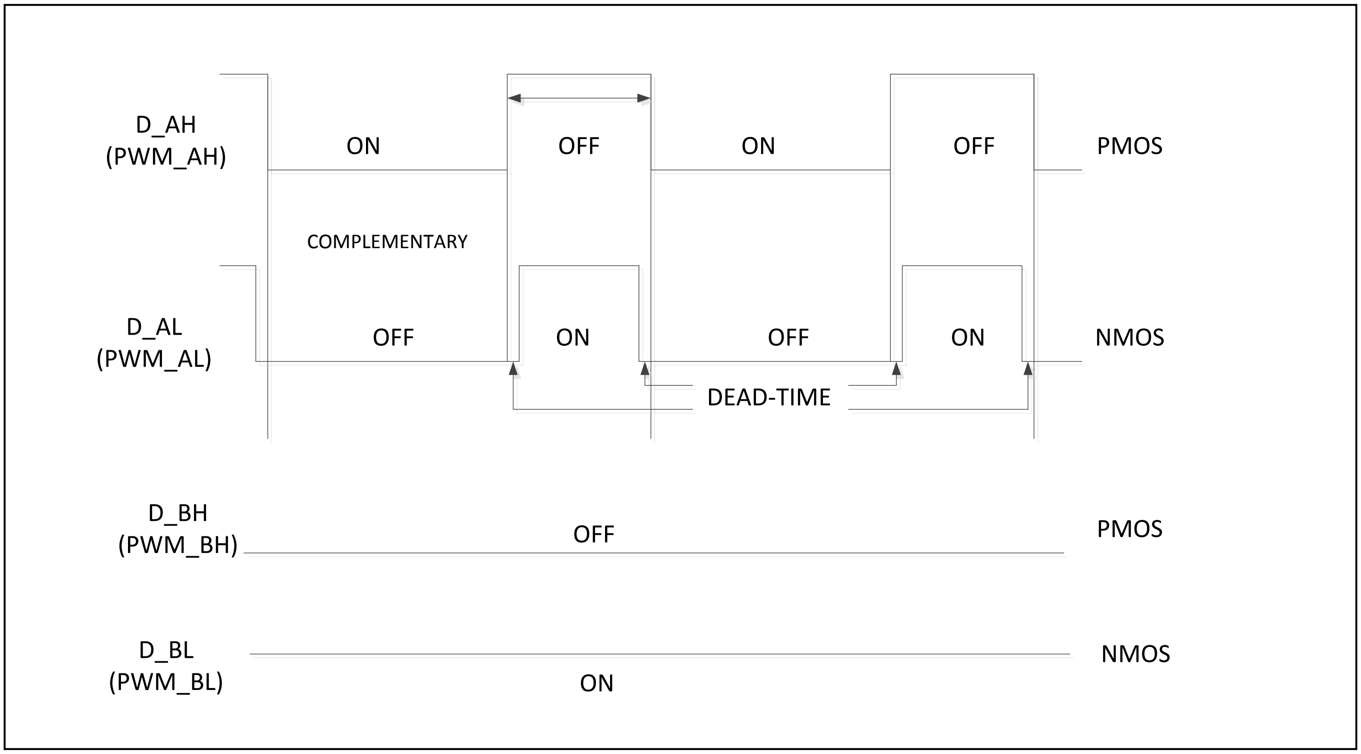


Figure 9. PWM phases and duty cycles in Mode 1.

Before we explain how to determine D_AH, we define the TEC current flowing from Side A to Side B as positive. Similarly, we define the TEC voltage as positive in Mode 1.

Assume for a moment that the TEC current $i\_TEC[n]$ is less than the target TEC current $i\_set[n]$. Then the current error $i\_err[n] = i\_set[n] - i\_TEC[n]$ is greater than zero. Subject to the proportional and integral gains, this positive $i\_err[n]$ will gradually drive the output of the PI controller $e\_PI[n]$ up.

Consequently, as shown in *Figure 4*, V_A needs to increase and V_B needs to be grounded in order for $i\_TEC[n]$ to catch up with $i\_set[n]$. Reviewing Equation 16, the on-time of Q_{AH} needs to increase so V_A can go up.

