Application Note

Method to Enhance Authentication Security of BQ26100

Hugo Zhang

ABSTRACT

In most electronic products, the system vendor needs to ensure some of sub-systems/sub-units (slave) in the system are original and qualified. For example, the smart phone or notebook maker need to ensure the battery pack are original and qualified. This process is to secure their commercial interests, and secure their brand to avoid any artifact issues, because it will hurt their brand greatly once there is any critical safety issue in the field, for example, smart phone is smoking, catching on fire, or even exploded. So, the system vendor needs to authenticate the sub-systems/sub-units. TI BQ26100 is one dedicated authentication IC, with SHA-1 algorithm. This application note introduces a novel method to enhance the authentication security of BQ26100.

Table of Contents

List of Figures

Trademarks

All trademarks are the property of their respective owners.

1 Introduction

There are many methods to authenticate the slave. Some methods are listed below.

One simple method is to add one resistor in the slave. The host detects its resistance. And if the resistance is in the expected range, then the host will consider the slave is qualified. This method is simple and low cost, but easy for cheap copy.

Another method is to add one EEPROM device (like TI BQ2022A) in the slave. EEPROM stores some specific data. The host read the data from EEPROM, and compare with the data stored in the host. If they are matched, the host will consider the slave is qualified. But the data is transmitted transparently via communication line. So, it is easy to capture the data, and also easy for cheap copy.

The more complicated method is to add authentication device with SHA-1/SHA-256 algorithm in the slave. The host will generate a serial of random data (challenge), and send to the slave. Both host and slave will use the same challenge, key and function to calculate the result (digest). Then the host will read the digest from the slave, and compare with the digest that host itself calculates. If they are matched, the host will consider the slave is qualified. With this method, it is not easy for cheap copy.

The following section describes the BQ26100 and SHA-1 in detail.

2 BQ26100 and SHA-1 Introduction

Figure 2-1 is the simplified schematic of BQ26100. BQ26100 applies single wire communication SDQ. SDQ pin is a multi-functional pin: communication pin and power supply pin.

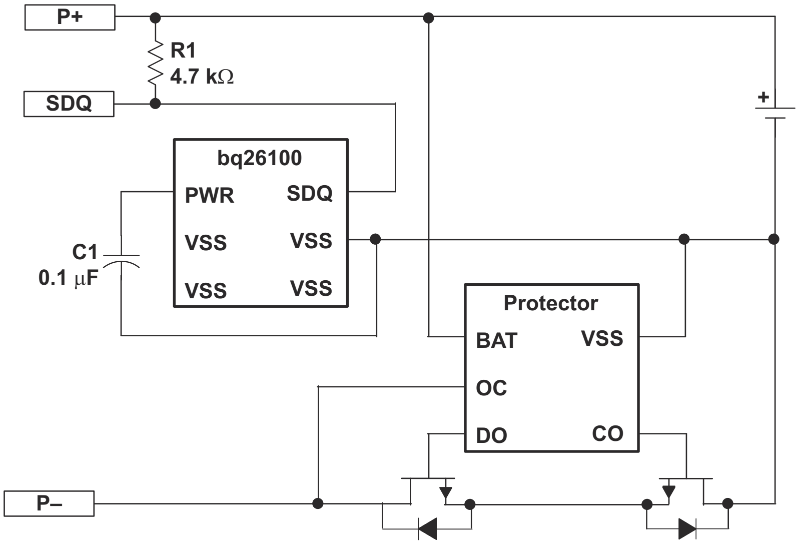


Figure 2-1. Simplified Schematic of BQ26100

BQ26100 embeds SHA-1 authentication algorithm. Figure 2-2 shows the flow of SHA-1 authentication.

1. The host and slave store the same 16 bytes key.
2. The host generates 20 bytes random number as challenge. And also send to the slave.
3. Both the host and slave use the same key, the same challenge, the same function SHA-1 to calculate the results: 20 bytes digest.
4. The host read the result from the slave.
5. Host compares the two results.
6. If the two digests are matched, then consider the slave is qualified.

From the steps above, we can see that the 20 bytes challenge and 20 bytes digest are detectable, and are transmitted transparently through communication line, as shown in Figure 2-2 with green arrows. The third party can easily detect those data with logic analyzer. But theoretically, even they know the challenge and digest, they cannot deduce the SHA-1 key. This is the advantage of SHA-1 authentication.

