

Application Note

具有密封配置的 BQ769X2 OTP 编程示例指南



Jie Wu

摘要

BQ76942 (支持三节至 10 节串联电池) 和 BQ76952 (支持三节至 16 节串联电池) 是电池包中使用的电池监测器和保护器件。这两个器件集成了一次性可编程 (OTP) 存储器。将初始化设置写入 OTP 有助于在系统上电前保存初始化配置。用户需要在产品线上编写 OTP。本应用手册逐步介绍了硬件设置和软件过程, 可帮助用户节省完成 OTP 配置的时间。

内容

1 简介.....	2
2 硬件设置.....	2
3 CRC 和校验和编程及示例命令.....	3
3.1 CRC 计算示例.....	3
3.2 校验和计算示例.....	4
4 OTP 编程示例.....	4
4.1 配置进入密封模式的编程步骤.....	4
4.2 OTP 编程流程图.....	5
4.3 OTP 逐步命令示例.....	6
5 总结.....	7
6 参考资料.....	7

插图清单

图 2-1. OTP 编程硬件设置方框图.....	2
图 3-1. CRC 软件示例代码.....	3
图 3-2. 具有 CRC 的写入命令格式.....	3
图 3-3. 读取具有 CRC 的命令格式.....	3
图 3-4. 用于校验和计算的软件编码.....	4
图 3-5. 具有校验和及 CRC 的命令.....	4
图 4-1. OTP 进度流程图.....	5

商标

所有商标均为其各自所有者的财产。

1 简介

BQ769X2 器件可用于锂离子、锂聚合物和磷酸铁锂电池包。该器件的特性包括高精度监控电池电压、可配置的保护功能、主机控制型电池平衡、内部温度感测和集成式 OTP 存储器。对于希望在产品线上永久配置器件参数的用户，可以使用 OTP 存储器来存储一些关键参数。本应用手册重点逐步介绍了 OTP 编程，使设计流程更加简单。

2 硬件设置

在开始进行 OTP 配置之前，请注意 BQ769X2 电源；pack+ 和 pack- 之间的电压必须在约 10V-12V 的范围内。

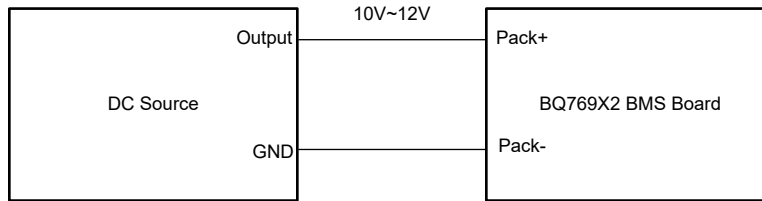


图 2-1. OTP 编程硬件设置方框图

3 CRC 和校验和编程及示例命令

代码示例包括不同命令中的 CRC 计算和校验和计算。以下各节显示了相关软件代码示例和应用示例。

3.1 CRC 计算示例

图 3-1 显示了 CRC 软件编码示例。

```

92 // Calculates the checksum when writing to a RAM register. The checksum is the
93 {
94     unsigned char i;
95     unsigned char checksum = 0;
96
97     for (i = 0; i < len; i++) checksum += ptr[i];
98
99     checksum = 0xff & ~checksum;
100
101     return (checksum);
102 }
103
104 unsigned char CRC8(unsigned char *ptr, unsigned char len)
105 //Calculates CRC8 for passed bytes. Used in i2c read and write functions
106 {
107     unsigned char i;
108     unsigned char crc = 0;
109     while (len-- != 0) {
110         for (i = 0x80; i != 0; i /= 2) {
111             if ((crc & 0x80) != 0) {
112                 crc *= 2;
113                 crc ^= 0x107;
114             } else
115                 crc *= 2;
116
117             if ((*ptr & i) != 0) crc ^= 0x107;
118         }
119         ptr++;
120     }
121     return (crc);
122 }
123
    
```