To summarize the analysis above, a less than target TEC current results in a positive $i\_err[n]$, which drives $e\_PI[n]$ up, and a longer T_{QAH_ON} is needed to boost the TEC current. *Figure 10* illustrates the intuitive relationship between $e\_PI[n]$ and T_{QAH_ON} for a closed-loop TEC current control.

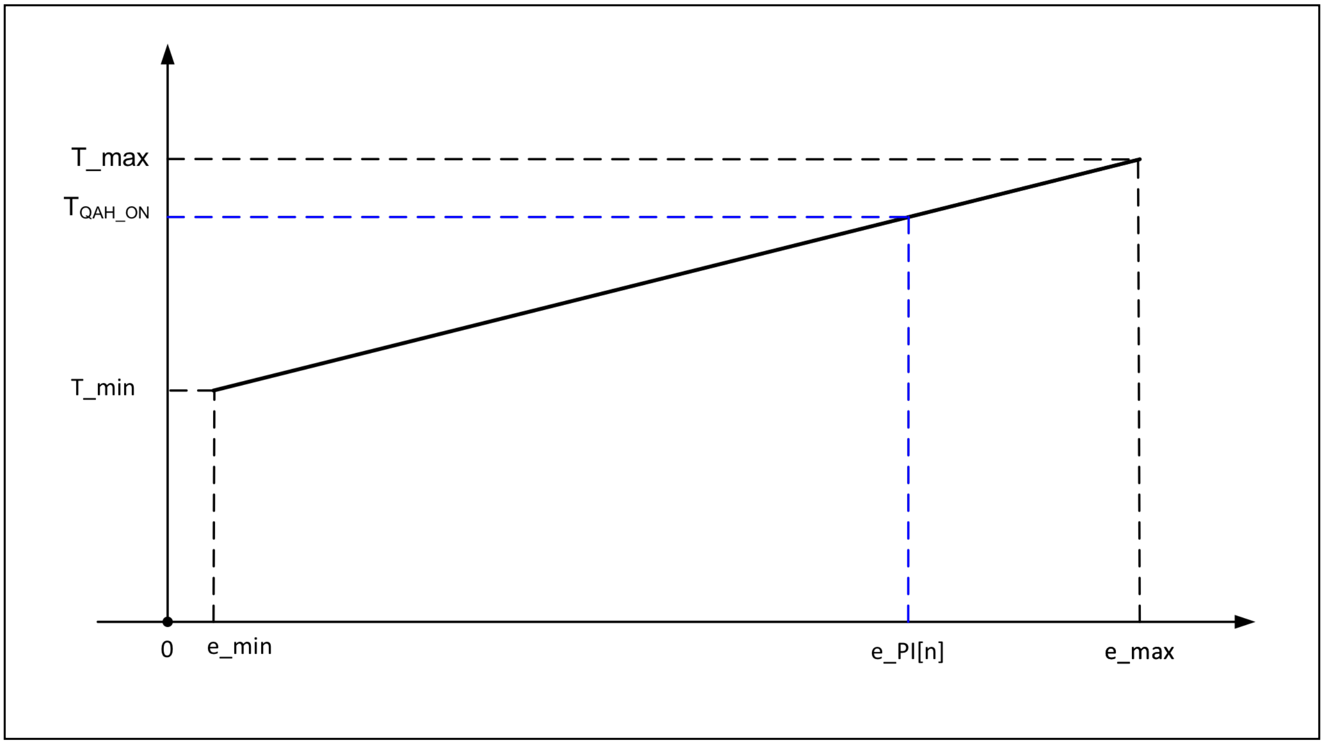


Figure 10. T_{QAH_ON} vs. e_PI[n].

The fundamental relationship is: the larger e_PI[n], the longer T_{QAH_ON}. Note that we have introduced limits (e_max and e_min) for e_PI[n] and T_{QAH_ON} to avoid integral windup and reduce overshooting. Thus, e_PI[n] will be clamped to e_max if it becomes greater than e_max. Similarly, if e_PI[n] goes below e_min, it will be clamped to e_min.

