### BOOT\_ChooseAndLoadSW\_CLSW

#### BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF

This function checks the state of the loaded SIF header file.

Prototype:

void BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF(

const uint32\_t p\_SIFAddress,

const uint32\_t p\_ENVMAddress,

boolean\_t\* p\_FlashReadFailed,

ts\_LoadStatus\* p\_LoadStatus)

Parameters:

Function return : Not used

p\_SIFAddress(R) : SIF Header address

p\_ENVMAddress(R) : ENVM address

p\_FlashReadFailed(W) : Read failure status

p\_LoadStatus(W) : Load status

Calls:

BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF\_SHA\_ENVM

BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF\_SHA\_RMM

BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF\_Header\_FNOR

BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF\_Header\_RMM

LIBUTI\_SHA\_ComputeSHA256

LIBUTI\_MEM\_MemsNotEqual

Preconditions:

None

##### Input Data

Data:

None

Preconditions:

None.

##### Output Data

Data:

SIF\_HeaderBuffer[C\_BOOT\_SIF\_HEADER\_MAX\_LENGTH]: uint8\_t

##### Requirements

REQ\_SDDD\_BOOT\_BOOT\_ChooseAndLoadSW\_CLSW000xx-0x

*[COV.REQ\_BOOT\_SRD-00087; COV.REQ\_BOOT\_SRD-00169; COV.REQ\_BOOT\_SRD-00170]*

BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Causes | | | |
| Effects | [Initialize the BOOT data size] | | | |
| [SIF Header address is stored in FLASH] | | | [SIF Header address is not stored in FLASH] |
| **[**Retrieve the SHA256 of the SIF Header from the PLD Compagnon ENVM**]** | | | **[**Retrieve the SHA256 of the SIF Header from the RMM**]** |
| [The FLASH read is successful during the first read] | | | [The FLASH read is not successful during the first read] |
| [the BOOT data size is in range] | | [The BOOT data size is out of range] | Exit the function |
| [SIF Header address is stored in FLASH] | [SIF Header address is not stored in FLASH] | Set {p\_LoadStatus->OutOfOperationalCondition} to TRUE  And exit the function |
| **[**Retrieve the SIF header from FNOR**]** | **[**Retrieve the SIF header from RMM**]** |
| [The FLASH read is successful during the second read] | [The FLASH read is not successful during the second read] |
| **[**Compute the SHA2\_256 checksum of the data**]**  **[**check the corrupted state of SIF Header**]** | Exit the function |

[Initialize the BOOT data size]: Set {BOOT data size} to {C\_BOOT\_SIF\_HEADER\_MAX\_LENGTH}

[SIF Header address is stored in FLASH]: Bit0 of {p\_SIFAddress} is equal to 0.

**[**Retrieve the SHA256 of the SIF Header from the PLD Compagnon ENVM**]** corresponds to the following call:

**BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF\_SHA\_ENVM**

* **Function return:** Not used
* **IN:** {p\_ENVMAddress}
* **OUT:** address of{Boot Data}
* **IN/OUT:** {p\_FlashReadFailed}

[SIF Header address is not stored in FLASH]: Bit0 of {p\_SIFAddress} is different from 0.

**[**Retrieve the SHA256 of the SIF Header from the RMM**]** corresponds to the following call:

**BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF\_SHA\_RMM**

* **Function return:** Not used
* **IN:** {p\_SIFAddress}
* **OUT:** address of{Boot Data}
* **OUT:** {p\_FlashReadFailed}
* **OUT:** {p\_LoadStatus}

[The FLASH read is not successful during the first read]: {p\_FlashReadFailed} is different from FALSE.

[The FLASH read is successful during the first read]: {p\_FlashReadFailed} is equal to FALSE.

[The BOOT data size is out of range]: {BOOT data size} is strictly higher than {C\_BOOT\_SIF\_HEADER\_MAX\_LENGTH}.

[the BOOT data size is in range]: {BOOT data size} is lower or equal than {C\_BOOT\_SIF\_HEADER\_MAX\_LENGTH}.

**[**Retrieve the SIF header from FNOR**]** corresponds to the following call:

**BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF\_Header\_FNOR**

* **Function return:** Not used
* **IN:** {p\_SIFAddress}
* **OUT:** address of{Boot Data}
* **OUT:** {SIF\_HeaderBuffer}
* **IN/OUT:** {p\_FlashReadFailed}

**[**Retrieve the SIF header from RMM**]** corresponds to the following call:

**BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF\_Header\_RMM**

* **Function return:** Not used
* **IN:** {p\_SIFAddress}
* **OUT:** address of{Boot Data}
* **OUT:** {SIF\_HeaderBuffer}
* **IN/OUT:** {p\_FlashReadFailed}

[The FLASH read is successful during the second read]: {p\_FlashReadFailed} is equal to FALSE.

[The FLASH read is not successful during the second read]: {p\_FlashReadFailed} is different from FALSE.

**[**Compute the SHA2\_256 checksum of the data**]** corresponds to the following call:

**LIBUTI\_SHA\_ComputeSHA256**

* **Function return:** Not used
* **IN:** {SIF\_HeaderBuffer}
* **IN:** {Boot Data size}
* **OUT:** address of{SHA256 Buffer}

**[**check the corrupted state of SIF Header**]:** corresponds to the following call:

**LIBUTI\_MEM\_MemsNotEqual**

* **Function return:** {p\_LoadStatus->Corrupted}
* **IN:** {SHA256 Buffer Data}
* **IN:** {Boot Data in sha256}
* **IN:** {SHA256\_BLOCK\_SIZE}

The read of{SHA256\_BLOCK\_SIZE} is done by 4 bytes.

Traceability: Refined

Rationale:

Mean of verification: Test

[END\_REQ\_SDDD\_BOOT\_CLSW\_LoadSW\_Load\_LoadAndTestSIF\_000xx-0x]