



Technical Note

NAND Flash Security

Overview

As the demand for NAND Flash memory continues to accelerate and application diversity expands, the need for more sophisticated NAND Flash security features grows as well. Micron meets these growing security requirements with two NAND Flash security features: unique ID and one-time programmable (OTP) area.

The unique ID is a 64-bit serial number programmed into every NAND Flash device that Micron manufactures. The unique ID is factory set and cannot be replaced or modified in any manner. By using a 64-bit serial number, Micron is able to generate and guarantee 2^{64} , or 1.8×10^{19} , possible serial number combinations.

For more information regarding the unique ID, please contact a Micron representative.

The Micron OTP area is 10 full pages and is left for users to program as they desire. To read, program, and protect the OTP area, please refer to the appropriate Micron NAND Flash memory data sheet.



Security Solutions

Micron NAND Flash memory with unique ID and OTP offers designers options for building a more robust and reliable system. By utilizing the security techniques outlined in this document, designers can provide two additional levels of system security to help reduce the risk associated with system fraud, piracy, and intellectual property theft.

The simplest security solution involves using the unique ID in conjunction with other critical system-component serial numbers, such as those of the microprocessor and/or ASIC device, to employ a serial number verification technique called component authentication.

Component authentication is a helpful security solution in that it reduces the threat associated with component swapping. Systems that do not employ some type of component authentication are more susceptible to having critical system components swapped out with unapproved components, thereby compromising the original system design and possibly rendering the system inoperative.

If all critical system-component serial numbers (the unique ID) are stored somewhere within the system code, these serial numbers can be verified at boot time. If any of the component serial numbers do not match the previously stored serial numbers, an error could be generated, and the system can be checked for swapped components. Designers should design systems so they will not boot until the situation is resolved.

Although component authentication is an effective means of preventing critical system-component swapping, it does not protect the system code. To help protect the system code, Micron recommends a more sophisticated security solution called code authentication.

Code authentication is a helpful security solution in that it provides a mechanism to verify code integrity. Verifying code integrity ensures that the system code has not been modified from its original form. Systems that do not utilize some type of code authentication are more susceptible to unwanted code changes. Any code changes could compromise the original system design, reduce overall system performance, and potentially render the system inoperative.

The Micron OTP area can be used to store and protect values for code authentication. Customers can use a hash-type algorithm to generate an original, one-of-a-kind digest number and store it in the OTP area. This digest number is then retrieved during the system boot-code verification process.

At boot-up, the system uses the same algorithm to recompute the digest number from the current system code. If the two digest numbers do not match, the system code has been compromised and the system should be prohibited from booting until the situation is resolved.

By utilizing Micron NAND Flash security features to invoke both component and code authentication security solutions, system designers can be assured that their system is protected with the most sophisticated security solutions available.



Component Authentication

Component authentication matches stored critical system-component serial numbers against current component serial numbers detected on system boot. Matching the serial numbers protects against component swapping by prohibiting the system from booting if any critical system-component serial number has changed.