Although the duty cycle of the buck converter can run from 0 to 100%, it is good practice to keep the PWM duty cycles within a certain range. As shown in *Figure 10*, T_min denotes the minimum available on-time for Q_{AH} and it corresponds to e_min. At the other end, T_max denotes the maximum available on-time for Q_{AH} and it corresponds to e_max.

We assume T_{QAH_ON} increases linearly as e_PI[n] increases. Then it is simple math to calculate T_{QAH_ON} according to *Figure 10*.

$$T_{QAH\_ON} = \frac{e\_PI[n] - e\_\min}{e\_\max - e\_\min}(T\_\max - T\_\min) + T\_\min \qquad \text{(Eq. 18)}$$

Once T_{QAH_ON} is determined using Equation 18, D_AH can be calculated using the relationship shown in Table 1 and D_AL can be obtained using Equation 17. Then one can set up the PWM channels and drive the TEC accordingly.

Note: In Mode 2, D_BH and D_BL can be calculated in a similar way.

Summary

This application note detailed how to control a TEC using the DS4830A optical microcontroller. We first converted a prototype analog controller’s transfer function to the discrete domain using bilinear transformation. Then a PI controller was presented for the current loop. All procedures and their implementation using the DS4830A were systematically explained. For further information, lab results and an example code are provided in Appendix C and Appendix D, respectively.

Appendix A: Expressions for Gc(z) and GF(z)

In Part II, we omitted the analytical expressions for $G_C(z)$ and $G_F(z)$ for the sake of succinctness. The expressions are laid out here.

Plugging Equation 8 into Equation 3, we have:

$$G_C(z) = \frac{v_1(z)}{v_{err}(z)} = -\frac{1 + \frac{2}{T}\frac{z-1}{z+1}(\tau_{16} + \tau_{18} + \tau_b) + \frac{4}{T^2}(\frac{z-1}{z+1})^2 \tau_{16}(\tau_{18} + \tau_b)}{(\tau_c + \tau_{17})\frac{2}{T}\frac{z-1}{z+1}[1 + \frac{2}{T}\frac{z-1}{z+1}(\tau_a + \tau_{18}) + \frac{4}{T^2}(\frac{z-1}{z+1})^2 \tau_a \tau_{18}]} - 1 \qquad \text{(Eq. 19)}$$

Where:

$\tau_{16} = R3C2$,
$\tau_{17} = R2C3$,
$\tau_{18} = R1C1$,
$\tau_a = R3C3$,
$\tau_b = R2C1$, and
$\tau_c = R2C2$.

Define the following intermediate variables:

$a_3 = 1 + \frac{2}{T}(\tau_{16} + \tau_{18} + \tau_b) + \frac{4}{T^2}\tau_{16}(\tau_{18} + \tau_b)$,
$a_2 = 3 + \frac{2}{T}(\tau_{16} + \tau_{18} + \tau_b) - \frac{4}{T^2}\tau_{16}(\tau_{18} + \tau_b)$,
$a_1 = 3 - \frac{2}{T}(\tau_{16} + \tau_{18} + \tau_b) - \frac{4}{T^2}\tau_{16}(\tau_{18} + \tau_b)$,
$a_0 = 1 - \frac{2}{T}(\tau_{16} + \tau_{18} + \tau_b) + \frac{4}{T^2}\tau_{16}(\tau_{18} + \tau_b)$,
$b_3 = \frac{2}{T}(\tau_c + \tau_{17}) + \frac{4}{T^2}(\tau_c + \tau_{17})(\tau_a + \tau_{18}) + \frac{8}{T^3}(\tau_c + \tau_{17})\tau_a\tau_{18}$,
$b_2 = \frac{2}{T}(\tau_c + \tau_{17}) - \frac{4}{T^2}(\tau_c + \tau_{17})(\tau_a + \tau_{18}) - \frac{24}{T^3}(\tau_c + \tau_{17})\tau_a\tau_{18}$,
$b_1 = -\frac{2}{T}(\tau_c + \tau_{17}) - \frac{4}{T^2}(\tau_c + \tau_{17})(\tau_a + \tau_{18}) + \frac{24}{T^3}(\tau_c + \tau_{17})\tau_a\tau_{18}$, and
$b_0 = -\frac{2}{T}(\tau_c + \tau_{17}) + \frac{4}{T^2}(\tau_c + \tau_{17})(\tau_a + \tau_{18}) - \frac{8}{T^3}(\tau_c + \tau_{17})\tau_a\tau_{18}$,

