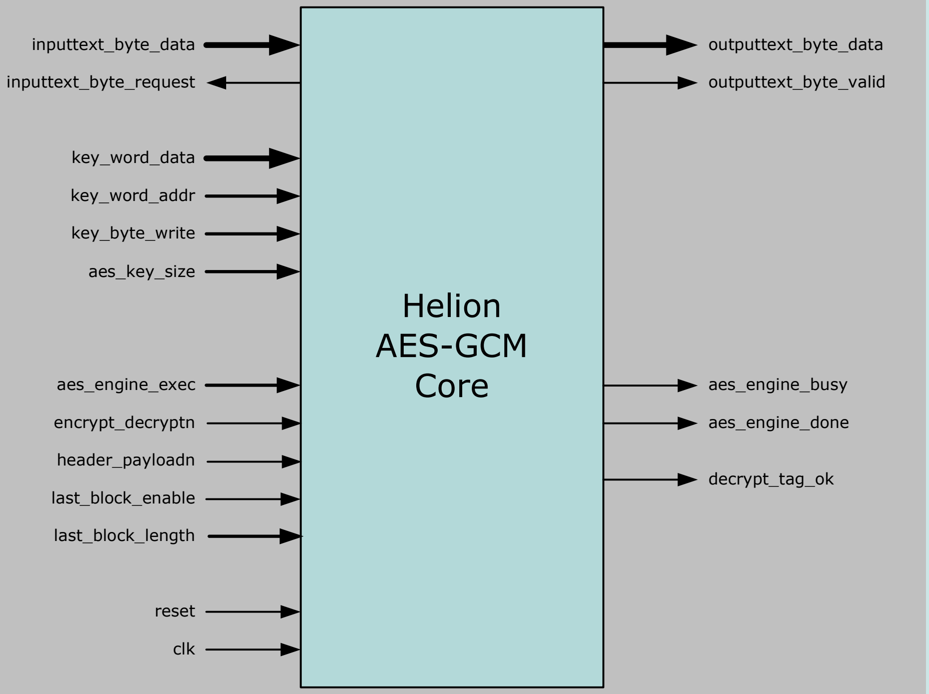
Helion Technology

PRODUCT BRIEF | **AES-GCM IP cores for ASIC**



Features

- Implements Galois/Counter (GCM) authenticated encryption mode to NIST SP800-38D
- Supports all AES key sizes (128, 192, and 256 bits) with integrated key expansion
- 96-bit Nonce/IV support
- Performs AES and GHASH functions needed for GCM including final block padding, tag appending and checking
- Simple 8-bit data interface for easy system integration
- Suitable for use in IPsec, MACsec, IEEE1619.1 and other applications
- Available in multiple versions providing optimal area/performance AES-GCM solution in ASIC

Deliverables

- Fully synthesisable RTL source code
- VHDL/Verilog testbench with test vectors
- User documentation

Overview

AES-GCM is an authenticated encryption block cipher mode which provides data confidentiality, integrity and origin authentication based on a single secret key, and is described formally in NIST Special Publication SP800-38D. The implementation of GCM described in this brief targets medium throughput applications, with emphasis on low resource usage and ease of use via a byte-wide interface.

The Helion AES-GCM core integrates all of the underlying functions required to implement AES in GCM mode including round-key expansion, counter mode logic, hash length counters, final block padding, and tag appending and checking features. The only external logic required is to form the Nonce block from various application specific packet header fields. Support is provided for both optional header and zero-length payload, and configurable tag length, making the core suitable for IPsec (RFC4106), MACsec (IEEE802.1ae) and Tape Storage (IEEE1619.1) applications.

Helion Technology Limited
Ash House, Breckenwood Road,
Fulbourn, Cambridge CB21 5DQ, England

Functional Description

Background

The Helion AES-GCM cores use AES-CTR operations to provide data encryption or decryption, and GHASH operations to provide message authentication. Data to be processed will be a series of atomic messages, each of which has its own Nonce/IV and will have its own authentication tag appended. Within each message, the data may be classed as "header" (which is not encrypted, but is authenticated) or "payload" (which is both encrypted and authenticated). This maps well into typical network packet formats, where the header must be unchanged through the encryption process, but needs to be protected by the authentication. Messages with no header or no payload are also supported.

Getting started

The user drives the core by issuing a sequence of commands via the *aes_engine_exec* port, and the core's status is indicated at each stage by the *aes_engine_busy* and *aes_engine_done* flags. Before any data processing can occur, a master AES key must be loaded into the core using the byte-writable 32-bit key interface. Key pre-processing to derive the internal GHASH key must then be initiated by the user issuing an EXEC_KEY command to the core, and indicating the AES key size to be used (via *aes_key_size*).

For each message

Before the start of each message to be encrypted or decrypted, the Nonce/IV must also be loaded by issuing an EXEC_INIT command to the core. The 128-bit Nonce/IV (96 bits used) is then transferred into the core using the byte-wide data input interface (*inputtext_byte_data*).

Processing the message data

Once the core has been initialised with a valid Key and Nonce/IV, the Message data can be processed. This is performed using multiple 128-bit block encrypt/decrypt operations, initiated by issuing one or more EXEC_DATA commands to the core. Control inputs are used to indicate the direction (*encrypt_decryptn*) and data type (*header_payloadn*) for each operation.

