



US 20210051010A1

(19) **United States**(12) **Patent Application Publication****Yang et al.**(10) **Pub. No.: US 2021/0051010 A1**(43) **Pub. Date: Feb. 18, 2021**(54) **MEMORY DEVICE PROVIDING DATA SECURITY****H04L 9/32** (2006.01)**G06F 21/60** (2006.01)(71) Applicant: **PUFsecurity Corporation**, Hsinchu County (TW)(52) **U.S. Cl.**CPC **H04L 9/0894** (2013.01); **G06F 3/0622** (2013.01); **G06F 3/0637** (2013.01); **G06F 7/588** (2013.01); **H04L 9/3278** (2013.01); **H04L 9/0869** (2013.01); **G06F 21/602** (2013.01); **G06F 3/0673** (2013.01)(72) Inventors: **Ching-Sung Yang**, Hsinchu County (TW); **Meng-Yi Wu**, Hsinchu County (TW); **Chia-Cho Wu**, Hsinchu County (TW)(21) Appl. No.: **16/896,244**

(57)

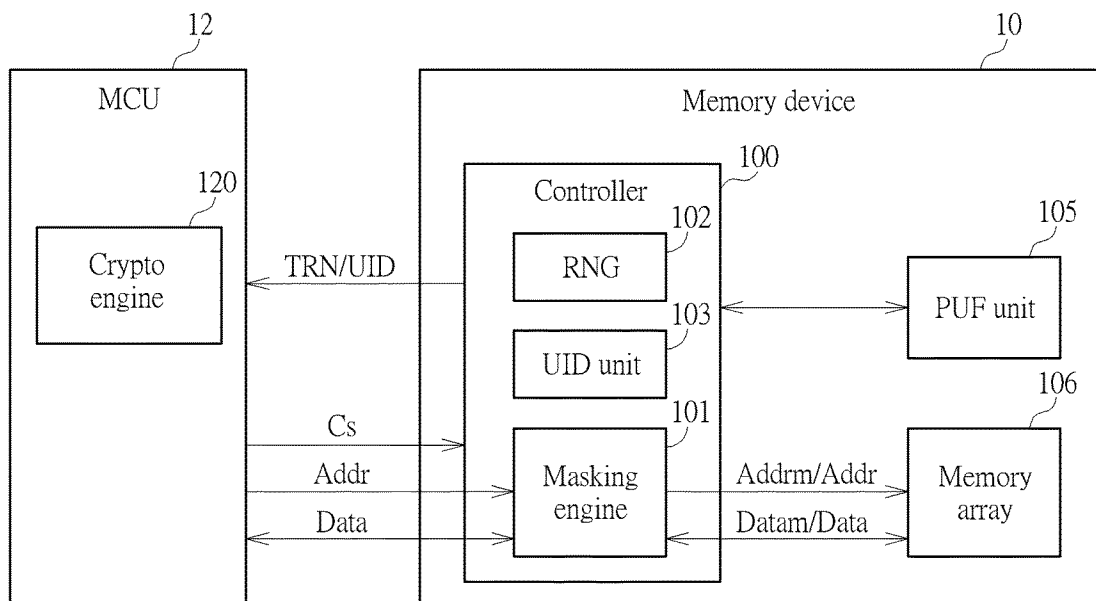
ABSTRACT(22) Filed: **Jun. 9, 2020****Related U.S. Application Data**

(60) Provisional application No. 62/887,679, filed on Aug. 16, 2019.

Publication Classification(51) **Int. Cl.****H04L 9/08** (2006.01)**G06F 3/06** (2006.01)

A memory device includes a physically unclonable function (PUF) unit, a controller and a memory array. The PUF unit is used to provide a random bit pool. The controller is coupled to the PUF unit and is used to extract a random bit sequence from the random bit pool. The controller includes a masking engine. The masking engine is used to perform a key derivation function to stretch the extracted random bit sequence and to mask an input signal. The memory array is coupled to the masking engine and is used to store according to the masked input signal.