Now we can simplify Equation 20 to:

$$G_C(z) = \frac{v_1(z)}{v_{err}(z)} = -\frac{(a_3 + b_3)z^3 + (a_2 + b_2)z^2 + (a_1 + b_1)z + (a_0 + b_0)}{b_3 z^3 + b_2 z^2 + b_1 z + b_0} \qquad \text{(Eq. 20)}$$

After taking the inverse z-transform of Equation 21, we have Equation 9. The associated digital filter coefficients can be determined using the Excel calculator presented in Appendix B.

Similar to the process above, we can obtain the expression for $G_F(z)$. Plugging Equation 8 into Equation 4, we have:

$$G_F(z) = \frac{v_2(z)}{v_{set}(z)} = 1 + \frac{C_1}{C_2 + C_3} \frac{1 + \tau_{16} \frac{2}{T} \frac{z-1}{z+1}}{(1 + \tau_a \frac{2}{T} \frac{z-1}{z+1})(1 + \tau_{18} \frac{2}{T} \frac{z-1}{z+1})} \qquad \text{(Eq. 21)}$$

Define the following intermediate variables:

$c_2 = \frac{C_1}{C_2 + C_3}(1 + \frac{2}{T}\tau_{16})$,

$c_1 = \frac{2C_1}{C_2 + C_3}$,

$c_0 = \frac{C_1}{C_2 + C_3}(1 - \frac{2}{T}\tau_{16})$,

$d_2 = 1 + \frac{2}{T}(\tau_a + \tau_{18}) + \frac{4}{T^2}\tau_a\tau_{18}$,

$d_1 = 2 - \frac{8}{T^2}\tau_a\tau_{18}$, and

$d_0 = 1 - \frac{2}{T}(\tau_a + \tau_{18}) + \frac{4}{T^2}\tau_a\tau_{18}$,

Now we can simplify Equation 22 to:

$$G_F(z) = \frac{v_2(z)}{v_{set}(z)} = \frac{(c_2 + d_2)z^2 + (c_1 + d_1)z + (c_0 + d_0)}{d_2 z^2 + d_1 z + d_0} \qquad \text{(Eq. 22)}$$

After taking the inverse z-transform of Equation 23, we have Equation 10. The associated digital filter coefficients can be determined using the calculator presented in Appendix B.

Appendix B: Calculating Digital Filter Coefficients

We have provided the Digital_Filter_Coeff_Cal spreadsheet that can automatically calculate the digital filter coefficients in Equations 9, 10, and 15.

To determine the coefficients in Equations 9 and 10, one only needs to specify the values of R1, R2, R3, C1, C2, C3 (component functions shown in *Figure 5*), and T (the thermal-loop sampling period) near the top of the calculator. The coefficients (As, Bs, Cs, and Ds) will be generated near the bottom of the file, displayed in purple or yellow cells.

Similarly, one only needs to specify K_p, K_I, and T_c in cells D88 to D90 to calculate the coefficients in Equation 15. This information will be provided in cells D92 to D94.

Appendix C: Lab Results

We ran the example code on a DS4830A Evaluation Kit board to get transient responses and H-bridge efficiency.

