Application Note

Supercapacitor Health Measurement with the BQ25856-Q1

Ethan Galloway, Michael Bradbourne

ABSTRACT

Supercapacitors offer the highest power density compared to regular capacitors or batteries. High power density combined with low ESR means that supercapacitors are becoming more and more widespread. Supercapacitors are being used to provide a power backup for car computers, car doors, or to provide supplement power for servers. However, supercapacitors decrease in capacitance and the equivalent series resistance, or ESR, increases over time. Knowledge of a capacitor's current ESR and capacitance is critical to make sure that these capacitors can function when needed. Together with a microcontroller, the BQ2575X and BQ2585X-Q1 IC families can provide an approximate measurement of the ESR of the supercapacitor and capacitance to help estimate the capacitor's end-of-life, or EOL. These ICs can also help extend the lifespan of a supercapacitor by limiting the charging and discharging current and precisely controlling the output voltage. This application note provides an outline and example code for how to roughly measure the ESR and capacitance of supercapacitors with the BQ25856-Q1 and a MSPM0L2228 MCU.

Table of Contents

Trademarks

All trademarks are the property of their respective owners.

Copyright © 2026 Texas Instruments Incorporated

1 Introduction

Supercapacitors are becoming more and more prevalent as backup systems. Supercapacitors can provide backup power to end equipment such as medical equipment, car doors, and automotive computers. Supercapacitors can also provide supplementary power to servers.

Supercapacitors offer a much longer lifetime and power density compared to battery technologies; however, the impact of aging still needs to be considered. As a supercapacitor ages, the ESR increases and the capacitance decreases. Both of these factors work together to decrease the total usable energy and power in the supercapacitor.

These capacitors need to be reliable and have a long lifetime for their applications. This application note goes over how to quickly measure the capacitance and ESR of a supercapacitor. With the ESR and capacitance information, supercapacitors can be replaced before the capacitor reaches end-of-life (EOL).

Copyright © 2026 Texas Instruments Incorporated

2 How to Extend the Life of Supercapacitors

The functional lifespan of a supercapacitor is dependent on the voltage, charging current, mechanical stress, and operating temperature. Many supercapacitor manufacturers determine their capacitors to have reached EOL when the capacitance decreases to 80% of the nominal value or the equivalent series resistance, or ESR, increases to 200% of the nominal value. Check with the manufacturer of each supercapacitor to determine when the supercapacitor has reached EOL.

This application note primarily focuses on how to measure the ESR and capacitance of a supercapacitor. Nevertheless, here are a few areas that can reduce the overall lifespan of supercapacitors and how the BQ2585X-Q1 can be used to mitigate the decreasing life span.

2.1 Effects of Voltage

Overvolting a supercapacitor decreases the total lifetime of a supercapacitor. Charging the supercapacitor to a lower voltage helps extend the operating lifespan of a supercapacitor. The BQ2575X and the BQ2585X-Q1 charge controllers have a 0.5% voltage accuracy on the feedback resistors to precisely charge the supercapacitors to their recommended full charge voltage.

2.2 Effects of Current

Large amounts of current sinking or sourcing to a supercapacitor will cause internal heating. Sink or source currents exceeding the supercapacitor's specification decrease the lifespan of the device. The BQ2575X and BQ2585X-Q1 family have discharge current limits to limit the total sourcing currents of the supercapacitor through the IBAT_REV register. The BQ2575X and BQ2585X-Q1 family also have charge current settings to accurately regulate the current into the supercapacitor.

2.3 Effects of Temperature

Supercapacitor lifespan decreases significantly with increasing temperature. Please check with the supercapacitor manufacturer for the temperature requirements of the supercapacitor. The BQ2575X and BQ2585X-Q1 are JEITA compliant and can work with a thermistor to reduce the charge current or charge voltage when the supercapacitor gets warm.

Copyright © 2026 Texas Instruments Incorporated

3 Supercapacitor ESR and Capacitance

3.1 Estimating Capacitor Capacitance and ESR

Supercapacitors and capacitors can be modeled to be a resistor and capacitor as shown in Figure 3-1.

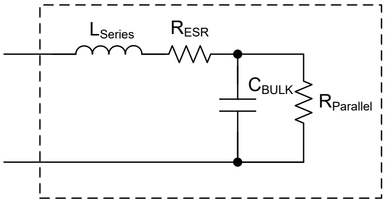


Figure 3-1. Capacitor Model

Since the capacitance and ESR are the defining characteristics of the supercapacitor health and will change over time, this application note focuses on measuring these values.

The defining equation for current in a capacitor is shown in Equation 1:

$$i = C\frac{dV}{dt} \tag{1}$$

The BQ2585X-Q1 and BQ2575X family of parts can act as a constant current source. Using a constant current source means that the change in voltage is linear over time. Equation 1 can be rearranged into Equation 2 to calculate the capacitance:The BQ2585X-Q1 ICs can measure the change in voltage with the on-board ADC. Measure the initial voltage after the start of charging and the final voltage before disabling charging. The change in voltage can be determined by Equation 3:

$$C = \frac{i\Delta t}{\Delta V} \tag{2}$$

$$\Delta V = V_{Final} - V_{Initial} \tag{3}$$

The capacitance can be calculated by Equation 4:

$$C = \frac{i\Delta t}{V_{Final} - V_{Initial}} \tag{4}$$

By measuring the change in voltage over a known time with a known current, the above equation can calculate the capacitance of the supercapacitor. The capacitance can be tracked over time to determine if a capacitor will reach EOL soon.

