AES-XTS

AES-XTS Storage Encrypt/Decrypt Engine

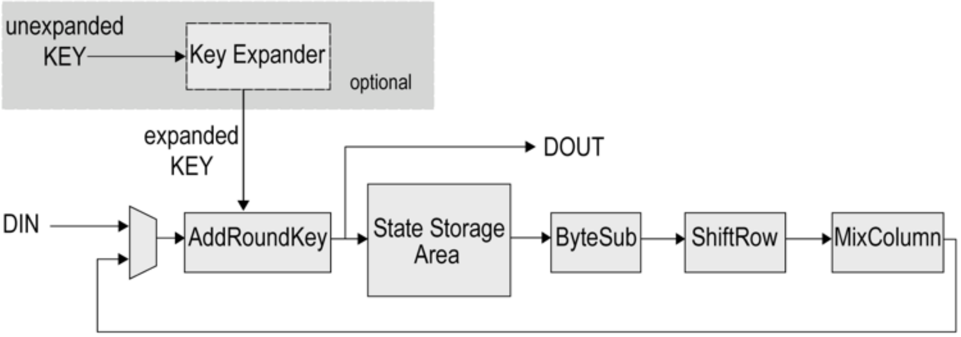
intel FPGA

The AES-XTS encryption IP core implements hardware encryption/decryption for sector-based storage data. It uses the AES block cipher, in compliance with the NIST Advanced Encryption Standard, as a subroutine. The core processes 128 bits per cycle, and is programmable for 128- and 256-bit key lengths.

Two architectural versions are available to suit system size and throughput requirements. The **High Throughput** XTS-X is more compact and can process 128 bits/cycle independent of the key size. The **Higher Throughput** XTS-X2 can process 256 bits/cycle independent of the key size. Both versions have a 128-bit data path.

XTS (XEX-based Tweaked Codebook Mode with Ciphertext Stealing) is a mode of AES that has been specifically designed to encrypt fixed size data where a possible threat has access to the stored data.

Block Diagram



FEATURES

- Encrypts and decrypts using the AES Rijndael Block Cipher Algorithm
- Implemented according to the IEEE P1619™/D16 standard
- NIST-Validated
- Capable of processing 128 bits/cycle
- Employs user-programmable key size of 128 or 256 bits
- Two architectural versions:
  - The AES-XTS-X version is smaller and can process 128 bits/cycle for all key sizes
  - The AES-XTS-X2 version can process 256 bits/cycle for all key sizes
- Arbitrary IV length
- Easy integration & implementation
  - Works with the integrated key expansion function
  - Fully synchronous, uses only the rising clock-edge, single-clock domain, no false or multicycle timing paths, scan-ready, LINT-clean, reusable design
  - Simple input and output interface, optionally bridged to AMBA™ interfaces or integrated with a DMA engine.
- Available in VHDL or Verilog source code format, or as a targeted FPGA

Applications

The AES-XTS can be utilized for a variety of encryption applications including full disk encryption (FDE), cloud storage encryption, network security and embedded systems to secure sensitive data such as in IoT devices or automotive systems.

Verification

The core has been verified through extensive synthesis, place and route and simulation runs. It has also been embedded in several products, and is proven in FPGA technologies.

Support

The core as delivered is warranted against defects for ninety days from purchase. Thirty days of phone and email technical support are included, starting with the first interaction. Additional maintenance and support options are available.

Export Permits

This core implements encryption functions and as such it is subject to export control regulations. Export to your country may or may not require a special export license. Please contact CAST to determine what applies in your specific case.

Key Expander

The AES algorithm requires an expanded key for encryption or decryption. The KEXP AES key expander core is included with the AES-XTS core.

During encryption, the key expander can produce the expanded key on the fly while the AES core is consuming it. For decryption, though, the key must be pre-expanded and stored in an appropriate memory before being used by the AES core. This is because the core uses the expanded key backwards during decryption.

Related Products

AES in CBC, CCM, CFB, CTR, ECB, GCM, LRW, and OFB modes are also available as stand-alone cores.

AES-P: run-time programmable AES core supporting CBC, CFB, CTR, ECB and OFB modes.

Implementation Results

The AES-XTS can be mapped to any Intel® FPGA device (provided sufficient silicon resources are available). The following are sample Intel results with all core I/Os assumed to be routed on-chip.

AES-XTS High Throughput (-X) Intel Results

Technology	Logic Resources	Memory Resources	Freq. (MHz)	Throughput (Gbps)
Arria 10 GX (-1)	4,674 ALMs	228 RAMB	170	21.76
Stratix V (-1)	4,831 ALMs	228 RAMB	210	26.88

AES-XTS High Throughput (-X2) Intel Results

Technology	Logic Resources	Memory Resources	Freq. (MHz)	Throughput (Gbps)
Arria 10 GX (-1)	8,526 ALMs	436 RAMB	150	38.40
Stratix V (-1)	8,921 ALMs	436 RAMB	180	46.08

Deliverables

The core is available in RTL (VHDL or Verilog) source code or as a targeted FPGA netlist. Its deliverable package includes the following:

- Sophisticated self-checking HDL testbench
- C Model & test vector generator
- Sample simulation & synthesis scripts
- User documentation