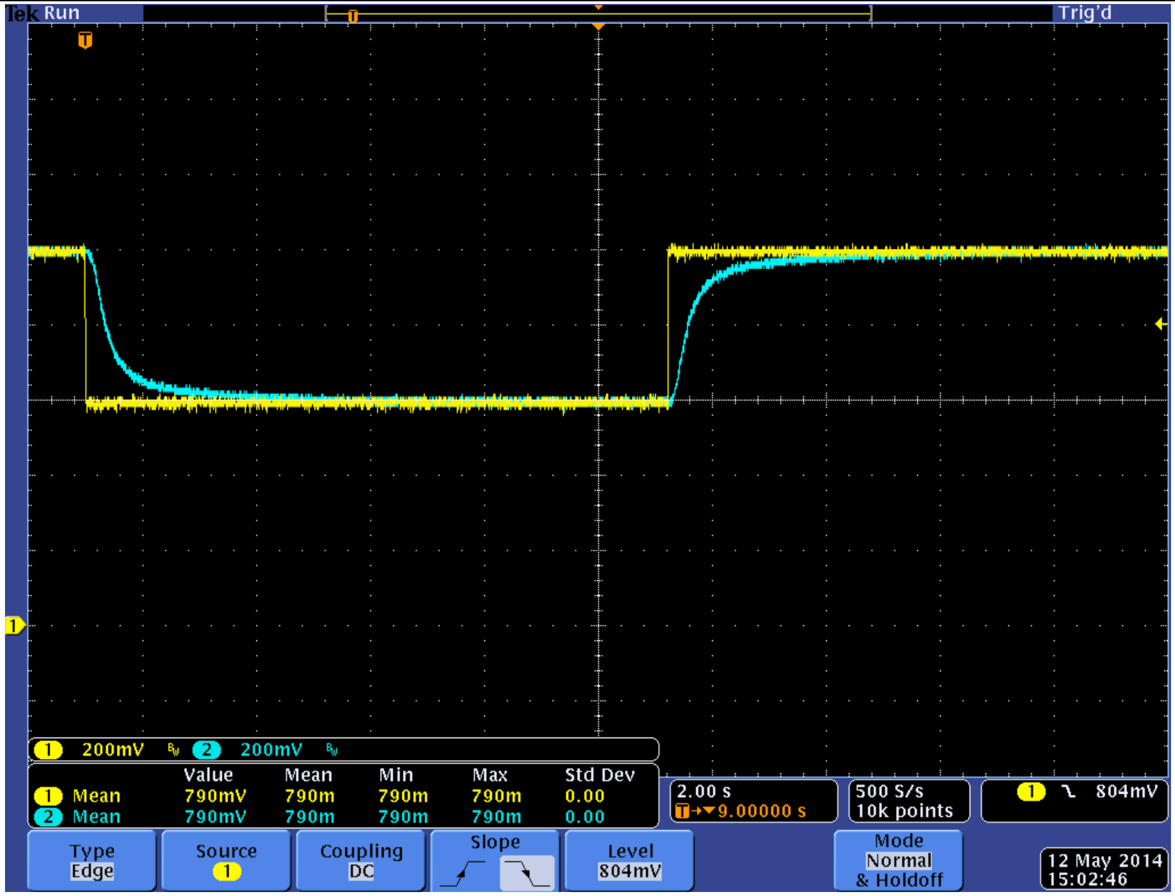
A. Transient Responses

Figure 11 shows the transient responses. The yellow curve is the set-point voltage and the red curve is the thermistor voltage.

The experiment setup and procedures follow:

1) A transmitter optical subassembly (TOSA) mounted on a heatsink. The TOSA has a built-in TEC and thermistor.

2) DC power supplies:
   a. 3.3V for the EV board
   b. Variable DC supply for the set-point voltage
3) Connections:
   a. Connect TEC (+) of the TOSA to the R_TECC pad next to jumper LOADA on the EV kit board.
   b. Connect TEC (-) of the TOSA to the R_TECC pad next to jumper LOADB on the EV kit board.
   c. Connect the thermistor signal of the TOSA to test point TECC_TEMP on the EV kit board.
   d. Connect TOSA ground to EV kit board ground.
   e. Connect the set-point voltage to ADC_D7P and ground to ADC_D7N on the EV kit board.
4) Power-on sequence:
   a. Power on the DS4830A EV kit.
   b. Turn on the set-point voltage with an initial value of 0.75V.
5) Testing the TEC control functionality:
   a. Decrease the set-point voltage to 0.40V and wait for the thermistor voltage to settle.
   b. Increase the set-point voltage back to 0.75V and wait for the thermistor voltage to settle.



```
Tek Run                                                    Trig'd

1  200mV  BW   2  200mV  BW
              Value     Mean    Min     Max     Std Dev
1  Mean       790mV     790m    790m    790m    0.00
2  Mean       790mV     790m    790m    790m    0.00

2.00 s            500 S/s        1  ⤵  804mV
T→▼9.00000 s      10k points

Type     Source    Coupling    Slope    Level      Mode            12 May 2014
Edge     1         DC                   804mV      Normal          15:02:46
                                                   & Holdoff
```

Figure 11 The thermistor voltage as set-point down from 0.75V to 0.40V, then back to 0.75V.

