Application Note

BQ27Z746 Hardware Protection Threshold Calibration

Kipp Hayes

ABSTRACT

The BQ27Z746 has several programmable hardware protection options that allow the user flexibility to adjust the integrated protection thresholds for a specific battery and application. These hardware protection values include over-voltage protection (OVP), under-voltage protection (UVP), over-current in charge (OCC), over-current in discharge (OCD), short-circuit detection (SCD), BAT_SP to PACK - short detection (BDP), BAT_SP to PACK + short detection (BCP), BAT_SN to PACK - short detection (BDN), BAT_SN to PACK + short detection.

This app note covers the protector tuning, protector delay, as well as programming the protection in production.

Table of Contents

Trademarks

All trademarks are the property of their respective owners.

1 Protector Tuning

This section covers how to set the protection values using the protector tuning method.

The protector tuning method does not require any reference voltage to be placed on the device to trim the protections. The expected typical error for UVP and OVP protections is 50mV. The expected typical error for OCC,OCD, and SCD protections is 3mV.

Step 1: Download and install the latest version of BQSTUDIO-TEST.

Step 2: Connect EV2400 with BQ27Z746EVM as in Figure 1-1.

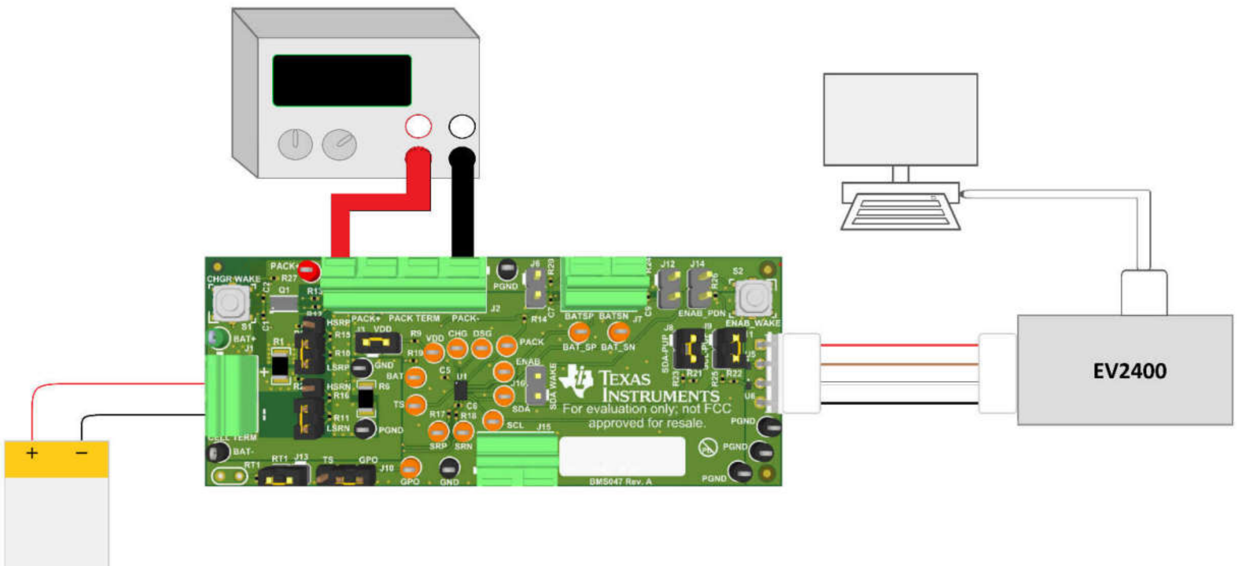

Figure 1-1. Hardware Setup

Step 3: Open BQStudio, navigate to the calibration plugin, and select the Protector Tuning Panel.