The ESR also needs to be tracked over time. The voltage drop across the capacitor's ESR is defined by Ohm’s Law and the resistance of the capacitor ESR can be solved for with \:

$$R_{ESR} = \frac{V_{ESR}}{i} \tag{5}$$

While the device is sourcing current to charge the supercapacitor, the voltage measured from the capacitor is the sum of two voltages, the capacitor voltage and the voltage drop across the capacitor's ESR (V_{ESR}). The capacitor can be measured with no sinking or sourcing current to make V_{ESR} cancel out. Here, the final voltage is defined as the voltage right before charging ends. The ending voltage is the voltage on the supercapacitor with no current. The ending voltage has no current; therefore, there is no voltage across the ESR.

Subtracting the two voltage measurements results in the voltage drop across the supercapacitor ESR:

$$V_{Final} - V_{Ending} = \left(V_{ESR} + V_{CAP}\right) - \left(V_{CAP}\right) = V_{ESR} \tag{6}$$

Copyright © 2026 Texas Instruments Incorporated

Plugging in the result from Equation 6 into Equation 5 and knowing the charge current setting can determine the ESR of the supercapacitor. The ESR can also be tracked over the lifetime of the supercapacitor to estimate if the supercapacitor has reached its end of life.

To make sure the current stays constant and the capacitor does not charge to full during the test, the following equation can be used to calculate the ending voltage of the capacitor after the test.

$$V_{Ending} = \frac{1}{C}\int_0^t i \, \mathrm{d}\, t + V_{Start} \quad (7)$$

Since the charge current, *i*, is a constant value, Equation 7 can be rearranged into:

$$V_{Final} = \frac{i\Delta t}{C} + V_{Start} \quad (8)$$

This equation can be used to make sure the BQ2585X-Q1 ICs stay in constant current mode and that the test does not fully charge the capacitor. The BQ2585X-Q1 also has status registers to show if the IC is constant voltage mode or constant current mode. The status registers can be read after the test to make sure that the IC stayed in constant current mode.

The following flow chart shows how to measure the health of a supercapacitor with a BQ25856-Q1:

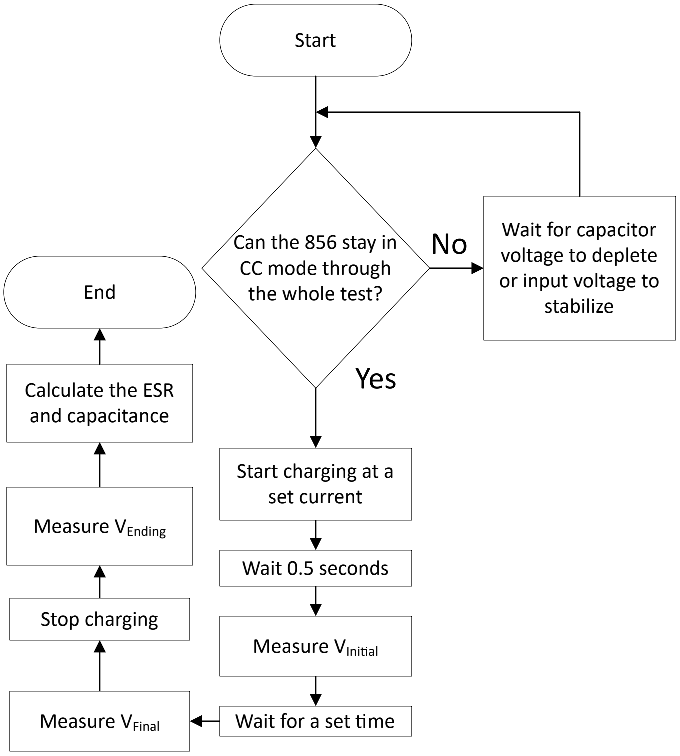


Figure 3-2. Supercapacitor Health Measurement Flow Chart

Copyright © 2026 Texas Instruments Incorporated

3.2 Measuring Supercapacitor Health With The BQ25856-Q1

In the following example, the BQ25856-Q1 IC in conjunction with the MSPM0L2228 MCU measures the capacitance and ESR of eight supercapacitors in series. The BQ25856-Q1 with the MCU host controller can measure the ESR and capacitance with the following steps:

1. Measure the voltage of the capacitor
2. Calculate the total charge time and charge current to ensure the BQ25856-Q1 remains in constant current mode throughout the charging process
3. Start charging the capacitor
4. Measure the $V_{Initial}$ after charging begins
5. Measure the V_{Ending} right before charging ends
6. Stop charging the capacitor
7. Measure V_{Final} when there is no charge current=
8. Calculate the capacitance using $V_{Initial}$, V_{Ending} and the time between the two measurements
9. Calculate the ESR with the charge current, V_{Ending}, and V_{Final}.