1



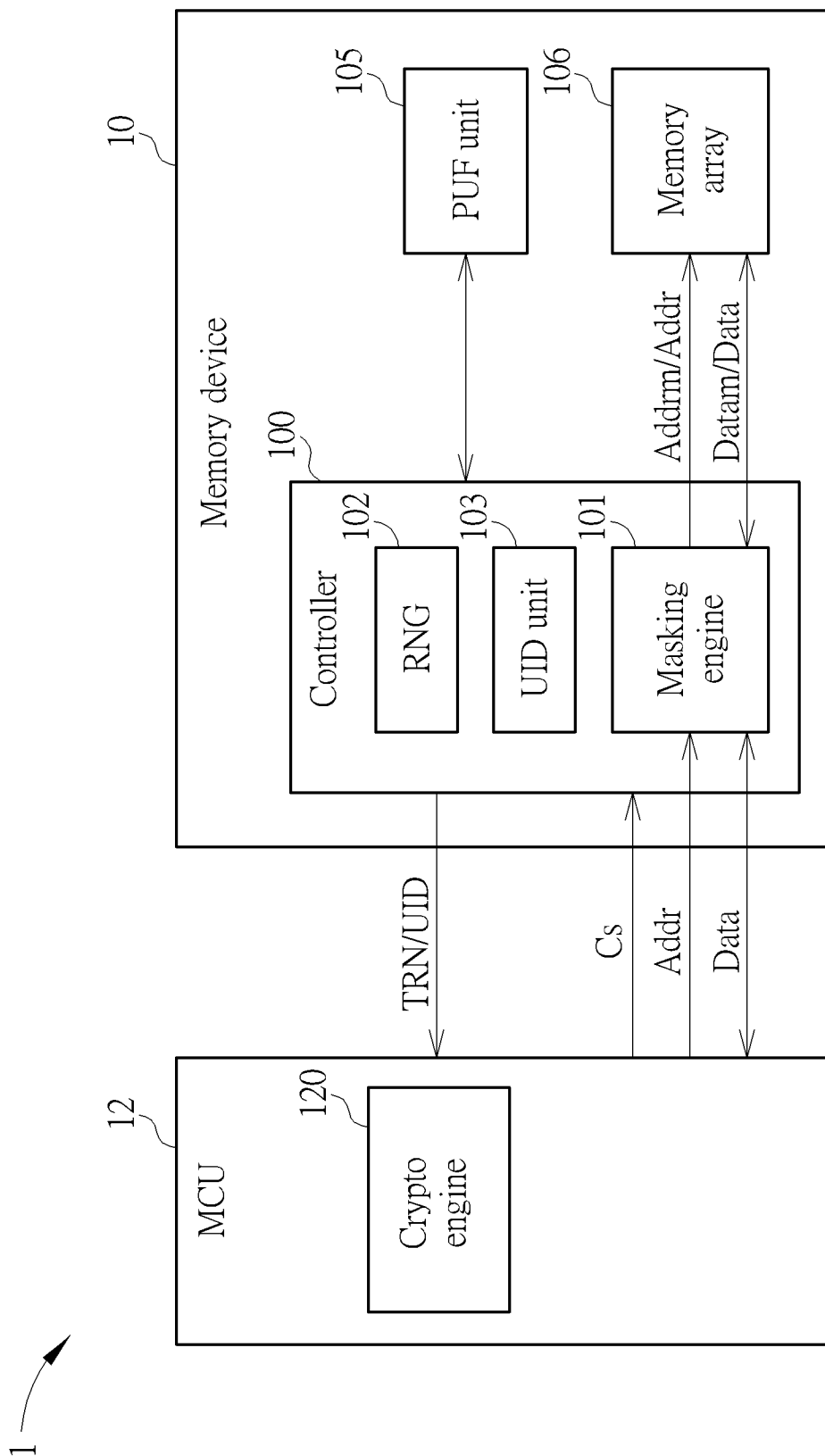


FIG. 1

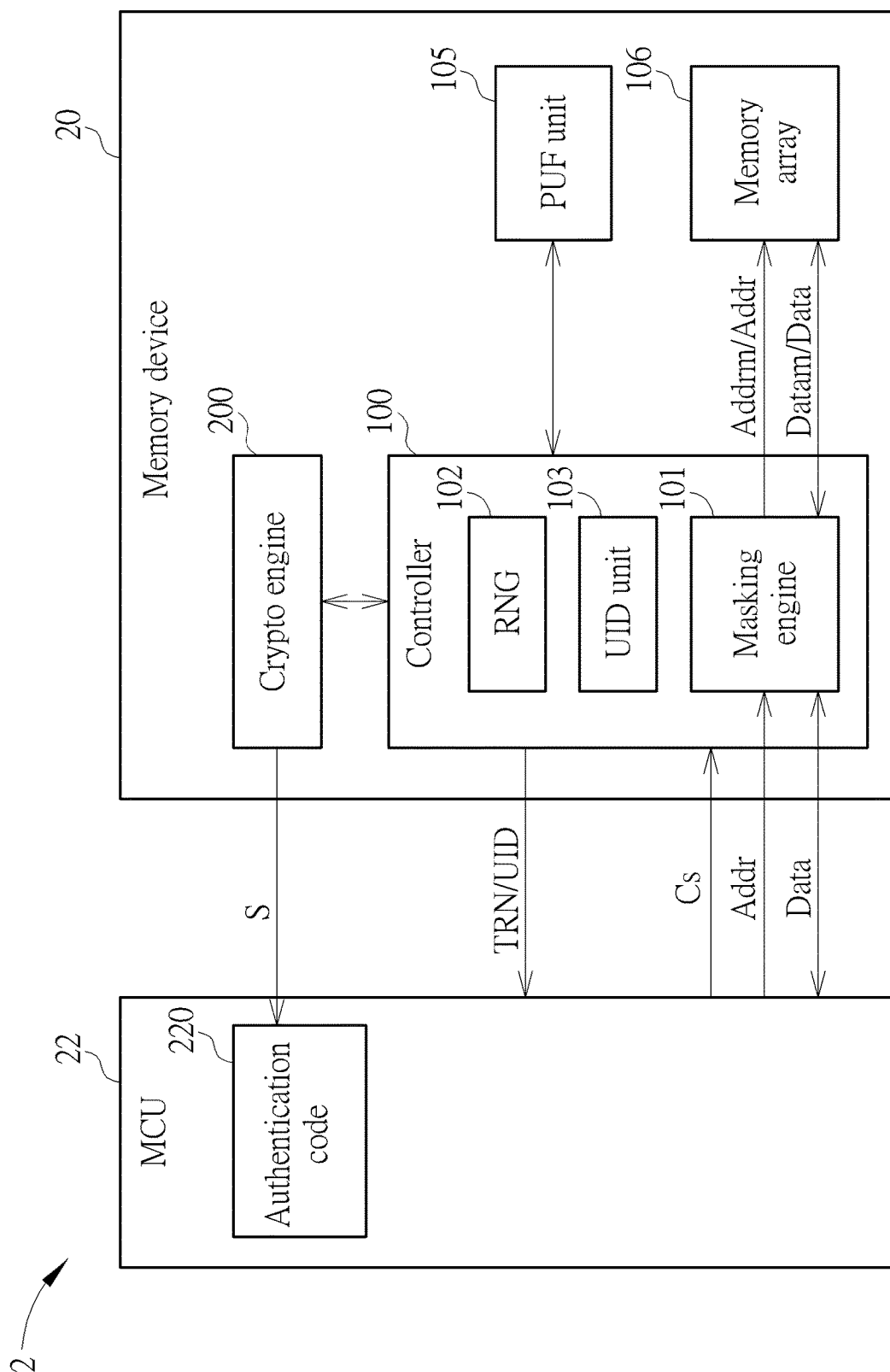


FIG. 2

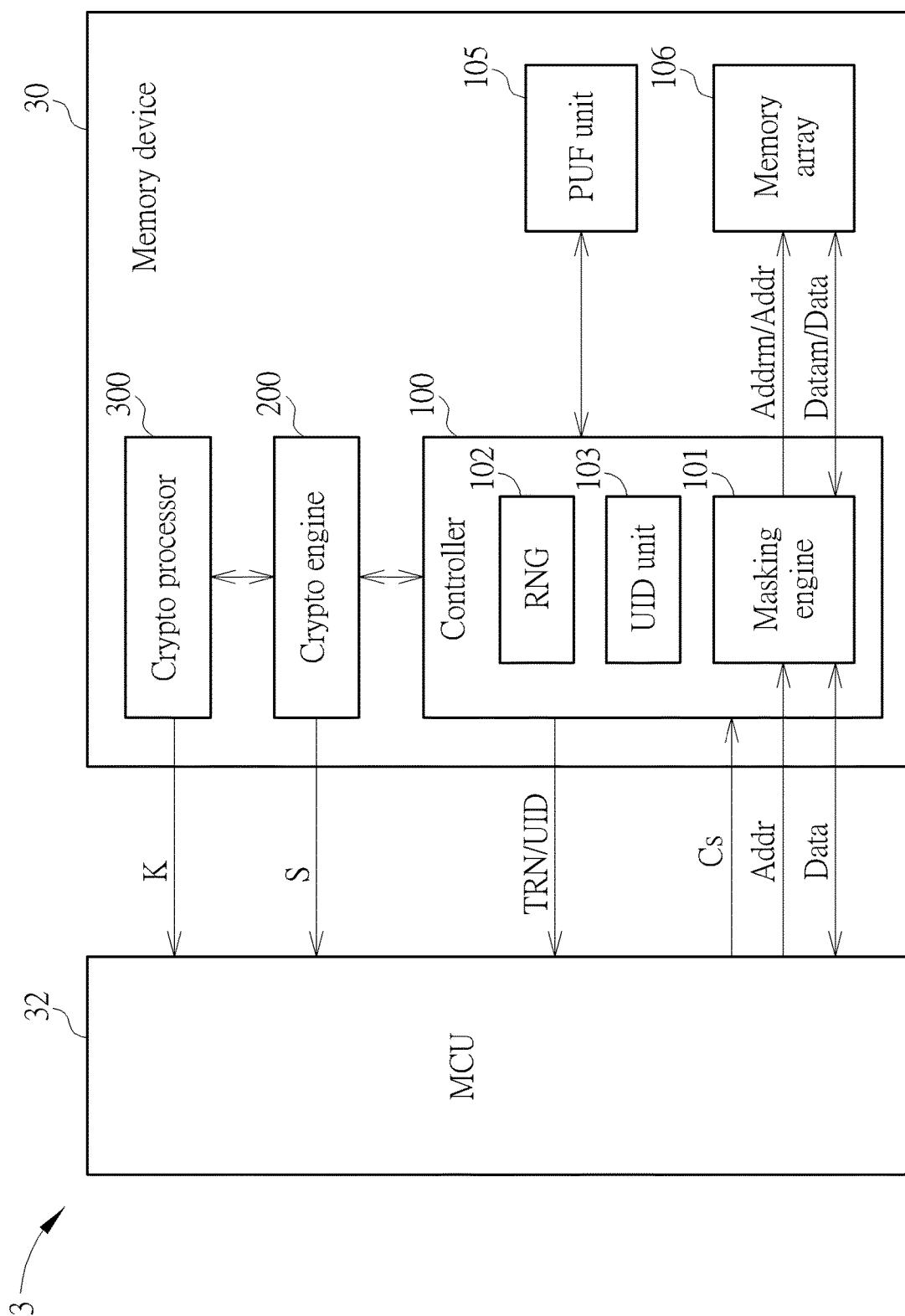


FIG. 3

MEMORY DEVICE PROVIDING DATA SECURITY

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This non-provisional application claims priority of U.S. Provisional Patent Application No. 62/887,679, filed on 16 Aug. 2019, included herein by reference in its entirety.

