
Hello and welcome to this presentation of the STM32H5 symmetric crypto coprocessors. It covers the features of the AES and SAES modules, which are widely used for cryptographic applications

## AES feature list

- NIST FIPS197 compliant AES implementation
- AES chaining modes
  - Electronic codebook (ECB)
  - Cipher block chaining (CBC)
  - Counter (CTR) mode
  - Galois counter mode (GCM)
  - Galois message authentication code (GMAC)
  - Counter with CBC-MAC (CCM) mode
- AES operation modes on 128-bit data blocks , 128 or 256-bit keys
  - Encryption, Decryption (with associated key derivation mode)
- **Can load shared keys from Secure AES**
- AHB slave with suspend/resume & DMA support (IN + OUT channels)
- 32-bit data words swapping support (bit, byte or half-word)
- **Atomic key writing/loading enforcement**

**Number of cycles required to process a 128-bit block**
**Clock frequency= AHB clock of peripheral**

| Key size | ECB | CBC | CTR | GCM | | | | CCM | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Init | Header | Payload | Tag | Init | Header | Payload | Tag |
| 128b | 51 [+59] (*) | | 51 | 64 | 35 | 51 | 59 | 63 | 55 | 114 | 58 |
| 256b | 75 [+82] (*) | | 75 | 88 | 35 | 75 | 75 | 87 | 79 | 162 | 82 |