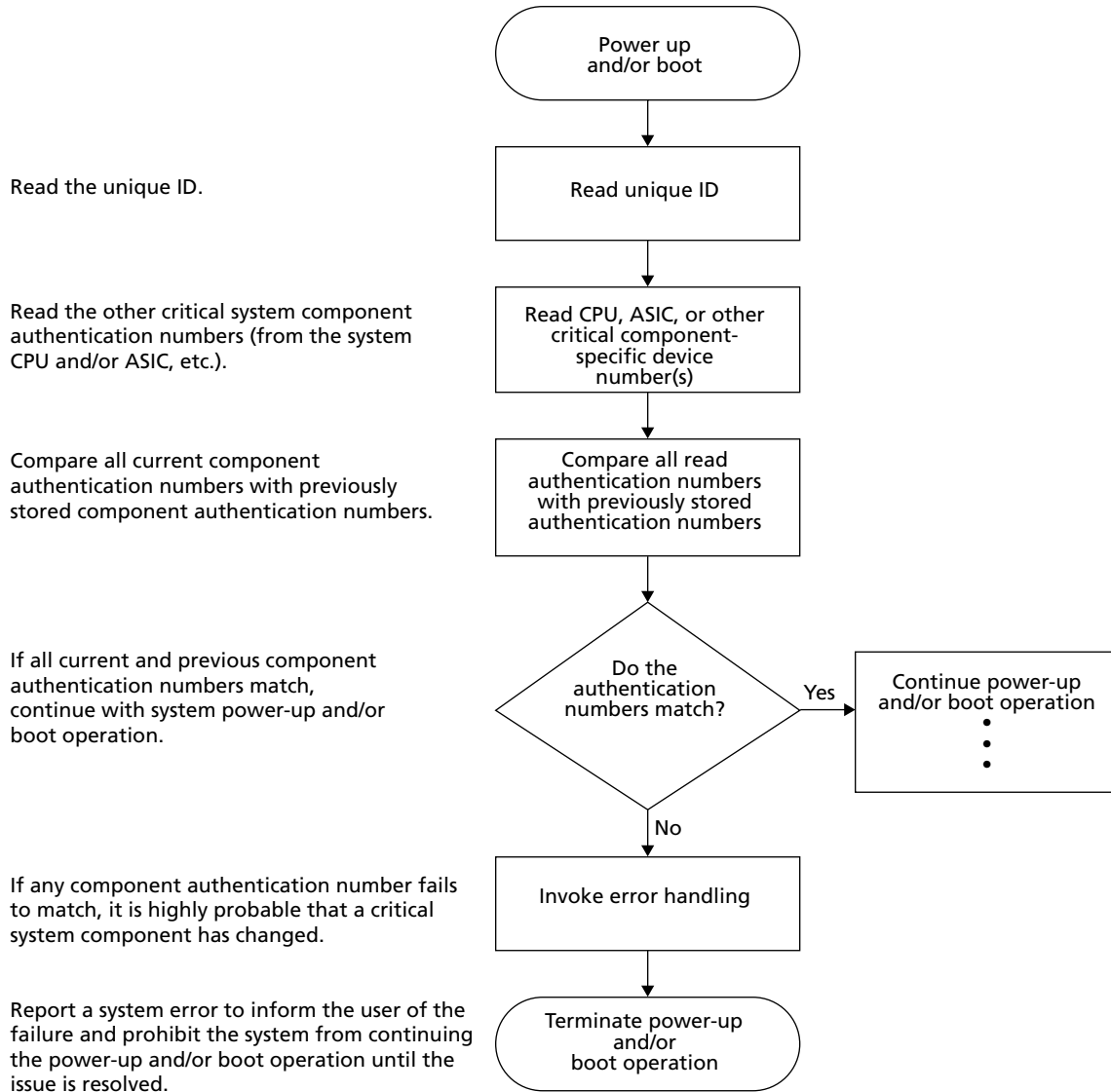
To implement a component authentication security solution, store the serial number for each critical system component somewhere in the system code. During the boot process, read the stored serial numbers and compare them to the boot-read serial numbers.

If all critical system-component serial numbers match, continue the boot process. If any critical system-component serial number does not match, it is highly probable that a critical system component has been swapped. If this occurs, prohibit the system from booting. See Figure 1 on page 4 for a component authentication flowchart.

By implementing a component authentication security solution using the unique ID, designers can reduce the risk of system fraud, piracy, and intellectual property theft.



Figure 1: Component Authentication Flow





Code Authentication

Code authentication is a code integrity check to ensure that the original version of software (system code) has not been modified or replaced. A hash algorithm can be applied to the original system code to protect the system from code tampering.