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0002] The invention relates to memory devices, and in particular, to a memory device providing data security.

2. Description of the Prior Art

[0003] Information security has become a great concern of electronic circuits as information technology and communication technology advance. Information security involves preventing unauthorized data access, use, modification, inspection and recording. It is important to maintain information security for memory devices.

SUMMARY OF THE INVENTION

[0004] According to an embodiment of the invention, a memory device includes a physically unclonable function (PUF) unit, a controller and a memory array. The PUF unit is used to provide a random bit pool. The controller is coupled to the PUF unit and is used to extract a random bit sequence from the random bit pool. The controller includes a masking engine. The masking engine is used to perform a key derivation function to stretch the extracted random bit sequence and to mask an input signal. The memory array is coupled to the masking engine and is used to store according to the masked input signal.

[0005] These and other objectives of the present invention will no doubt become obvious to those of ordinary skill in the art after reading the following detailed description of the preferred embodiment that is illustrated in the various figures and drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] FIG. 1 is a block diagram of a cryptographic system according to an embodiment of the invention.

[0007] FIG. 2 is a block diagram of a cryptographic system according to another embodiment of the invention.

[0008] FIG. 3 is a block diagram of a cryptographic system according to another embodiment of the invention.

DETAILED DESCRIPTION

[0009] As used herein, the term “true random” refer to a bit stream or a data sequence that is substantially 50% in a hamming weight and an inter-device (ID) hamming distance, and is substantially 1 in a minimum entropy (min-entropy).

[0010] FIG. 1 is a block diagram of a cryptographic system 1 according to an embodiment of the invention. The cryptographic system 1 may include a memory device 10 and a microprocessor (MCU) 12 coupled thereto. The cryptographic system 1 may be applicable in an internet of things (IoT) network. The MCU 12 may acquire data from an external device or access data from the memory device 10.

The external device may be a sensor or a network. The memory device 10 may provide security functions including secure data storage, unique identity generation, true random number generation and secure key storage, thereby saving data processing resources of the MCU 12, protecting data from unauthorized access and enhancing data security.

[0011] The MCU 12 may include a crypto engine 120. The crypto engine 120 may be implemented by software code executable by the MCU 12. The memory device 10 may include a controller 100, a physically unclonable function (PUF) unit 105 and a memory array 106. The controller 100 may be coupled to the MCU 12, the PUF unit 105 and the memory array 106. The controller 100 may include a masking engine 101, a random number generator (RNG) 102 and a unique identifier (UID) unit 103. The masking engine 101, the random number generator 102 and the UID unit 103 may be coupled to the PUF unit 105. The masking engine 101 may be coupled to the memory array 106. The PUF unit 105, the memory array 106 and the controller 100 may form an integrated circuit.

[0012] The crypto engine 120 may perform an authentication process to provide assurance of the authenticity of data access, and hence, to control data access to the memory device 10. Upon verifying an authorized access, the crypto engine 120 may send a security command Cs to the memory device 10 to grant data access to the memory array 106. The memory device 10 may receive the security command Cs, and control the data access to the memory array 106 according to the security command Cs. The memory array 106 may be a NAND flash memory. The data access may be read access and/or write access.