图 3-1. CRC 软件示例代码

图 3-2 显示了一种写入命令格式，其中包括基于 I2C 通信的 CRC 计算。例如，将子命令 0x0022 (FET_ENABLE 子命令) 写入，并进行 CRC 计算。byte3 的 CRC 是针对 [0x10 0x3E 0x22] 计算的；CRC 计算结果为 0x63。第二字节 [0x00] 的 CRC 为 0x00。相关的数据序列为 0x10 0x3E 0x22 0x63 0x00 0x00。

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5
Device address	Subcommand/Data memory read or write	Register Address MSB	CRC calculation result(Byte0~Byte2)	Register Address LSB	CRC calculation result(Byte4)
0x10	0x3E	0x22	0x63	0x00	0x00

图 3-2. 具有 CRC 的写入命令格式

图 3-3 显示了包括 CRC 的读取命令格式。示例读取 0x14(VCell 1)，回读日期为 0x0B68。在 CRC 计算中，byte4 的 CRC 是针对 [0x10 0x14 0x11 0x68] 计算；CRC 计算结果为 0x33。第二字节 [0x0B] 的 CRC 为 0x31。相关的数据序列为 0x10 0x14 0x11 0x68 0x33 0x0B 0x31。

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6
Device address	Register to be read	Read command	Read back data MSB	CRC calculation result(Byte0~Byte3)	Read back data LSB	CRC calculation result(Byte5)
0x10	0x14	0x11	0x68	0x33	0x0B	0x31

图 3-3. 读取具有 CRC 的命令格式

3.2 校验和计算示例

图 3-4 显示了一个用于校验和计算的软件编码。

```

70
71 unsigned char Checksum(unsigned char *ptr, unsigned char len)
72 // Calculates the checksum when writing to a RAM register. The checksum is the inverse of the sum of the bytes.
73 {
74     unsigned char i;
75     unsigned char checksum = 0;
76
77     for (i = 0; i < len; i++) checksum += ptr[i];
78
79     checksum = 0xff & ~checksum;
80
81     return (checksum);
82 }
77
    
```

图 3-4. 用于校验和计算的软件编码

图 3-5 显示了具有校验和格式的 I2C 命令示例。例如，将数据 0x8C 写入寄存器 0x9261，然后将校验和数据及长度写入 0x60/0x61。校验和是根据地址和数据 (0x61 0x92 0x8C) 计算。长度还包括器件地址 (0x10) 和子命令 (0x3E) 的两个字节，但不包括 CRC 字节，在本例中总长度为 5。

数据格式如下所示：

0x10 0x3E 0x61 0xAD 0x92 0xF7 0x8C 0xAD ;

0x10 0x60 0x80 0xDE 0x05 0x1B ;

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
Device address	Subcommand	Register Address MSB	CRC calculation result(Byte0~Byte2)	Register Address LSB	CRC calculation result(Byte4)	Data to write	CRC calculation result(Byte6)
0x10	0x3E	0x61	0xAD	0x92	0xF7	0x8C	0xAD
Byte0	Byte1	Byte2	Byte3	Byte4	Byte5		
Device address	Checksum buffer	Checksum result	CRC calculation result(Byte0~Byte2)	Data length	CRC calculation result(Byte4)		
0x10	0x60	0x80	0xDE	0x05	0x1B		

图 3-5. 具有校验和及 CRC 的命令

4 OTP 编程示例

4.1 配置进入密封模式的编程步骤

1. 通过读取其中一个已编程寄存器，检查器件上是否已完成 OTP 编程。通电时，寄存器报告默认值或 OTP 中编程的值（如果 OTP 已编程）。如果尚未完成 OTP 编程，则转至后续步骤。
2. 读取 0x12 电池状态 [SEC1,SEC0] 位，以验证器件是否处于完全访问模式 (0x01)。
3. 如果器件处于完全访问模式，则进入 CONFIG_UPDATE 模式- (子命令 0x0090)。如果未进入该模式，请解封器件，然后返回步骤二检查器件是否处于完全访问模式。
4. 在数据存储器中配置寄存器设置。
5. 退出 CONFIG_UPDATE 模式 - (子命令 0x0092)。
6. 读取数据存储器寄存器以验证所有参数均已成功写入。
7. 进入 CONFIG_UPDATE 模式。
8. 检查电池状态 [OTPB] 位是否清除，以验证是否满足 OTP 编程条件。
9. 读取 OTP_WR_CHECK() (子命令 0x00A0)。如果返回值为 0x80，则满足 OTP 编程条件。
10. 如果 OTP_WR_CHECK 指示满足条件，则发送 OTP_WRITE() 子命令 (0x00A1)。
11. 等待 100 ms。从 0x40 读取以检查 OTP 编程是否成功 (0x80 表示成功)。
12. 进入密封模式。
13. 退出 CONFIG_UPDATE 模式 - (子命令 0x0092)。