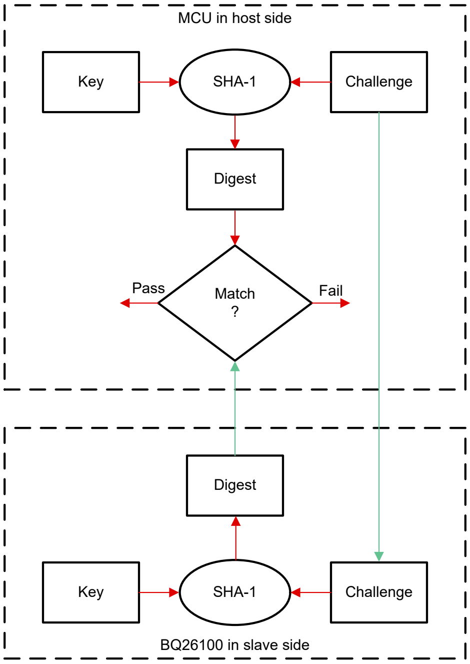


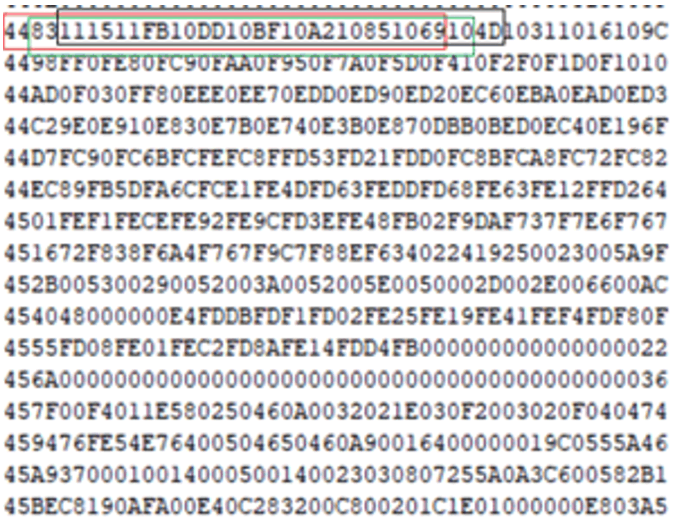
Figure 2-2. Typical Flow of SHA-1 Authentication

To enhance the security level of SHA-1 authentication, one simple method is that the host use the first 20 bytes digest as another 20 bytes challenge and run authenticate for the second time.

3 Risk of Using BQ26100 SHA-1 for Authentication

BQ26100 provides a secure enough authentication. But from the description in Section 2, we know that the host (MCU) also need to store the 16 bytes key in the flash. But the MCU flash might be hacked physically and then read out. Once they get all the data in the flash of MCU, then they can try the limited combinations of 16 consecutive bytes data as challenge, with limited time and resource, and find the correct keys, as shown in the Figure 3-1. Figure 3-1 illustrates the partial data in MCU flash.

For example, the first 16 bytes 0x4483111511FB10DD10BF10A210851069 will be tried first. If authentication is failed, then try 0x83111511FB10DD10BF10A21085106910. If still failed, then try 0x111511FB10DD10BF10A210851069104D, and so on.



```
4483111511FB10DD10BF10A210851069104D10311016109C
4498FF0FE80FC90FAA0F950F7A0F5D0F410F2F0F1D0F1010
44AD0F030FF80EEE0EE70EDD0ED90ED20EC60EBA0EAD0ED3
44C29E0E910E830E7B0E740E3B0E870DBB0BED0EC40E196F
44D7FC90FC6BFCFEFC8FFD53FD21FDD0FC8BFCA8FC72FC82
44EC89FB5DFA6CFCE1FE4DFD63FEDDFD68FE63FE12FFD264
4501FEF1FECEFE92FE9CFD3EFE48FB02F9DAF737F7E6F767
451672F838F6A4F767F9C7F88EF634022419250023005A9F
452B005300290052003A0052005E0050002D002E006600AC
454048000000E4FDDBFDF1FD02FE25FE19FE41FEF4FDF80F
4555FD08FE01FEC2FD8AFE14FDD4FB000000000000000022
456A000000000000000000000000000000000000000000036
457F00F4011E580250460A0032021E030F2003020F040474
459476FE54E76400504650460A90016400000019C0555A46
45A93700010014000500140023030807255A0A3C600582B1
45BEC8190AFA00E40C283200C800201C1E01000000E803A5
```