The following results are tested under the conditions below:

- V_{AC} = 10V
- I_{CHG} = 1A
- t_{pulse} = 1s
- $C_{OUT\_nominal}$ = 10F x 8S1P
- $R_{ESR\_nominal}$ = 35 mΩ x 8S1P

In Figure 3-3 the BQ25856-Q1 takes approximately three seconds while charging the capacitor to measure the capacitance and ESR. The measurements can be completed in a shorter time if desired.

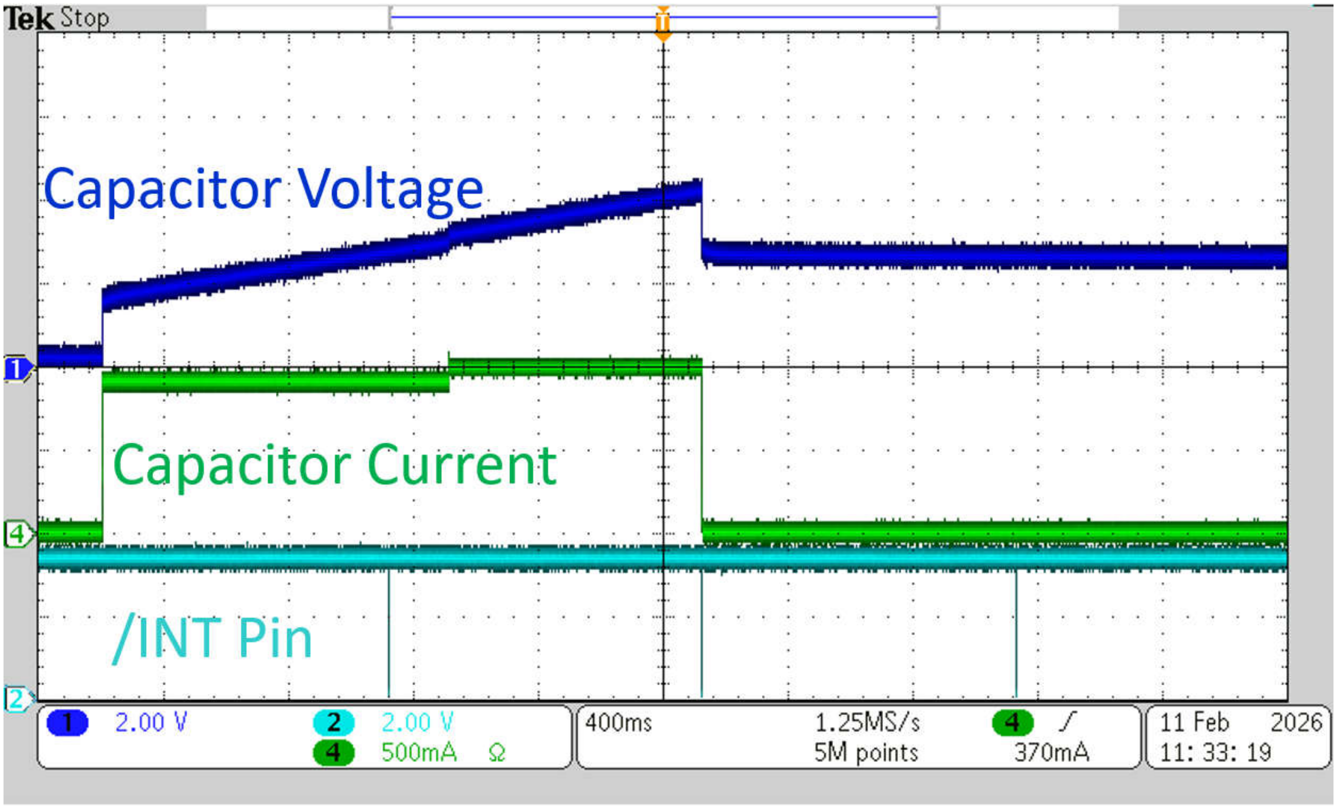


Figure 3-3. Sample Measurement Waveform

Copyright © 2026 Texas Instruments Incorporated

Figure 3-3 shows a sample waveform demonstrating the measurement code implementation. The device begins to charge the capacitor with the I_{CHG} current, which is probed on channel 4. The capacitor voltage seen by the IC is probed on channel 1. An interrupt is asserted from the /INT pin of the device every time the ADC measurement occurs. The /INT pin is monitored on channel 2 to demonstrate the timing of the code. The first pulse on channel 2 shows the first ADC measurement to determine $V_{Initial}$. After this, the timer begins and the charge runs for 1 second. The second pulse on channel 2 shows the second ADC measurement to determine V_{Ending}. The charging is disabled immediately after the ADC measurement, and there is another waiting period to let the capacitor voltage settle. The third pulse on channel 2 shows the last ADC measurement to determine V_{Ending}. Using Equation 5 and Equation 4, the ESR and capacitance can be determined with the ADC measurements from the BQ25856-Q1, the known charge current from the BQ25856-Q1 settings, and the known charge time from the MCU timer.