[0013] The PUF unit 105 may store a random bit pool and generate a PUF response from the random bit pool in response to a PUF challenge. The OTP memory may be antifuse-based and the random bit pool may be programmed into the OTP memory during manufacturing setup. The random bit pool may include a plurality of PUF bits that are truly random. The PUF unit 105 may output the PUF response according to a predetermined selecting algorithm. In some embodiments, the PUF unit 105 may select the first 1K PUF bits as the PUF response. In other embodiments, the PUF unit 105 may select PUF bits from rows of memory cells in a predetermined row order, e.g. selecting PUF bits from odd rows in an ascending order, to serve as the PUF response. The PUF unit 105 may include a one-time programmable (OTP) memory. In some embodiments, the one-time programmable memory may be replaced with a non-volatile memory containing a plurality of true random bits. For example, the non-volatile memory may be 64-bit-by-64-bit flash memory cells, and each row, column or diagonal line of the memory cells may contain true random bits. In some embodiments, the plurality of true random bits may be updated regularly.

[0014] The controller 100 may operate the PUF unit 105 and the masking engine 101 to provide secure data storage. Upon receiving the security command Cs of granting data access, the controller 100 may extract a random bit sequence from the random bit pool in the PUF unit 105, the masking engine 101 may perform a key derivation function to stretch the extracted random bit sequence and to mask an input signal with the stretched random bit sequence, and the memory array 106 may store according to the masked input signal. The input signal may include an access address Addr or a data sequence Data. The masking of the input signal

with the stretched random bit sequence may be data masking or address masking, and may involve performing an XOR operation on the stretched random bit sequence and the data sequence Data or the access address Addr in a bitwise manner. In data masking, the masking engine 101 may mask the access address Addr with the stretched random bit sequence to generate a derived key, and mask the data sequence Data with the derived key to generate a masked data sequence Datam, and the memory array 106 may store the masked data sequence Datam at the access address Addr. In some embodiments, the masking engine 101 may store the derived key in a local memory for recovering the masked data sequence Datam in a read operation. For example, in a read operation, the masking engine 101 may read the masked data sequence Datam at the access address Addr, mask the masked data sequence Datam with the derived key to recover the data sequence Data, and transmit the data sequence Data to the MCU 12. In address masking, the masking engine 101 may mask the data sequence Data with the stretched random bit sequence to generate a derived key, and mask the access address Addr with the derived key to generate a masked access address Addr_m, and the memory array 106 may store the data sequence Data at the masked access address Addr_m. In some embodiments, the masking engine 101 may store the derived key in the local memory for reproducing the masked access address Addr_m in a read operation. For example, in a read operation, the masking engine 101 may receive the access address Addr from MCU 12, reproduce the masked access address Addr_m by masking the access address Addr and the derived key, read the data sequence Data at the masked access address Addr_m, and transmit the data sequence Data to the MCU 12. The data masking operation and the address masking operation enhance data security and protect data from unauthorized access.

[0015] The random number generator 102 may generate a true random number. In some embodiments, the crypto engine 120 may send a security command Cs including a request for a true random number to the controller 100, the controller 100 may extract a random bit sequence from the random bit pool in the PUF unit 105 in response to the request, and the random number generator 102 may generate a true random number TRN with the extracted random bit sequence, and transmit the true random number TRN to the crypto engine 120.

[0016] The UID unit 103 may generate a unique identifier. In some embodiments, the crypto engine 120 may send a security command Cs including a request for an unique identifier to the controller 100, the controller 100 may extract a random bit sequence from the random bit pool in the PUF unit 105 in response to the request, and the UID unit 103 may generate a unique identifier UID according to the extracted random bit sequence, and transmit the unique identifier UID to the crypto engine 120.

[0017] The PUF unit 105 may provide secure key storage. Specifically, a portion of the OTP memory in the PUF unit 105 may be reserved for storing secure keys. In some embodiments, the crypto engine 120 may send a security command Cs including a request for storing a secure key along with the secure key to the controller 100, and the PUF unit 105 may store secure key in the reserved portion of the OTP memory.