Figure 3-1. Partial Data in MCU Flash

4 Enhance the Security Level of BQ26100 SHA-1 Authentication

As a dedicated authentication device, one advantage of BQ26100 is that it can prevent physical hack. This is achieved by the dedicated and special process of BQ26100. So, we can make use of this advantage to enhance the security level of BQ26100 SHA-1 authentication. Figure 4-1 shows a modified flow of SHA-1 authentication.

In the modified flow of SHA-1 authentication, there is one more BQ26100 in the host side. With this additional BQ26100, the MCU in the host side doesn’t need to store the keys in the flash. This can avoid the risk of hacking in the MCU to get the keys. Thus, can enhance the authentication security of BQ26100.

The flows are:

1. Both BQ26100 in the host side and slave side store the same keys.
2. MCU generates 20 bytes random number as challenge. And also send to both BQ26100, as shown in Figure 4-1 with green arrows
3. Both BQ26100 in the host side and slave side use the same key, the same challenge, the same function SHA-1 to calculate the 20 bytes digest respectively.
4. MCU read the digests from both BQ26100 in the host side and slave side respectively, as shown in Figure 4-1 with blue arrows.
5. Host does not need to calculate the digest, but just compares the two returned digests.
6. If the two digests are matched, then consider the slave is qualified.

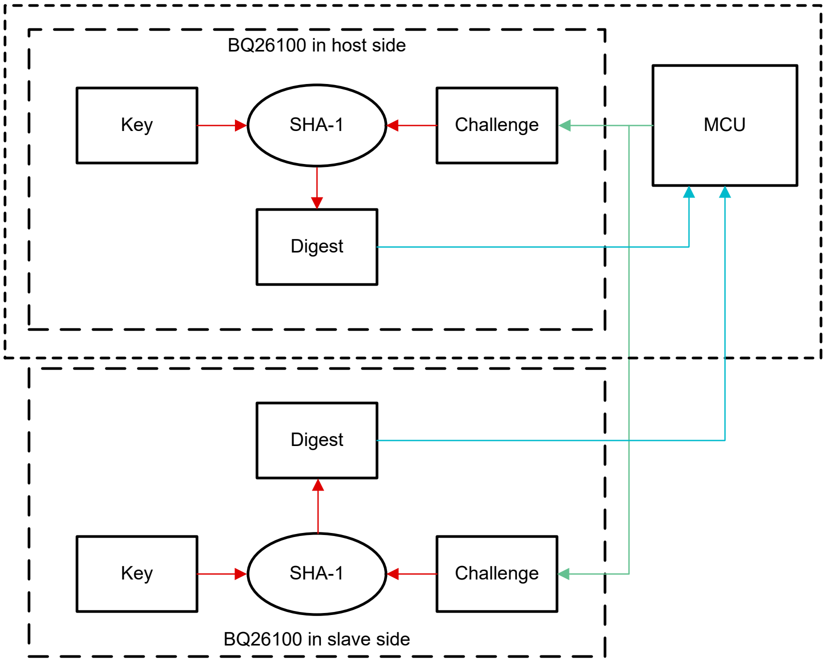


Figure 4-1. Modified Flow of SHA-1 Authentication

5 Summary

In this application note, a modified method to enhance authentication security of BQ26100 is introduced.

One more comment on this: almost all of the encryption and authentication technology might be hacked with some special methods. It depends on how much resource, money, and time it takes. The essential purpose of encryption and authentication are to raise the technical barrier and the cost barrier to a high enough level, so that the third party cannot get enough benefit from hacking and artifact.

6 References

- Texas Instruments, *bq26100 SHA-1/HMAC-based Security and Authentication IC with an SDQ Interface*, data sheet
- Texas Instruments, *How to Implement SHA-1/HMAC Authentication for bq26100*, application note

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), 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 will 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 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.

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

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