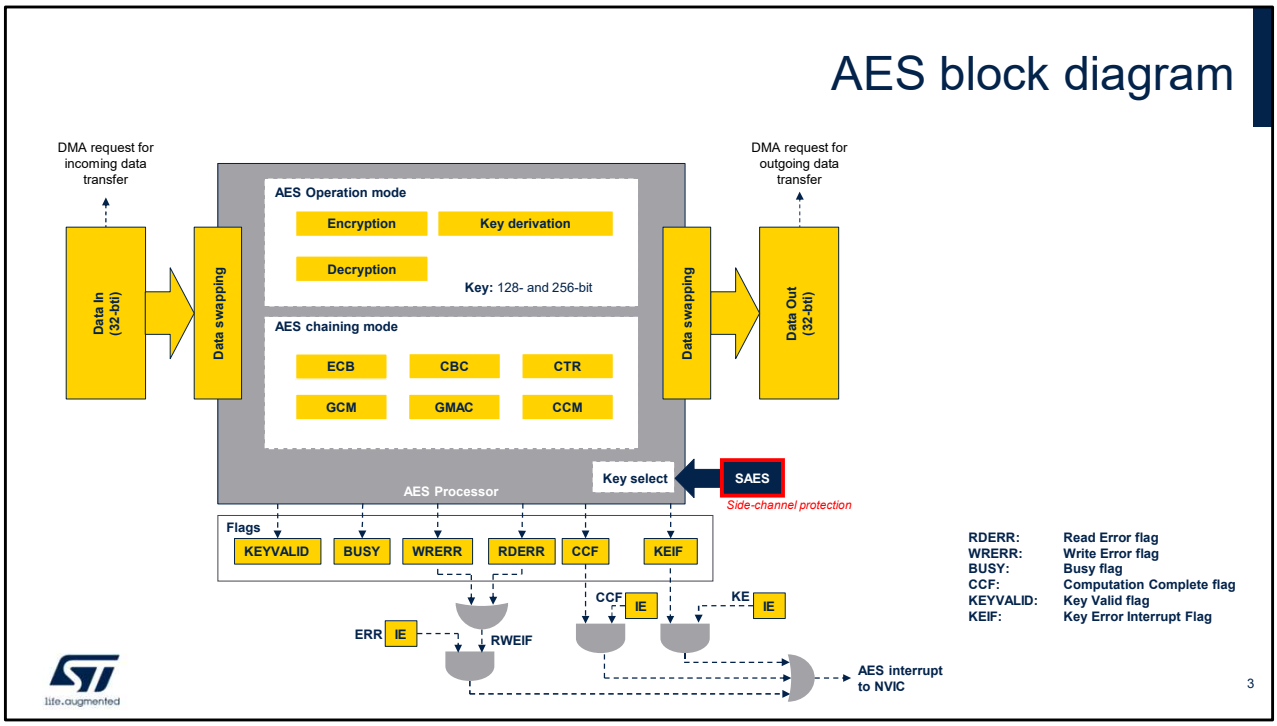
life.augmented

(*) For decryption you must add key derivation time, once

2

The AES module supports the Advanced Encryption System (AES) in three operating modes (encryption, decryption, key derivation).
It processes 128-bit data blocks using an encryption key that is either 128 or 256 bits long, based on the selected chaining mode.
Initiating the key-loading sequence sets the BUSY flag and clears the KEYVALID flag.
Once the amount of bytes defined by KEYSIZE is transferred to the AES_KEYRx registers, BUSY is cleared and KEYVALID set and the EN bit becomes writable.
The table indicates the number of clock cycles required to process a 128-bit block of data, according to the chaining mode and key size.

The AES module can load shared keys from the SAES module. This procedure is controlled by SAES.
A full data flow can be automated with the help of the Direct memory access controller (DMA).
Key derivation is supported for generating keys from hardware secret master key.




This block diagram highlights the features supported by the AES.

Multiple flags are supported by AES:

- The KEYVALID is set when a valid key is loaded in key registers.
- The KEIF is set when key information failed to load into key registers.
- The Read Error flag (RDERR) is set in the AES Status register when an unexpected read operation is detected during the computation phase or during the input phase.
- The Write Error flag (WRERR) is set in the AES Status register when an unexpected write operation is detected during the output phase or during the

computation phase.

Two extra flags are available for the AES accelerator to indicate the status of current operation:

- The Computation Complete flag (CCF) is set by hardware when the computation is complete.
- The Busy flag (BUSY), used only with GCM mode, indicates that a higher priority message can interrupt the current message during the GCM payload phase in encryption mode.

As shown some flags can generate interrupts toward the NVIC if the corresponding interrupt enable bit was previously set in the AES.

The AES module supports hardware key sharing with side-channel resistant SAES peripherals (Shared-key mode), controlled by SAES.

## SAES feature list

- AES chaining modes
  - Electronic codebook (ECB)
  - Cipher block chaining (CBC)
  - Counter (CTR) mode
  - Galois counter mode (GCM)
  - Galois message authentication code (GMAC)
  - Counter with CBC-MAC (CCM) mode
- Enhanced secure key storage
  - Hardware keys (DHUK, BHK)
    - Device-dependent, with DHUK
    - Application dependent, with BHK
  - Hardware secret key decryption (key unwrap)
  - Atomic key writing/loading enforcement

- Compliant AES operation modes on 128-bit data blocks, 128 or 256-bit keys
  - Encryption, Decryption (with associated key derivation mode)
- Key modes: normal, wrapped and shared key (loaded by faster CRYP engine)
- AHB slave with suspend/resume & DMA support (IN + OUT channels)
- Resistant to side channel attacks

| Number of cycles required to process a 128-bit block | | | | |
|---|---|---|---|---|
| Key size | Encryption | | Decryption | |
| | ECB | CBC | ECB | CBC |
| 128b | 480 | | 480 [+145] (*) | |
| 256b | 680 | | 680 [+230] (*) | |