4.2 OTP 编程流程图

图 4-1 显示了器件进入密封模式所需进行的 OTP 编程流程。

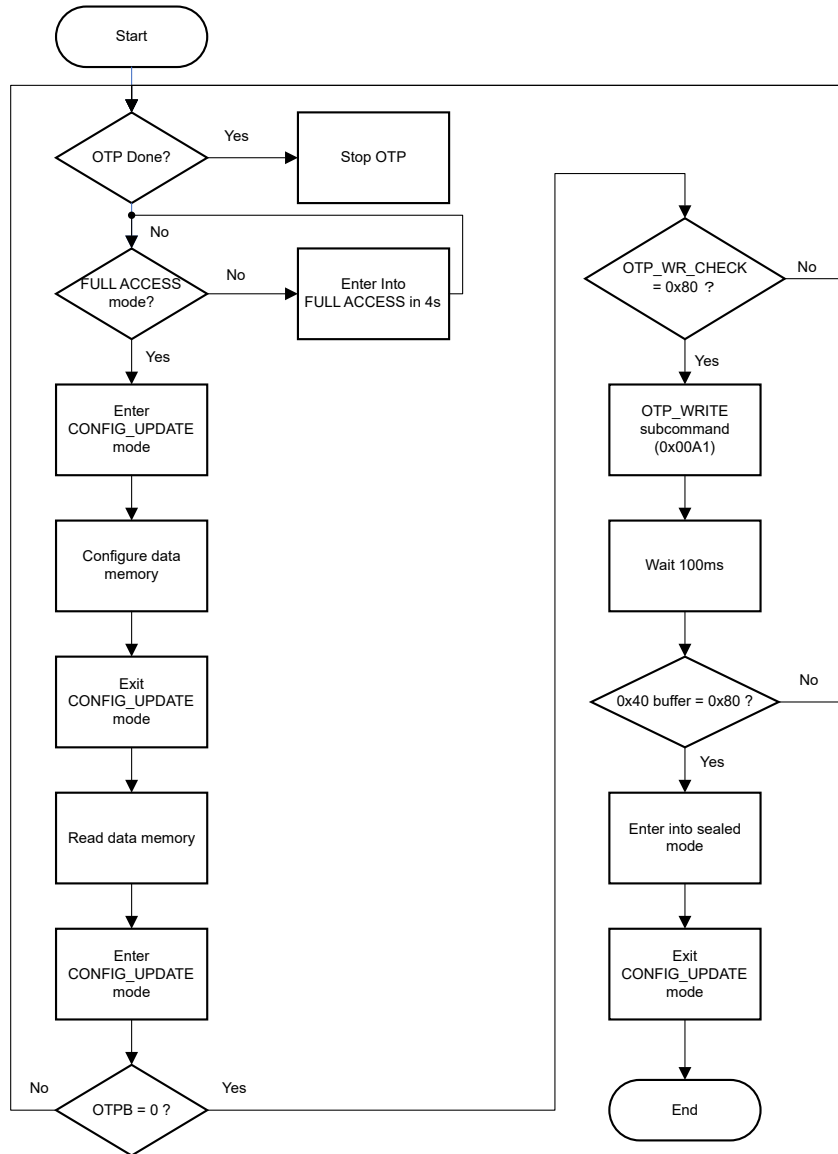


图 4-1. OTP 进度流程图

4.3 OTP 逐步命令示例

1. 通过读取其中一个已编程寄存器，检查器件上是否已完成 OTP 编程。通电时，寄存器报告默认值或 OTP 中编程的值（如果 OTP 已编程）。如果尚未完成 OTP 编程，则转至后续步骤。
 - a. 示例：更改 0x9180 设置，然后可以读取 0x9180 寄存器，并检查该寄存器是否显示默认值。如果显示默认值，则进入下一步。
 - i. 0x10 0x3E 0x80 Byte3(CRC) 0x91 Byte5(CRC)。
 - ii. 0x10 0x40 0x11 Byte3(Rdata) Byte4(CRC) Byte5(Rdata) Byte6(CRC)。
2. 读取 0x12 电池状态 [SEC1,SEC0] 位，以验证器件是否处于完全访问模式 (0x01)。
 - a. 读取 0x12 寄存器，如果 sec1=0，sec0=1，则器件处于完全访问模式。进入下一步。如果 sec1=1，sec0=1，则器件处于密封模式。用户需要先进入完全访问模式。
 - i. 0x10 0x12 0x11 Byte3(Rdata) Byte4(CRC) Byte5(Rdata) Byte6(CRC)。
 - b. 在读取数据存储器前 4 秒内进入完全访问模式。示例如下：Unseal_Key_Step1 0x1234；Unseal_Key_Step2 0x5678；FullAccess_Key_Step1 0x7856；FullAccess_Key_Step2 0x4321。
 - i. 0x10 0x3E 0x34 Byte3(CRC) 0x12 Byte5(CRC)；//Write Unseal_Key_Step1。
 - ii. 0x10 0x3E 0x78 Byte3(CRC) 0x12 Byte5(CRC)；//Write Unseal_Key_Step1。
 - iii. 0x10 0x12 0x11 Byte3(Rdata) Byte4(CRC) Byte5(Rdata) Byte6(CRC)；//Check Sec1 Sec0。
 - iv. 0x10 0x3E 0x56 Byte3(CRC) 0x12 Byte5(CRC)；//Write Unseal_Key_Step1。
 - v. 0x10 0x3E 0x21 Byte3(CRC) 0x12 Byte5(CRC)；//Write Unseal_Key_Step1。
 - vi. 0x10 0x12 0x11 Byte3(Rdata) Byte4(CRC) Byte5(Rdata) Byte6(CRC)；//Check Sec1 Sec0。
3. 如果器件处于完全访问模式，则进入 CONFIG_UPDATE 模式 - (子命令 0x0090) 。
 - a. 通过写入子命令 0x0090 进入 CONFIG_UPDATE 模式。
 - i. 0x10 0x3E 0x90 Byte3(CRC) 0x00 Byte4(CRC)
4. 在具有 CRC 和校验和的数据存储器中配置寄存器设置。