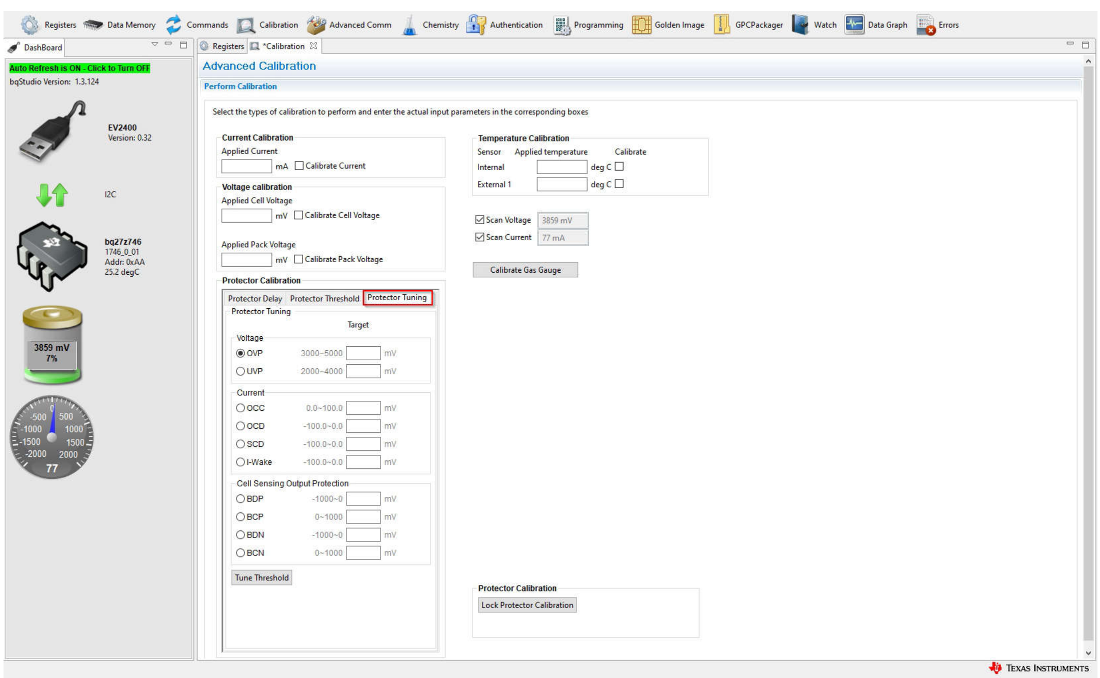

Figure 1-2. Protector Tuning Panel

Step 4: In the Protector Tuning panel, input the desired protection value into the corresponding protection that is being set. A green check mark appears indicating success.

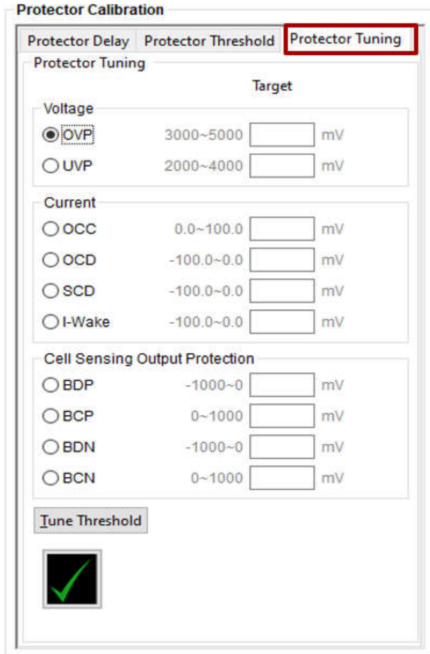

Figure 1-3. Protector Tuning Success

2 Protector Delay

This section covers how to set the protector delay values.

Step 1: Repeat steps 1 to 2 in the protector tuning section.

Step 2: Open BQStudio, navigate to the calibration plugin, and select the Protector Delay Panel.