Table 3-1. ADC Voltage Measurements

ADC Reading	Voltage (V)
Initial Voltage	2.586
Peak Voltage	4.000
Final Voltage	2.454

Table 3-2. Supercapacitor Parameter Calculations

Parameter	Result
Capacitance of eight capacitors in series	0.71F
ESR of the eight capacitors in series	1.55Ω
Average capacitance per capacitor	5.68F
Average ESR per capacitor	194mΩ

The full charging cycle can resume after the measurement has taken place until the capacitors are charged and ready to supply backup power. The calculated capacitance and ESR can be tracked over time to determine when the capacitors have reached end of life. For this test, the capacitors were old and worn and have reached end of life because the ESR is much greater than nominal and the capacitance has decreased to 70% of the original value. The capacitor has reached EOL when either the calculated capacitance falls below 80% of nominal value or the ESR exceeds 200% of nominal value.

Copyright © 2026 Texas Instruments Incorporated

3.3 Estimating the Capacitance and ESR in Reverse Mode

Measuring capacitance and ESR with the BQ25856-Q1 in reverse mode is also possible, provided certain conditions are met. The BQ25856-Q1 requires a load on the VAC side to drain current from the capacitor while the IC is in reverse mode. Also, the BQ25856-Q1 predominately acts as a constant voltage source when in reverse mode. This means that the supercapacitor will not have a linear current while sourcing power.

There are a few methods to measure the total current from the capacitor. First, if the system has a way to measure the average source current from the capacitor, the average current multiplied by time can be substituted for the integral of $\int_0^t i\,\mathrm{d}\,t$. This gives a value in coulombs. The equation for a capacitor with respect to charge is:

$$C = \frac{q}{V} \tag{9}$$

The capacitor equation can be rearranged into:

$$\frac{\int_0^t i\,\mathrm{d}\,t}{V_{Ending} - V_{Start}} = C \tag{10}$$

Equation 10 does not take into account the voltage drop on the capacitor's ESR. TI recommends making sure the capacitor currents at V_{Ending} and V_{Start} are approximately equal.

The ESR can be calculated by measuring the difference in voltage between two different capacitor currents at a known time:

$$\frac{V_{C2} - V_{C1} - \frac{1}{C}\int_0^t i\,\mathrm{d}\,t}{(i_2 - i_1)} = R_{ESR} \tag{11}$$

In Equation 11, V_{C1} and V_{C2} are the measured capacitor voltages at the capacitor source currents of i_1 and i_2. t is the time between the V_{C1} and V_{C2} measurements. The capacitance C can be calculated with Equation 10.

Alternatively, if the MCU cannot calculate the average current, the IBAT_REV register can be used to limit the current out of the capacitor to 5A, 10A, 15A, or 20A. With the source current of the capacitor limited, the current will be linear and the equations in Section 3.1 can be used to calculate the capacitance and ESR. Make sure the load on VAC loads the input enough that IBAT_REV becomes active and limits the discharge current.

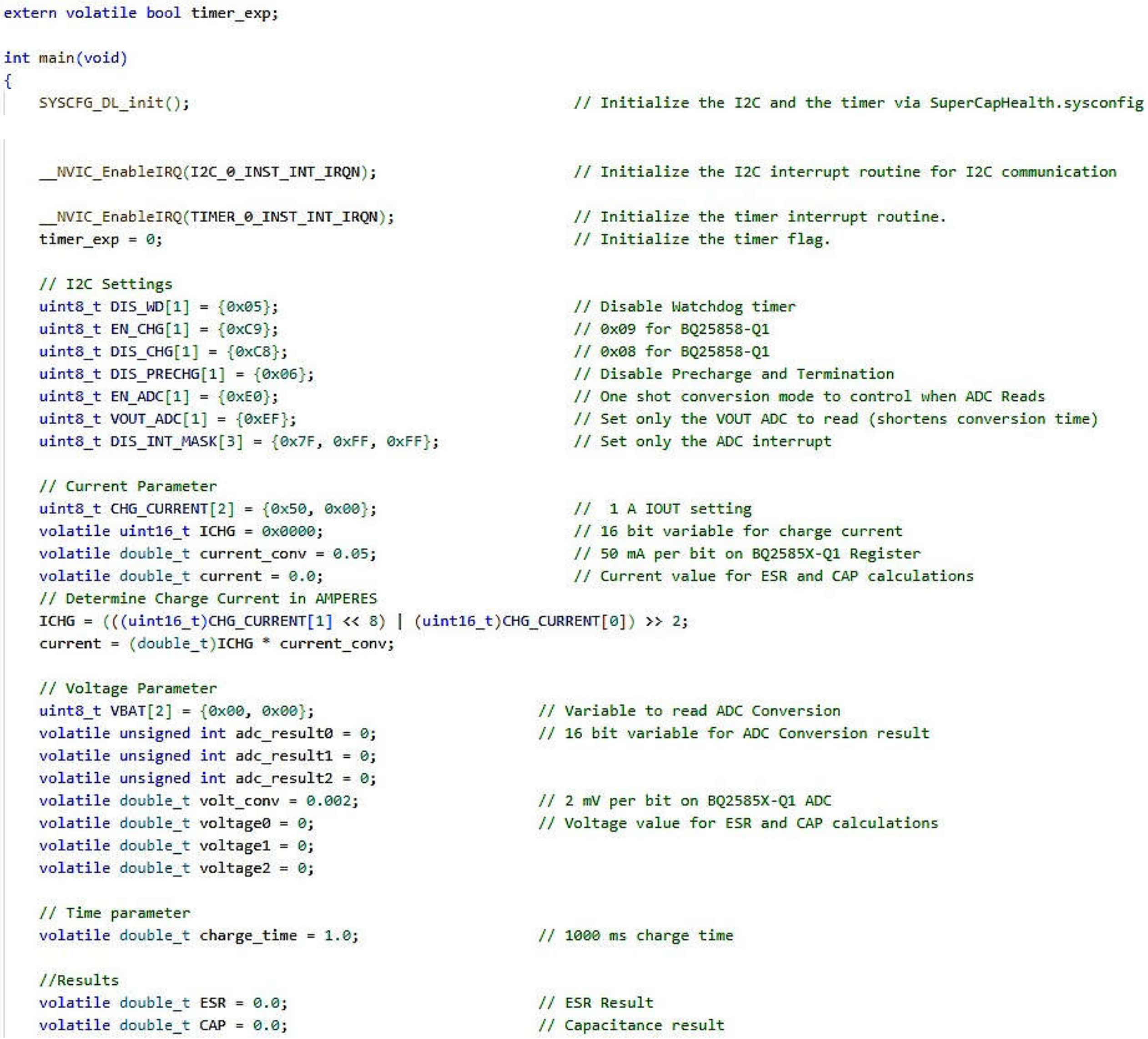
Copyright © 2026 Texas Instruments Incorporated