 - a. 将数据存储器配置为 0x9180 至 0x925D，如数据存储器表中的 [BQ76942 技术参考手册](#) 所示。
 - i. 0x10 0x3E 0x80 Byte3(CRC) 0x91 Byte5(CRC) Byte6(Wdata) Byte7(CRC) Byte8(Wdata) Byte8(CRC)。
 - ii. 0x10 0x60 Byte2(CheckSum) Byte3(CRC) Byte4(DataLength) Byte5(CRC)。
 - iii. 0x10 0x3E 0x82 Byte3(CRC) 0x91 Byte5(CRC) Byte6(Wdata) Byte7(CRC) Byte8(Wdata) Byte8(CRC)。
 - iv. 0x10 0x60 Byte2(CheckSum) Byte3(CRC) Byte4(DataLength) Byte5(CRC)。
 - v. 0x10 0x3E 0x5D Byte3(CRC) 0x92 Byte5(CRC) Byte6(Wdata) Byte7(CRC) Byte8(Wdata) Byte8(CRC)。
 - vi. 0x10 0x60 Byte2(CheckSum) Byte3(CRC) Byte4(DataLength) Byte5(CRC)。
5. 退出 CONFIG_UPDATE 模式。
 - a. 通过写入子命令 0x0092 退出 Config_Update 模式。
 - i. 0x10 0x3E 0x92 Byte3(CRC) 0x00 Byte5(CRC)。
6. 读取数据存储器寄存器以验证所有参数均已成功写入。
 - a. 将数据存储器寄存器读取为 0x9180 至 0x925D，如数据存储器表中的 [BQ76942 技术参考手册](#) 所示。
 - i. 0x10 0x3E 0x80 Byte3(CRC) 0x91 Byte5(CRC)。
 - ii. 0x10 0x40 0x11 Byte3(Rdata) Byte4(CRC) Byte5(Rdata) Byte6(CRC)。
 - iii. 0x10 0x3E 0x82 Byte3(CRC) 0x91 Byte5(CRC)。
 - iv. 0x10 0x40 0x11 Byte3(Rdata) Byte4(CRC) Byte5(Rdata) Byte6(CRC)。
 - v. 0x10 0x3E 0x5B Byte3(CRC) 0x92 Byte5(CRC)。
 - vi. 0x10 0x40 0x11 Byte3(Rdata) Byte4(CRC) Byte5(Rdata) Byte6(CRC)。
 - vii. 0x10 0x3E 0x5D Byte3(CRC) 0x92 Byte5(CRC)。
 - viii. 0x10 0x40 0x11 Byte3(Rdata) Byte4(CRC) Byte5(Rdata) Byte6(CRC)。
7. 进入 CONFIG_UPDATE 模式。
 - a. 通过写入子命令 0x0090 进入 CONFIG_UPDATE 模式。
 - i. 0x10 0x3E 0x90 Byte3(CRC) 0x00 Byte5(CRC)。
8. 检查电池状态 [OTPB] 位是否清除，以验证是否满足 OTP 编程条件。