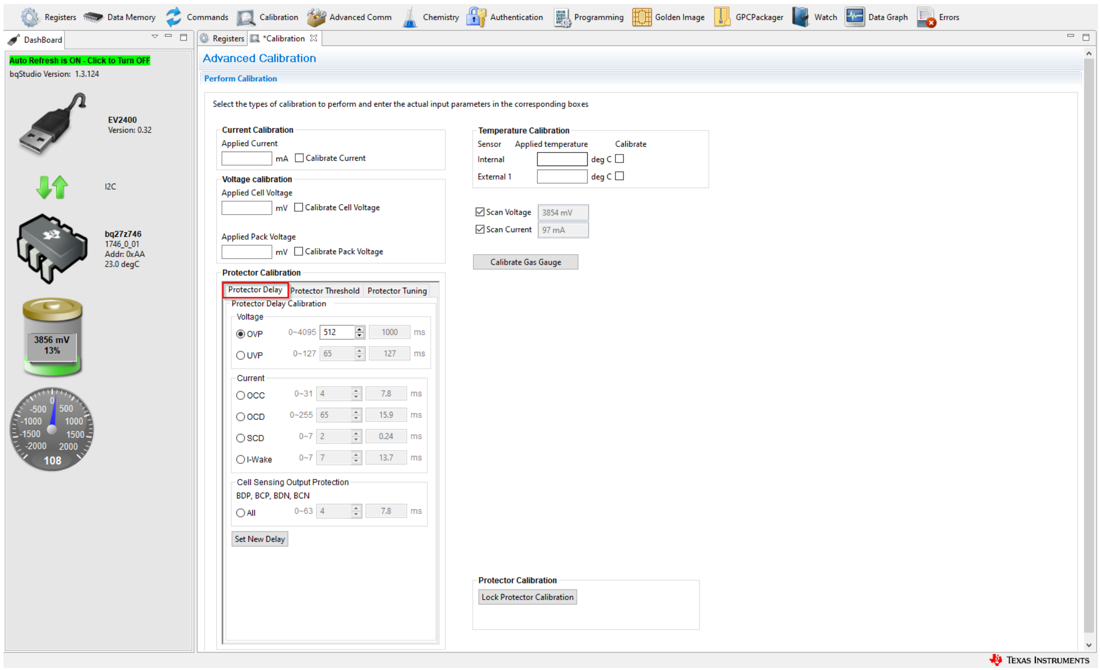

Figure 2-1. Protector Delay Panel

Step 3: In the Protector Delay panel, input the desired protection value into the corresponding protection that is being set. A green check mark appears indicating success.

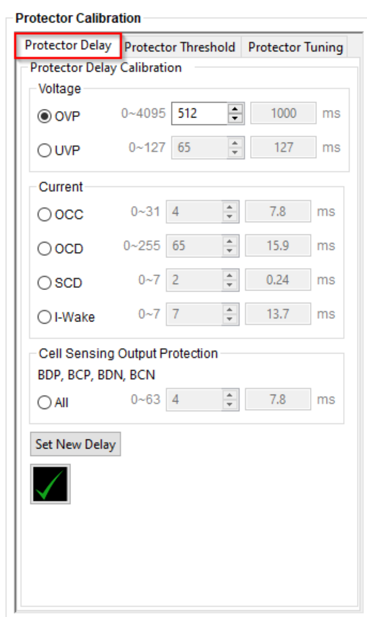

Figure 2-2. Protector Delay Success

3 Going to Production With Protector Tuning

With the protector tuning method, the protection values can set by writing to a register. This is specifically attractive for mass production simplification. The protection values are not included in the output golden image file when development is completed. Below are the additional steps to take in production to set the protections.

Step 1: Send CALIBRATION Mode command 0x002D to address AA and start register 3E on the gauge used during development. This development gauge has the final protection values set from the above Protector Tuning section.

Calibration mode is required to read and write the protection values register.

Step 2: Read ProtectorImage2 command 0xF0A2 to address AA and start register 3E on the gauge used during development.

This shows the hex value step of the protector values set. Note, these values do not correspond directly to decimal values, instead this is the hex value for the protector step set.

The register values read from the command look something like below:

A2 F0 01 00 17 2C 20 08 08 08 09 08 **27 1F 1B 5B 2E 04 5B 5C 5C 5C** 08 08 10 17 07 08 09 11 09 11

Bits 10-19 are the protection step values. Note these for use in programming onto the gauge in production. For seeing the bit mapping for ProtectorImage2, please refer to section 15.2.70 in the BQ27Z746 technical reference manual.