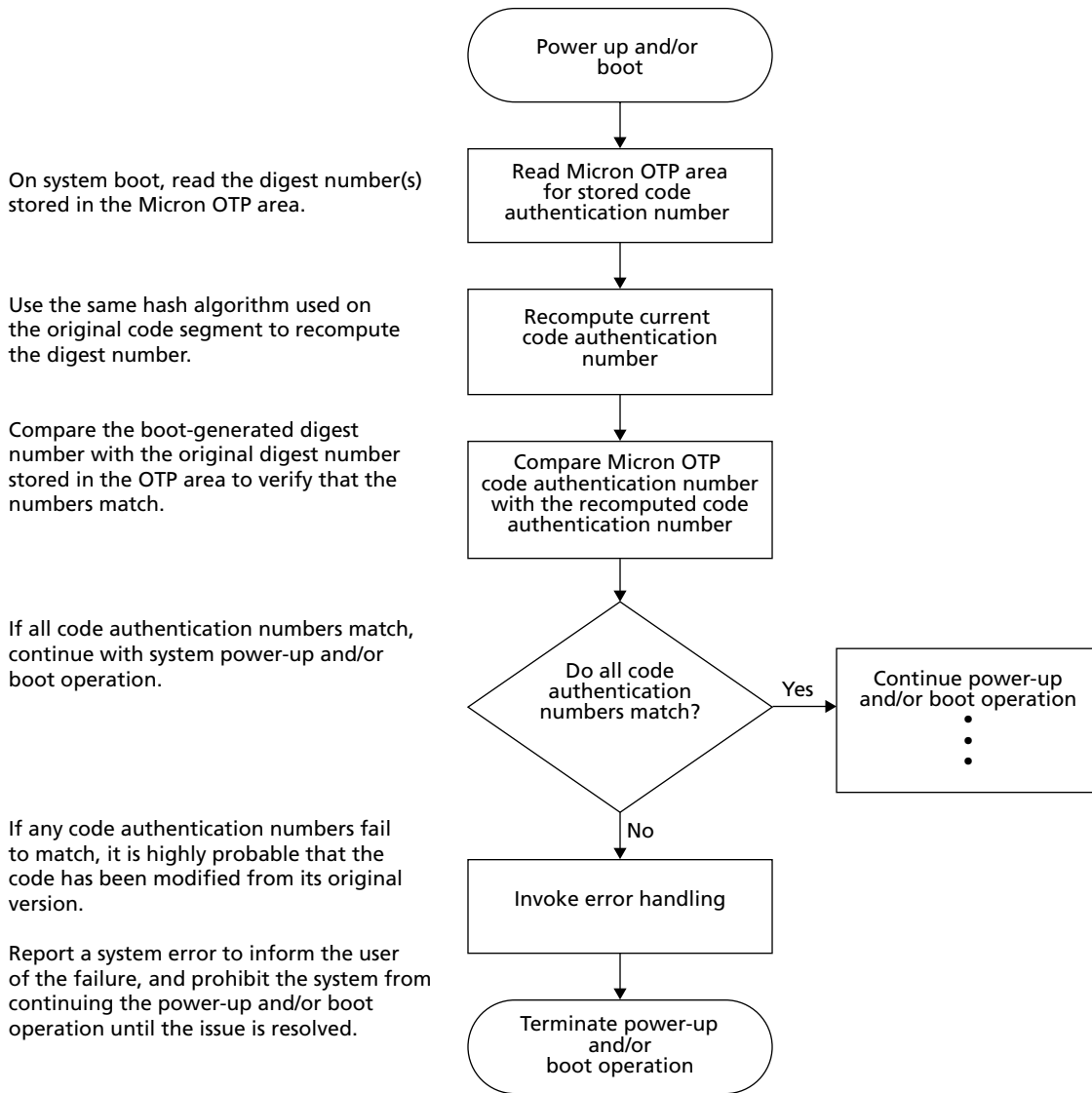
To implement a code authentication security solution, apply a hash algorithm (see “Hashing” beginning on page 7) to the original system code and program the hash output—the digest number—into the Micron OTP area. During boot operations, use the same hash algorithm on the same piece of code to recompute the digest number.

Compare the boot (recomputed) digest number(s) with the original OTP digest number(s). If the digest numbers match, continue the boot process. If the digest numbers fail to match, it is highly probable that the system code has been modified from its original state. If this occurs, prohibit the system from booting. See Figure 2 on page 6 for a code authentication flowchart.

By implementing a code authentication security solution using the Micron NAND OTP security feature, designers can reduce the risk of system fraud, piracy, and intellectual property theft.