- a. 读取 0x12 寄存器，以获取 OTPB 位状态。
 - i. 0x10 0x12 0x11 Byte3(Rdata) Byte4(CRC) Byte5(Rdata) Byte6(CRC)。
9. 读取 OTP_WR_CHECK() (子命令 0x00A0)。如果返回值为 0x80，则满足 OTP 编程条件。
 - a. 写入子命令 0x00A0 并从 0x40 读取，以获取 0x00A0 的数据。
 - i. 0x10 0x3E 0xA0 Byte3(CRC) 0x00 Byte5(CRC)。
 - ii. 0x10 0x40 0x11 0x80 Byte4(CRC)。
10. 如果 OTP_WR_CHECK 指示满足条件，则发送 OTP_WRITE() 子命令 (0x00A1)。
 - a. 写入子命令 0x00A1，以发送 OTP_WRITE()。
 - i. 0x10 0x3E 0xA1 Byte3(CRC) 0x00 Byte5(CRC)。
11. 等待 100 ms。从 0x40 读取以检查 OTP 编程是否成功 (0x80 表示成功)。
 - a. 从 0x40 读回 0x00A1 子命令数据。
 - i. 0x10 0x40 0x11 0x80 Byte4(CRC)。
12. 进入密封模式。
 - a. 写入 0x0030 进入密封模式，并读取 0x12 寄存器，以确保器件进入密封模式。
 - i. 0x10 0x3E 0x30 Byte3(CRC) 0x00 Byte5(CRC)。
 - ii. 0x10 0x12 0x11 Byte3(Rdata) Byte4(CRC) Byte5(Rdata) Byte6(CRC)。
13. 退出 CONFIG_UPDATE 模式 - (子命令 0x0092)。
 - a. 0x10 0x3E 0x92 Byte3(CRC) 0x00 Byte5(CRC)。

5 总结

本应用手册逐步介绍了 BQ769X2 OTP 编程流程，还介绍了硬件设置和示例软件命令格式。通过参考本应用手册，用户可以有效地设置硬件和软件元素，以在产品线上完成 OTP 配置。

6 参考资料

1. 德州仪器 (TI)，[BQ769x2 校准和 OTP 编程指南](#)，应用手册
2. 德州仪器 (TI)，[BQ76942 技术参考手册](#)，技术参考手册
3. 德州仪器 (TI)，[BQ769X2 软件开发指南](#)，应用手册
4. 德州仪器 (TI)，[基于 MSPM0 通过 I2C 的 BQ769X2 控制](#)，应用手册
5. 德州仪器 (TI)，[BQ76952：无法退出密封模式](#)，网站

重要通知和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的相关应用。严禁以其他方式对这些资源进行复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
版权所有 © 2025，德州仪器 (TI) 公司