(*) For decryption you must add key derivation time, once

life.augmented

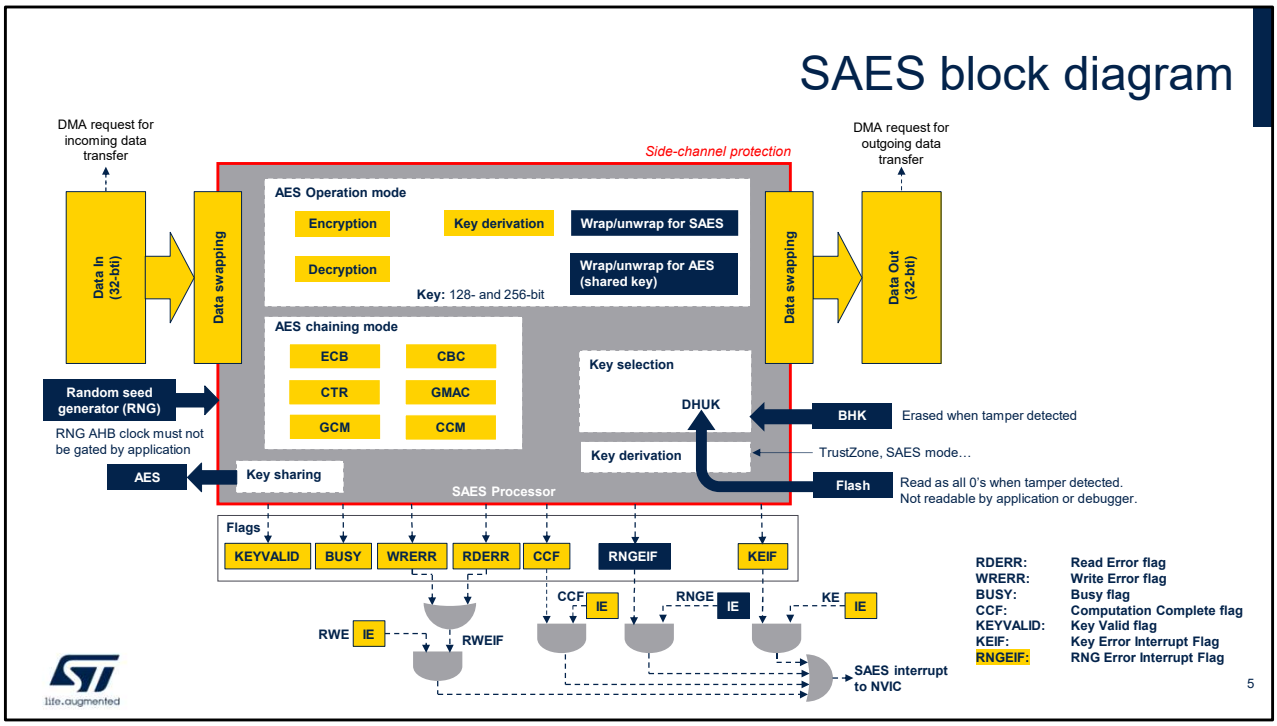
4

The SAES module supports the Advanced Encryption System (AES) in several operating modes described in the slide.
It processes 128-bit data blocks using an encryption key that is either 128 or 256 bits long, based on the selected chaining mode.
The table indicates the number of clock cycles required to process a block of data, according to the chaining mode and key size.

The SAES module offers the possibility to load secret keys by hardware, usable but not readable by the application.
Secret keys are boot hardware key BHK and derived hardware unique key DHUK.

It can also encrypt and decrypt application keys using these hardware-secret keys DHUK, XOR-ed or not with the application key BHK. With this feature, AES keys can be made usable by application software without being exposed in clear-text.

A full data flow can be automated with the help of the Direct memory access controller (DMA).
The SAES module incorporates a protection against side-channel attacks (SCA), including differential power analysis (DPA). It can share decrypted keys to the faster AES module. This procedure is controlled by SAES.




This SAES block diagram highlights the new features supported by the SAES, compared to AES.

- SAES fetches random numbers from the RNG peripheral automatically after a module reset triggered in the RCC.
- SAES has the possibility to load the secret keys DHUK and BHK by hardware.
- SAES can wrap (encrypt) and unwrap (decrypt) application keys using these hardware-secret keys DHUK, XOR-ed or not with BHK. An unwrapped key can be usable only in SAES, or it can be shared with the AES module (shared key mode)

BHK key can be cleared / erased when a tamper is detected, making all secrets undecipherable by the

attacker.
SAES cannot be used when RNGEIF is set. This flag is set when an error is detected while fetching a random number from RNG peripheral, due to, for example, bad entropy.

# Thank you



life.augmented

For more details on the new enhanced secure key storage feature and wrap/share key modes please refer to Enhanced secure key storage training module