[0018] Since the masking engine 101 may perform data masking and/or address masking on the data sequences

and/or access addresses, the memory device 10 may be used in an execute in place (XIP) method, in which programs are executed directly from the memory array 106 rather than copying the same into a volatile memory, thereby reducing the total amount of memory required.

[0019] The cryptographic system 1 employs the memory device 10 to provide security functions including secure data storage, unique identity generation, true random number generation and secure key storage, saving data processing resources of the MCU 12, enabling XIP operations while protecting data from unauthorized access and enhancing data security.

[0020] FIG. 2 is a block diagram of a cryptographic system 2 according to another embodiment of the invention. The cryptographic system 2 is different from the cryptographic system 1 in that an MCU 22 may include a non-volatile memory for storing authentication code 220, and a memory device 20 may further include a crypto engine 200. The crypto engine 200 may be implemented by a hardware circuit capable of loading the authentication code 220 from the MCU 12 upon power-up and executing the same. The following discussion will focus on the configurations and the operations of the authentication code 220 and the crypto engine 200. The crypto engine 200 may be coupled to the controller 100.

[0021] The crypto engine 200 may execute the authentication code 220 to perform an authentication process. The authentication process may include a sequence of authentication operations. The authentication code 220 may be firmware code for instructing the crypto engine 200 to perform the sequence of authentication operations. In some embodiments, the controller 100 may receive a sequence of security commands Cs from the MCU 22, the sequence of security commands Cs being used to execute the sequence of authentication operations. The controller 100 may instruct the crypto engine 200 to perform the sequence of authentication operations in response to the sequence of security commands Cs, and control the data access to the memory array 106 according to a result of the sequence of authentication operations. The controller 100 may grant the data access to the memory array 106 upon a successful authentication process, and may deny the data access to the memory array 106 upon a failed authentication process.

[0022] The crypto engine 200 may generate an entropy S by using the extracted random bit sequence and/or the true random number TRN. The MCU 22 may send a security command Cs including a request for an entropy to the controller 100. In one embodiment, in response to the request, the controller 100 may extract a random bit sequence from the random bit pool in the PUF unit 105 and instruct the random number generator 102 to generate the true random number TRN, and the crypto engine 200 may mask the true random number TRN with the extracted random bit sequence to generate the entropy S, and transmit the entropy S to the MCU 12. The true random number TRN, the extracted random bit sequence and the entropy S may be equal in length. In another embodiment, the crypto engine 200 may generate the entropy S by combining a plurality of bit in the true random number TRN in a predetermined period, e.g., 3 clock cycles, into an entropy bit, so as to generate the entropy S, and transmit the entropy S to the MCU 12. The entropy S may be shorter in length than that of the true random number TRN. In yet another embodiment, the crypto engine 200 may generate the entropy S by

combining a plurality of bit in the extracted random bit sequence in a predetermined period into an entropy bit, so as to generate the entropy S, and transmit the entropy S to the MCU 12. The entropy S may be shorter in length than that of the extracted random bit sequence.

[0023] Since the crypto engine 200 is implemented by hardware, the authentication process may be performed in a quicker and more efficient manner. Further, since the crypto engine 200 is located in the memory device 20, all authentication data for use in the authentication process may be kept inside the memory device 20 without being exposed to external circuits, enhancing the security level. The cryptographic system 2 employs the crypto engine 200 and the authentication code 220 to increase operation speed and efficiency of the authentication process, reduce the risk of the authentication key from being exposed to external circuits, protect data from unauthorized access, and save data processing resources of the MCU 22.

[0024] FIG. 3 is a block diagram of a cryptographic system 3 according to another embodiment of the invention. The cryptographic system 3 is different from the cryptographic system 2 in that an MCU 32 may not have the authentication code 220, and a memory device 30 may further include a crypto processor 300. The following discussion will focus on the configurations and the operations of the crypto processor 300. The crypto processor 300 may be coupled to the crypto engine 200.