Step 3: Send CALIBRATION Mode command 0x002D to address AA and start register 3E on the gauge in production.

Calibration mode is required to read and write the protection values register.

Step 4: Read ProtectorImage2 command 0xF0A2 to address AA and start register 3E on the gauge in production.

These values can differ depending on trim done during the manufacturing of the gauge by TI, and can look something like below:

A2 F0 01 23 17 2C 10 08 09 10 09 08 22 10 32 5A 3E 04 5B 4C 5C 5C **08 08 10 17 07 08 09 11 09 11**

The values in bold are accuracy trimming values for the protection and must not be changed on the gauge. This maintains protection trip accuracy that aligns with the data sheet.

Step 5: Combine the protection values from the development gauge with the super comp trim values read from the production gauge. From the above examples, looks like this:

A2 F0 01 23 17 2C 10 08 09 10 09 08 27 1F 1B 5B 2E 04 5B 5C 5C 5C 08 08 10 17 07 08 09 11 09 11

Step 6: Write the above combined values to address AA and start register 3E to the gauge in production.

Step 7: Write the check sum followed by the length to address AA and start register 0x60 to the gauge in production.

To calculate the checksum, take the least significant byte of the bitwise NOT of the sum of the command+data hex values written. The length is the command size + data size + checksum size + length size.

The above example checksum would be calculated as such:

Checksum =A2+F0+01+23+17+2C+10+08+09+10+09+08+27+1F+1B+5B+2E+04+5B+5C+5C+5C+08+08+10+17+07+08+09+11+09+11=485

Checksum = NOT(485) = FB7A

Checksum = 7A

Length = 2 command bytes + 30 data bytes + 1 checksum byte + 1 length byte = 34 bytes

34 dec to hex = 22, therefore what is written to 0x60 is 7A 22.

Step 8: The process then needs to be repeated starting at step 6 for the Protection Delay which is located in 0xF0A1. For this case, no bits need to be preserved and the register can be fully written to.

For example, reading the 0xF0A1 values from the development gauge reads back:

A1 F0 00 00 00 00 00 00 00 00 00 00 **05 41 04 6C 07 41 07 04** 00 00 00 00 00 00 00 00 00 00 00 00

The full register can be written to the production gauge without any issue.

The bit mapping for protector delay 0xF0A1 ProtectorImage1 is as follows:

Table 3-1. Bit Map

Offset	Name	Description
0-9	RSVD	Reserved
10	SCD_Delay	Short Circuit in Discharge Protection Delay
11	OCD_Delay	Overcurrent in Discharge Protection Delay
12-13	OVP_Delay	Overvoltage Protection Delay
14	OCC_Delay	Overcurrent in Charge Protection Delay
15	UVP_Delay	Undervoltage Proection Delay
16	IWK_Delay	I-Wake trigger Delay
17	BSENSE_Delay	Battery Sensing Protection Delay, BDP,BCP,BDN,BCN
18-29	RSVD	Reserved

Step 8: Send the ProtectorImageSave command 0xF0A3 followed by 0x00 to address AA and start register 3E.

Step 9: Read MACData() 0x40/0x5F

The ProtectorImageSave command saves the present state of the protector hardware registers to the configuration image in secure memory. Upon reading the MACData(), after writing, the read value is 00, if not, then there was an error. Check the technical reference manual section 15.2.71 to debug this error.

Step 9: Send the ProtectorImageLock command 0xF0A4 followed by 0x83de in little endian order, 0xde + 0x83 to address AA and start register 3E

Step 10: Read MACData() 0x40/0x5F

The ProtectorImageLock command locks the entire protector configuration image in secure memory to prevent any further modification. Note this is a permanent lock and cannot be unlocked or reversed. Upon reading the MACData(), after writing, the value is 00, if not, then there was an error. Check the technical reference manual section 15.2.72 to debug this error.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265