4 Summary

Supercapacitors have large benefits over electrolytic capacitors or batteries. Supercapacitors have a much greater power density and much longer lifetime compared to batteries. Also, supercapacitors can store more energy than electrolytic capacitors. However, supercapacitors still age and the usable power and energy decreases over time. The BQ2585X-Q1 and BQ2575X family of ICs can estimate the ESR and capacitance of supercapacitors. This information helps track the health of the supercapacitor and can be used to estimate the end-of-life for the capacitor.

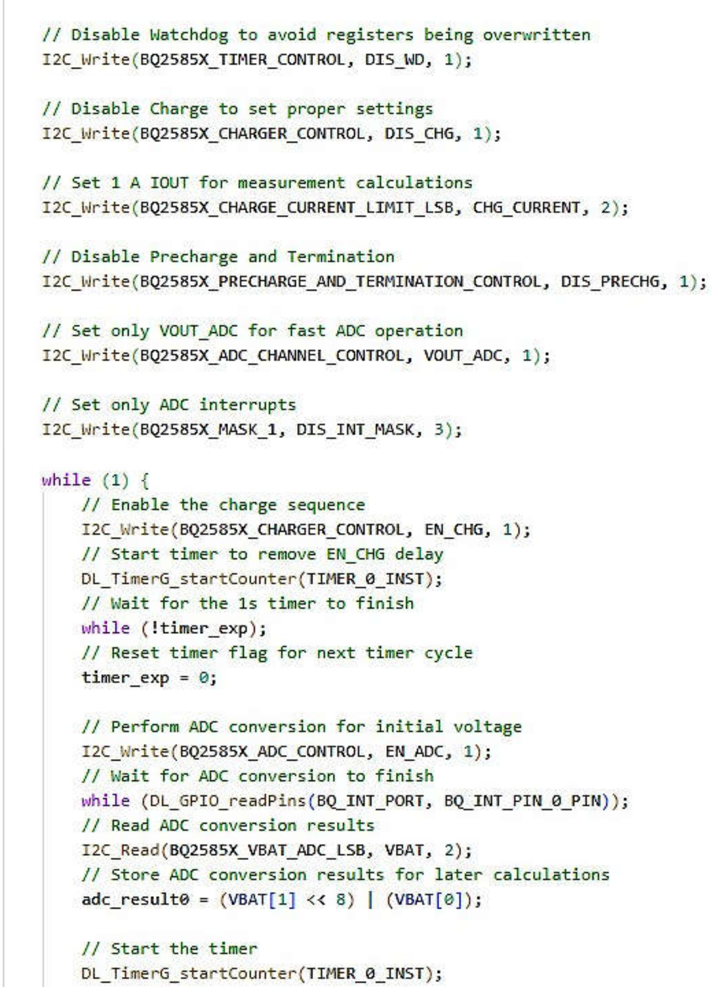
Copyright © 2026 Texas Instruments Incorporated