Figure 2: Code Authentication



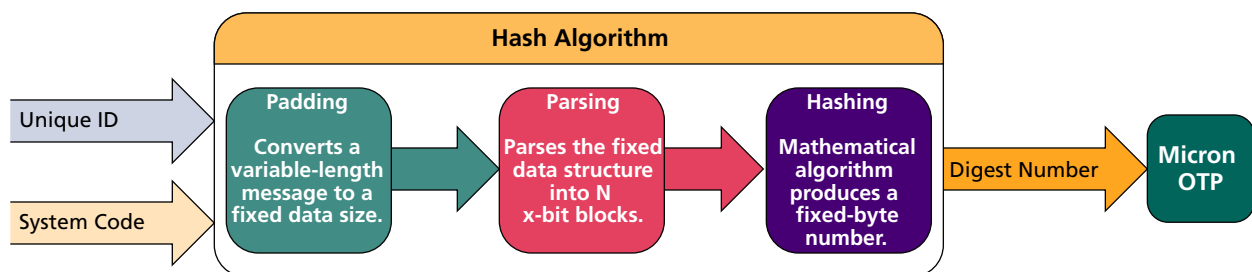
Note: Prior to releasing system code to production, customers must be sure to generate the digest number(s) and program it (them) into the OTP area. Customers also have the option to protect the OTP area against further programming using OTP protect features. Please contact your Micron representative for OTP details.

Hashing

To create a code authentication or digest number, apply a hash algorithm to the original version of code, then program this multibyte number into the Micron OTP area. For improved code protection, include the unique ID as part of the hash algorithm input string (see Figure 3).

Including the unique ID as part of the hash input string provides an extra layer of protection by guaranteeing that although the code may be duplicated, the code authentication number cannot be duplicated because the unique ID is just that—unique. Thus, by design, a hash solution using the unique ID guarantees a unique code authentication number.

Figure 3: Hash Algorithm



These factors contribute to hash selection:

- Desired security level
- Processor capabilities
- Available memory space

Typically, the higher the desired level of security, the stronger the hash algorithm must be. Stronger hash algorithms require a faster, more powerful processor to reduce hash computation time. Stronger hash algorithms also require more code space to store larger initialization variables. Designers should take care to include adequate storage capacity to accommodate the security features in their designs.

There are many types of hash algorithms used in the industry today. Table 1 on page 8 provides some common hash algorithm attributes. From the number of security bits provided for each hash algorithm, SHA-256 is considered to be the most secure hash algorithm, and CRC-16 is considered the least secure hash algorithm.


Table 1: Common Hash Algorithm Attributes

Hash Algorithm	Message Size (bits)	Block Size (bits)	Word Size (bits)	Message Digest (bits)	Security (bits)
CRC-16	$<2^{64}$	—	16	—	—
CDC-32	$<2^{64}$	—	32	—	—
MD-4	$<2^{64}$	512	32	128	64
MD-5	$<2^{64}$	512	32	128	64
SHA-1	$<2^{64}$	512	32	160	80
SHA-256	$<2^{64}$	512	32	256	128

Using the SHA-1 algorithm as an example, the Micron OTP area can store up to 102 different code authentication numbers in a single OTP page, enabling system designers to hash—and therefore protect—multiple sections of critical system code.

OTP Storage Capacity

Micron NAND Flash devices can accommodate multiple digest numbers, enabling designers to use many different sections of the system code in their security implementation. This provides a significant security advantage, as it dramatically decreases the likelihood of a hacker successfully breaching code authentication. For designs employing other hash algorithms, the number of digest numbers accommodated will vary based on the algorithm selected.



Summary

The Micron unique ID and OTP capabilities offer some of the most secure NAND Flash solutions available. By using Micron NAND Flash security features to implement component and code authentication security solutions, designers can protect critical system components and proprietary system software from unwanted attacks and alterations. This added system protection reduces the risk of system fraud, piracy, and intellectual property theft and makes system designs more robust and reliable.

References

The following resources were used in preparing this document:

<http://csrc.nist.gov>

<http://www.bitpipe.com/>

<http://burtleburtle.net/bob/hash/perfect.html>

<http://www.ciphersbyritter.com/GLOSSARY.HTM>

<http://www.networksorcery.com/enp/data/hashng.htm>



**8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900
prodmktg@micron.com www.micron.com Customer Comment Line: 800-932-4992
Micron, the M logo, and the Micron logo are trademarks of Micron Technology, Inc.
All other trademarks are the property of their respective owners.**



Revision History

Rev. B 5/07

- Changed “Micron unique ID” to “unique ID.”
- “Overview” on page 1: Revised description.
- “Security Solutions” on page 2: Revised description.
- Figure 2 on page 6: Revised step descriptions.
- “Hashing” on page 7: Revised description.
- “OTP Storage Capacity” on page 8: Revised description.
- Former Figure 4: “Micron OTP Storage Capacity Example” on page 8: Removed figure.

Rev. A 9/05

- Initial release.