[0025] The crypto processor 300 may include a circuit instructing a sequence of authentication operations, thereby further increasing operation speed and efficiency of the authentication process. The controller 100 may receive a security command Cs to initiate an authentication process. The controller 100 may instruct the crypto processor 300 to initiate the authentication process in response to the security command Cs. In turn, the crypto processor 300 may instruct the crypto engine to perform the sequence of authentication operations. Subsequently, the crypto engine 200 may perform the sequence of authentication operations to generate an authentication result. The controller 100 may control the data access to the memory array 106 according to the authentication result. In particular, the controller 100 may grant the data access to the memory array 106 upon a successful authentication process, and may deny the data access to the memory array 106 upon a failed authentication process.

[0026] The crypto processor 300 may generate a key K by using the entropy S and the extracted random bit sequence. The MCU 32 may send a security command Cs including a request for a key to the controller 100. In response to the request, the controller 100 may extract a random bit sequence from the random bit pool in the PUF unit 105 and instruct the crypto engine 200 to generate the entropy S, and the crypto engine 200 may mask the entropy S with the extracted random bit sequence to generate the key K, and transmit the key K to the MCU 12. The entropy S, the extracted random bit sequence and the key K may be equal in length.

[0027] Since the crypto processor 300 and the crypto engine 200 are both implemented by hardware, the authentication process may be performed in a quicker and more efficient manner. Since the crypto processor 300 and the crypto engine 200 are both located in the memory device 30, all authentication data for use in the authentication process may be kept inside the memory device 30 without being

exposed to external circuits, enhancing the security level. The cryptographic system 3 employs the crypto processor 300 to increase operation speed and efficiency of the authentication process, reduce the risk of the authentication key from being exposed to external circuits, protect data from unauthorized access, and save data processing resources of the MCU 32.

[0028] Those skilled in the art will readily observe that numerous modifications and alterations of the device and method may be made while retaining the teachings of the invention. Accordingly, the above disclosure should be construed as limited only by the metes and bounds of the appended claims.

What is claimed is:

1. A memory device comprising:
 - a physically unclonable function (PUF) unit configured to provide a random bit pool;
 - a controller coupled to the PUF unit and configured to extract a random bit sequence from the random bit pool, and comprising:
 - a masking engine configured to perform a key derivation function to stretch the extracted random bit sequence and to mask an input signal; and
 - a memory array coupled to the masking engine and configured to store according to the masked input signal.
2. The memory device of claim 1, wherein the input signal comprises an access address and a data sequence.
3. The memory device of claim 2, wherein the masking engine masks the access address with the stretched random bit sequence to generate a derived key, and then masks the data sequence with the derived key to generate a masked data sequence.
4. The memory device of claim 3, wherein the memory array stores the masked data sequence at the access address.
5. The memory device of claim 2, wherein the masking engine masks the data sequence with the stretched random bit sequence to generate a derived key, and then masks the access address with the derived key to generate a masked access address.
6. The memory device of claim 5, wherein the memory array stores the data sequence at the masked access address.
7. The memory device of claim 1, wherein the controller further comprises a unique identifier (UID) unit configured to generate an unique identifier according to the extracted random bit sequence.
8. The memory device of claim 1, wherein the controller further comprises:
 - a random number generator coupled to the PUF unit and configured to generate a true random number with the extracted random bit sequence.
9. The memory device of claim 8, further comprising:
 - a crypto engine coupled to the controller, and configured to generate an entropy by using the extracted random bit sequence and/or the true random number.
10. The memory device of claim 9, further comprising:
 - a crypto processor coupled to the crypto engine, and configured to generate keys by using the entropy and the extracted random bit sequence.
11. The memory device of claim 1 wherein the PUF unit, the memory array and the controller are formed in an integrated circuit.
12. The memory device of claim 1, wherein the PUF unit comprises a one-time programmable memory.

13. The memory device of claim 1, wherein the controller is configured to receive a security command, and control data access to the memory array according to the security command.

* * * * *