5 Appendix - Example MCU Code for the BQ25858-Q1



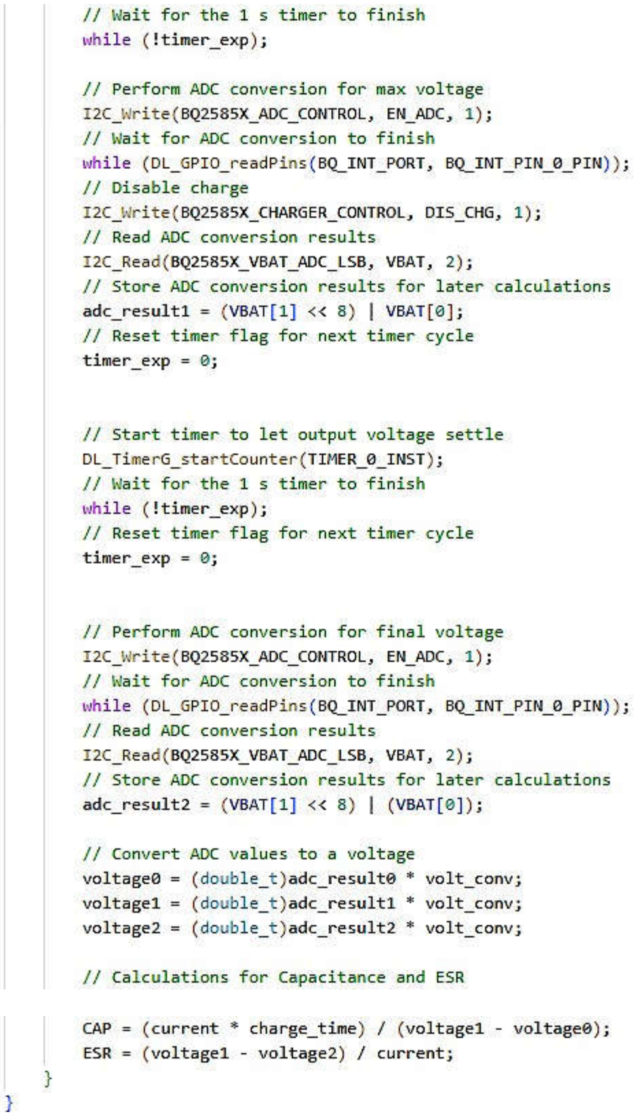
```
extern volatile bool timer_exp;

int main(void)
{
    SYSCFG_DL_init();                                           // Initialize the I2C and the timer via SuperCapHealth.sysconfig


    __NVIC_EnableIRQ(I2C_0_INST_INT_IRQN);                      // Initialize the I2C interrupt routine for I2C communication

    __NVIC_EnableIRQ(TIMER_0_INST_INT_IRQN);                    // Initialize the timer interrupt routine.
    timer_exp = 0;                                              // Initialize the timer flag.

    // I2C Settings
    uint8_t DIS_WD[1] = {0x05};                                 // Disable Watchdog timer
    uint8_t EN_CHG[1] = {0xC9};                                 // 0x09 for BQ25858-Q1
    uint8_t DIS_CHG[1] = {0xC8};                                // 0x08 for BQ25858-Q1
    uint8_t DIS_PRECHG[1] = {0x06};                             // Disable Precharge and Termination
    uint8_t EN_ADC[1] = {0xE0};                                 // One shot conversion mode to control when ADC Reads
    uint8_t VOUT_ADC[1] = {0xEF};                               // Set only the VOUT ADC to read (shortens conversion time)
    uint8_t DIS_INT_MASK[3] = {0x7F, 0xFF, 0xFF};               // Set only the ADC interrupt

    // Current Parameter
    uint8_t CHG_CURRENT[2] = {0x50, 0x00};                      //  1 A IOUT setting
    volatile uint16_t ICHG = 0x0000;                            // 16 bit variable for charge current
    volatile double_t current_conv = 0.05;                      // 50 mA per bit on BQ2585X-Q1 Register
    volatile double_t current = 0.0;                            // Current value for ESR and CAP calculations
    // Determine Charge Current in AMPERES
    ICHG = (((uint16_t)CHG_CURRENT[1] << 8) | (uint16_t)CHG_CURRENT[0]) >> 2;
    current = (double_t)ICHG * current_conv;

    // Voltage Parameter
    uint8_t VBAT[2] = {0x00, 0x00};                         // Variable to read ADC Conversion
    volatile unsigned int adc_result0 = 0;                  // 16 bit variable for ADC Conversion result
    volatile unsigned int adc_result1 = 0;
    volatile unsigned int adc_result2 = 0;
    volatile double_t volt_conv = 0.002;                    // 2 mV per bit on BQ2585X-Q1 ADC
    volatile double_t voltage0 = 0;                         // Voltage value for ESR and CAP calculations
    volatile double_t voltage1 = 0;
    volatile double_t voltage2 = 0;

    // Time parameter
    volatile double_t charge_time = 1.0;                    // 1000 ms charge time

    //Results
    volatile double_t ESR = 0.0;                            // ESR Result
    volatile double_t CAP = 0.0;                            // Capacitance result
```