The TOSA used in this experiment has a typical operating temperature of 35°C, which corresponds to a 0.40V thermistor voltage. At room temperature, the thermistor voltage is about 0.75V. We captured the transient response of the thermistor voltage when the set-point voltage changes from 0.75V to 0.40V and then back to 0.75V. The risetime and the falltime are both very short and there is no overshoot. The data is shown in **Table 2** below.

Table 2. Thermistor Voltage Rise Time and Fall Time

TEMPERATURE CHANGE	CORRESPONDING SET-POINT CHANGE	FALL TIME (90% TO 10%)	FALL TIME (95% TO 5%)	RISE TIME (90% TO 10%)	RISE TIME (95% TO 5%)
25°C → 50°C	0.75V → 0.40V	1.5s	1.8s	—	—
50°C → 25°C	0.40V → 0.75V	—	—	1.4s	2.1s

B. Efficiency

Figure 12 shows the H-bridge efficiency.

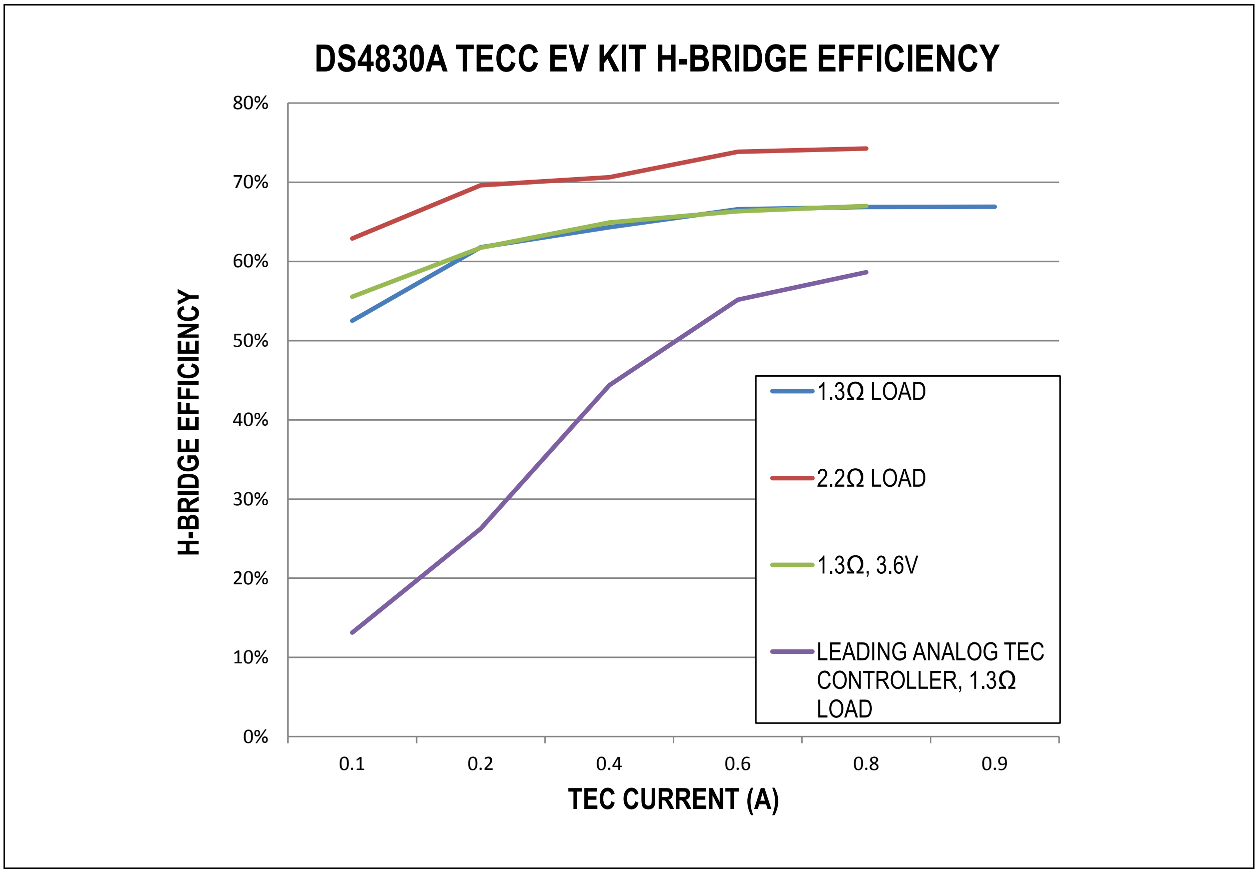


Figure 12. DS4830A TEC EV kit H-bridge efficiency.

The experiment setup and procedures follow:

1) One 3.3 or 3.6V power supply (V_{CC}) for the EV kit board.

2) Disable H-bridge (Remove J8, J9, and J10 jumpers) and measure power-supply output current, I_{CQ}

3) Connect a 1.3Ω or 2.2Ω resistor, R_{LOAD}, as a dummy load on J10 jumper.

4) Set 'gucIsetDebugEnable' to 1 and set 'gfIsetDebug' in firmware to different values (0.1A to 0.8A) and run the code on the EV kit board. Record the power-supply ouput current, I_S.

5) H-bridge input current is: $I_{IN} = I_S - I_{CQ}$

6) Efficiency = (gfIsetDebug x gfIsetDebug x R_{LOAD})/($I_{IN} \times V_{CC}$)

C. Long Term Wavelength and Power Drift

We ran long hours (about 64 hours) wavelength drift and power test. Agilent 86120C Multi-Wavelength Meter was used to measure the result. The wavelength reference is 1535.3 nm and the power one 0.73 db. The total wavelength drift is 12 pm from the reference and the laser power 0.11 db since the drift measurement was started. Values represent the minimum wavelength drift and power drift values subtracted from the maximum drift values. They are excellent results.

Table 3. Long Term Test Results

Test Running Hours	Reference Wavelength	Reference Laser Power	Max. Wavelength Drift	Max Laser Power Drift