Copyright © 2026 Texas Instruments Incorporated



```
// Disable Watchdog to avoid registers being overwritten
I2C_Write(BQ2585X_TIMER_CONTROL, DIS_WD, 1);

// Disable Charge to set proper settings
I2C_Write(BQ2585X_CHARGER_CONTROL, DIS_CHG, 1);

// Set 1 A IOUT for measurement calculations
I2C_Write(BQ2585X_CHARGE_CURRENT_LIMIT_LSB, CHG_CURRENT, 2);

// Disable Precharge and Termination
I2C_Write(BQ2585X_PRECHARGE_AND_TERMINATION_CONTROL, DIS_PRECHG, 1);

// Set only VOUT_ADC for fast ADC operation
I2C_Write(BQ2585X_ADC_CHANNEL_CONTROL, VOUT_ADC, 1);

// Set only ADC interrupts
I2C_Write(BQ2585X_MASK_1, DIS_INT_MASK, 3);

while (1) {
    // Enable the charge sequence
    I2C_Write(BQ2585X_CHARGER_CONTROL, EN_CHG, 1);
    // Start timer to remove EN_CHG delay
    DL_TimerG_startCounter(TIMER_0_INST);
    // Wait for the 1s timer to finish
    while (!timer_exp);
    // Reset timer flag for next timer cycle
    timer_exp = 0;

    // Perform ADC conversion for initial voltage
    I2C_Write(BQ2585X_ADC_CONTROL, EN_ADC, 1);
    // Wait for ADC conversion to finish
    while (DL_GPIO_readPins(BQ_INT_PORT, BQ_INT_PIN_0_PIN));
    // Read ADC conversion results
    I2C_Read(BQ2585X_VBAT_ADC_LSB, VBAT, 2);
    // Store ADC conversion results for later calculations
    adc_result0 = (VBAT[1] << 8) | (VBAT[0]);

    // Start the timer
    DL_TimerG_startCounter(TIMER_0_INST);
```

Copyright © 2026 Texas Instruments Incorporated



```
        // Wait for the 1 s timer to finish
        while (!timer_exp);

        // Perform ADC conversion for max voltage
        I2C_Write(BQ2585X_ADC_CONTROL, EN_ADC, 1);
        // Wait for ADC conversion to finish
        while (DL_GPIO_readPins(BQ_INT_PORT, BQ_INT_PIN_0_PIN));
        // Disable charge
        I2C_Write(BQ2585X_CHARGER_CONTROL, DIS_CHG, 1);
        // Read ADC conversion results
        I2C_Read(BQ2585X_VBAT_ADC_LSB, VBAT, 2);
        // Store ADC conversion results for later calculations
        adc_result1 = (VBAT[1] << 8) | VBAT[0];
        // Reset timer flag for next timer cycle
        timer_exp = 0;


        // Start timer to let output voltage settle
        DL_TimerG_startCounter(TIMER_0_INST);
        // Wait for the 1 s timer to finish
        while (!timer_exp);
        // Reset timer flag for next timer cycle
        timer_exp = 0;


        // Perform ADC conversion for final voltage
        I2C_Write(BQ2585X_ADC_CONTROL, EN_ADC, 1);
        // Wait for ADC conversion to finish
        while (DL_GPIO_readPins(BQ_INT_PORT, BQ_INT_PIN_0_PIN));
        // Read ADC conversion results
        I2C_Read(BQ2585X_VBAT_ADC_LSB, VBAT, 2);
        // Store ADC conversion results for later calculations
        adc_result2 = (VBAT[1] << 8) | (VBAT[0]);

        // Convert ADC values to a voltage
        voltage0 = (double_t)adc_result0 * volt_conv;
        voltage1 = (double_t)adc_result1 * volt_conv;
        voltage2 = (double_t)adc_result2 * volt_conv;

        // Calculations for Capacitance and ESR

        CAP = (current * charge_time) / (voltage1 - voltage0);
        ESR = (voltage1 - voltage2) / current;
    }
}
```

```
int main(void) {
  SYSCFG_DL_init(); // Initialize the I2C and the timer via SuperCapHealth.sysconfig
  __NVIC_EnableIRQ(I2C_0_INST_INT_IRQN); // Initialize the I2C interrupt routine for I2C
communication
  __NVIC_EnableIRQ(TIMER_0_INST_INT_IRQN); // Initialize the timer interrupt routine.
  timer_exp = 0; // Initialize the timer flag.
  // I2C Settings
  uint8_t DIS_WD[1] = {0x05}; // Disable Watchdog timer
  uint8_t EN_CHG[1] = {0xC9}; // 0x09 for BQ25858-Q1
  uint8_t DIS_CHG[1] = {0xC8}; // 0x08 for BQ25858-Q1
  uint8_t DIS_PRECHG[1] = {0x06}; // Disable Precharge and Termination
  uint8_t EN_ADC[1] = {0xE0}; // One shot conversion mode to control when ADC Reads
  uint8_t VOUT_ADC[1] = {0xEF}; // Set only the VOUT ADC to read (shortens conversion time)
  uint8_t DIS_INT_MASK[3] = {0x7F, 0xFF, 0xFF}; // Set only the ADC interrupt for the best results
  // Current Parameter
  uint8_t CHG_CURRENT[2] = {0x50, 0x00}; // 1 A IOUT setting
  volatile uint16_t ICHG = 0x0000; // 16 bit variable for charge current volatile
  double_t current_conv = 0.05; // 50 mA per bit on BQ2585X-Q1 Register
  volatile double_t current = 0.0; // Current value for ESR and CAP calculations
  // Determine Charge Current in AMPERES
  ICHG = (((uint16_t)CHG_CURRENT[1] << 8) | (uint16_t)CHG_CURRENT[0]) >> 2;
  current = (double_t)ICHG * current_conv;
  // Voltage Parameter
  uint8_t VBAT[2] = {0x00, 0x00};
  // Variable to read ADC Conversion
  volatile unsigned int adc_result0 = 0; // 16 bit variable for ADC Conversion result
  volatile unsigned int adc_result1 = 0;
```