64	1535.3 nm	0.73 db	12 pm	0.11 db

The experiment setup and procedures follow:

1) A transmitter optical subassembly (TOSA) mounted on a heatsink. The TOSA has a built-in TEC and thermistor.

2) DC power supplies:
    a. 3.3V for the EV board
    b. A second current source (about 30 mA) for the TOSA I_{bias}

3) Remove the J6 Jumper. Connect the 2nd current source (+) to the pin closer to the DS4830A socket and (-) to the ground of the PCB.

4) Connect a single mode (yellow) optical fiber cable from TOSA to the Optical Input of the Wavelength Meter.

5) Turn on both powers and wait a few minutes for the laser to stabilize. Start the test.

D. Wavelength and Power Drift Under Extreme Temperature

The following table shows the extreme temperature (-40 -> +85°) wavelength drift and power test result. We used Agilent 86120C Multi-Wavelength Meter to measure the result and an oven to change the temperature. The wavelength reference is 1535.385 nm and the power one -0.1 db at 25°. During tests, we kept the oven temperature for at least 30 minutes at -40° and 85°. The total wavelength drift is 24 pm from the reference and the laser power 0.27 db since the drift measurement was started. Values represent the minimum wavelength drift and power drift values subtracted from the maximum drift values. They are excellent results.

Table 4. Extreme Temperature Test Results

Temperature Range	Reference Wavelength at 25°	Reference Laser Power at 25°	Max. Wavelength Drift	Max Laser Power Drift
-40° -> +85°	1535.385 nm	-0.1db	24 pm	0.27

The experiment setup and procedures are as the same as the test for Long Term *Wavelength and Power Drift* except that the PCB and TOSA has to be put inside the Oven and the test also doesn't need to run long hours.

Appendix D: Example Code

For the source code (in C) of our DS4830A TEC driver, please contact Maxim support. It should be noted that our code was customized for the lab setup described in Appendix C, which means that all the parameters, coefficients, and configurations are based on that particular setup. Users should **NOT** try running the code directly on their own TEC control board.