Copyright © 2026 Texas Instruments Incorporated

```
  volatile unsigned int adc_result2 = 0;
  volatile double_t volt_conv = 0.002; // 2 mV per bit on BQ2585X-Q1 ADC
  volatile double_t voltage0 = 0; // Voltage value for ESR and CAP calculations
  volatile double_t voltage1 = 0;
  volatile double_t voltage2 = 0;
  // Time parameter
  volatile double_t charge_time = 1.0; // 1000 ms charge time
  //Results
  volatile double_t ESR = 0.0; // ESR Result
  volatile double_t CAP = 0.0; // Capacitance result
  // Disable Watchdog to avoid registers being overwritten
  I2C_Write(BQ2585X_TIMER_CONTROL, DIS_WD, 1);
  // Disable Charge to set proper settings
  I2C_Write(BQ2585X_CHARGER_CONTROL, DIS_CHG, 1);
  // Set 1 A IOUT for measurement calculations
  I2C_Write(BQ2585X_CHARGE_CURRENT_LIMIT_LSB, CHG_CURRENT, 2);
  // Disable Precharge and Termination
  I2C_Write(BQ2585X_PRECHARGE_AND_TERMINATION_CONTROL, DIS_PRECHG, 1);
  // Set only VOUT_ADC for best operation
  I2C_Write(BQ2585X_ADC_CHANNEL_CONTROL, VOUT_ADC, 1);
  // Set only ADC interrupts for best operation
  I2C_Write(BQ2585X_MASK_1, DIS_INT_MASK, 3);
  while (1) {
    // Enable the charge sequence
    I2C_Write(BQ2585X_CHARGER_CONTROL, EN_CHG, 1);
    // Start timer to remove EN_CHG delay
    DL_TimerG_startCounter(TIMER_0_INST);
    // Wait for the 1 s timer to finish
    while (!timer_exp);
    // Reset timer flag for next timer cycle
    timer_exp = 0;
    // Perform ADC conversion for initial voltage
    I2C_Write(BQ2585X_ADC_CONTROL, EN_ADC, 1);
    // Wait for ADC conversion to finish
    while (DL_GPIO_readPins(BQ_INT_PORT, BQ_INT_PIN_0_PIN));
    // Read ADC conversion results
    I2C_Read(BQ2585X_VBAT_ADC_LSB, VBAT, 2);
    // Store ADC conversion results for later calculations
    adc_result0 = (VBAT[1] << 8) | (VBAT[0]);
    // Start the timer
    DL_TimerG_startCounter(TIMER_0_INST);
    // Wait for the 1 s timer to finish
    while (!timer_exp);
    // Perform ADC conversion for max voltage
    I2C_Write(BQ2585X_ADC_CONTROL, EN_ADC, 1);
    // Wait for ADC conversion to finish
    while (DL_GPIO_readPins(BQ_INT_PORT, BQ_INT_PIN_0_PIN));
    // Disable charge
    I2C_Write(BQ2585X_CHARGER_CONTROL, DIS_CHG, 1);
    // Read ADC conversion results
    I2C_Read(BQ2585X_VBAT_ADC_LSB, VBAT, 2);
    // Store ADC conversion results for later calculations
    adc_result1 = (VBAT[1] << 8) | VBAT[0];
    // Reset timer flag for next timer cycle
    timer_exp = 0;
    // Start timer to let output voltage settle
    DL_TimerG_startCounter(TIMER_0_INST);
    // Wait for the 1 s timer to finish
    while (!timer_exp);
    // Reset timer flag for next timer cycle
    timer_exp = 0;
    // Perform ADC conversion for final voltage
    I2C_Write(BQ2585X_ADC_CONTROL, EN_ADC, 1);
    // Wait for ADC conversion to finish
    while (DL_GPIO_readPins(BQ_INT_PORT, BQ_INT_PIN_0_PIN));
    // Read ADC conversion results
    I2C_Read(BQ2585X_VBAT_ADC_LSB, VBAT, 2);
    // Store ADC conversion results for later calculations
    adc_result2 = (VBAT[1] << 8) | (VBAT[0]);
    // Convert ADC values to a voltage
    voltage0 = (double_t)adc_result0 * volt_conv;
    voltage1 = (double_t)adc_result1 * volt_conv;
    voltage2 = (double_t)adc_result2 * volt_conv;
    // Calculations for Capacitance and ESR
    CAP = (current * charge_time) / (voltage1 - voltage0);
    ESR = (voltage1 - voltage2) / current;
  }
}
```

Copyright © 2026 Texas Instruments Incorporated

6 References

1. Wiley Advanced, *The Many Deaths of Supercapacitors: Degradation, Aging, and Performance Fading*,webpage.
2. Texas Instruments, *BQ25856-Q1: Automotive, Standalone/I2C Controlled, 1- to 14-Cell Bidirectional Buck Boost Battery Charge Controller*, datasheet.
3. Texas Instruments, *Capacitor Backup Circuits with the BQ25856-Q1*, application note.

Copyright © 2026 Texas Instruments Incorporated

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you fully indemnify TI and its representatives against any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale, TI’s General Quality Guidelines, or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.

Copyright © 2026, Texas Instruments Incorporated

Last updated 10/2025