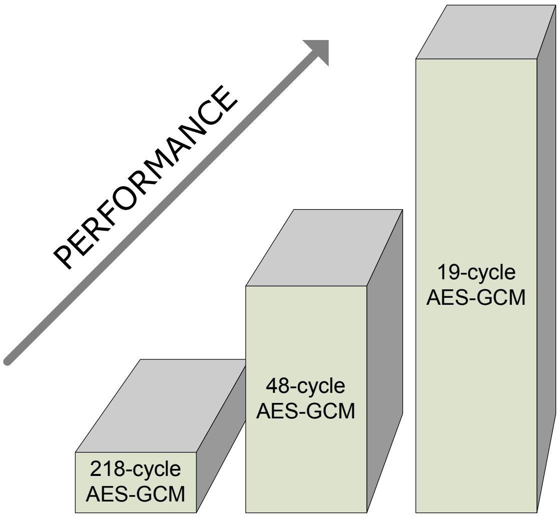
Input Data blocks enter the core via the byte-wide data input interface (*inputtext_byte_data*), and the resulting output data blocks emerge from the core via the byte-wide data output interface (*outputtext_byte_data*). Simple handshaking controls indicate when input data bytes are required or when output data bytes are valid.

At the end of the message

The last header or payload block may be less than the full 128 bits, and so its presence and length in bytes are indicated to the core using the "last_block" control inputs. Once the last message block has been encrypted/ decrypted, the tag will either be appended to the output data (encrypt direction), or will be checked against the received tag (decrypt direction) and the tag check output flag (*decrypt_tag_ok*) will be driven accordingly. The process can then be repeated for the next message, by loading a new key (if required) or the next Nonce/IV.

Core Choice

Helion always offer a range of solutions so that the throughput requirements of any application can be closely matched with optimum area efficiency. In this case, Helion have three levels of performance available; we name them to reflect the minimum number of clock cycles taken to process each 16-byte data block. NOTE. The actual number of cycles taken by the core to process this block varies with exact core choice and the keysize selected (see table on next page).



The smallest member of the family is the **"218-cycle" AES-GCM** core which takes a minimum 218 clock cycles to encrypt or decrypt each 16-byte data block using a 128-bit key.

For higher throughputs, the **"48-cycle" AES-GCM** core offers over four times the performance of the 218-cycle core while using less than twice its logic area. It takes a minimum 48 clock cycles to encrypt or decrypt each 16-byte data block using a 128-bit key.

The highest performance member of the family is the **"19-cycle" AES-GCM** core, which offers over twice the performance of the 48-cycle core while using approximately twice its logic area. It takes a minimum 19 clock cycles to encrypt or decrypt each 16-byte data block using any key size.

Each version of the core is available with support for one, two or all three AES key sizes (128, 192 and 256-bit).

Core Throughput

The tables below show the number of cycles and the maximum data throughput as a function of core clock frequency, for each version of the AES-GCM core, for each supported key size.

	AES-GCM 218-cycle			AES-GCM 48-cycle			AES-GCM 19-cycle		
key size	128	192	256	128	192	256	128	192	256
clock cycles used per 16-byte block	218	274	298	48	56	64	19	19	19
data throughput (Mbps per MHz)	0.58	0.46	0.42	2.6	2.2	2.0	6.7	6.7	6.7

For any specific application, a core version can be chosen that will achieve the required throughput, with an appropriate and achievable core clock frequency, not forgetting to take into account the inevitable per-message overheads which may be incurred (e.g Key setup, Nonce/IV setup and message finalisation).

Logic Utilisation and Performance

The table below shows typical gatecounts and maximum clock rates for some popular configurations of the Helion AES-GCM IP cores.

	AES-GCM Enc/Dec		
version	**218-cycle** (128/256-bit key)	**48-cycle** (128/192/256-bit key)	**19-cycle** (128/256-bit key)
typical gatecount	<13k gates	<19k gates	<39k gates
typical max clock rate (65nm)	400MHz	400MHz	400MHz

Note that exact figures will depend significantly on the target library used, as well as the synthesis method and options, so these numbers should be treated as preliminary guidance only.

Looking for Higher Rates?

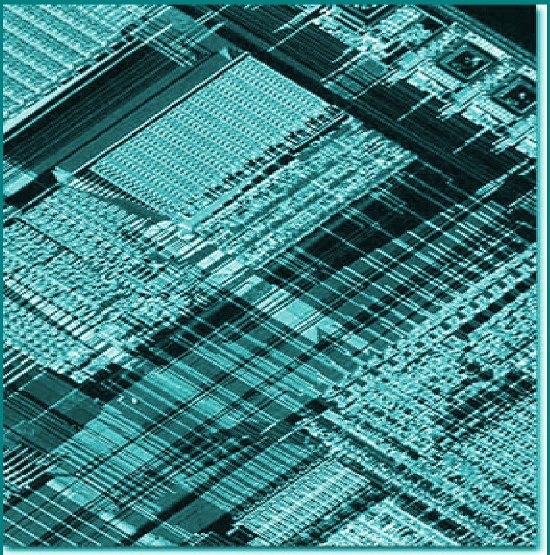
The AES-GCM mode was originally developed for use at very high data rates, where existing authentication mechanisms were running out of bandwidth. As well as the medium rate solutions presented here, Helion also have faster AES-GCM core families aimed at multi-Gigabit data rates. Please take a look at our “Giga” AES webpage at http://www.heliontech.com/aes_giga.htm, or contact Helion for more information on these faster AES-GCM solutions.

Ordering Information

Before ordering it is necessary to decide which of our family of AES-GCM cores will best fit your application. First decide between the **218-cycle**, **48-cycle**, and **19-cycle** cores according to the data throughput required and logic resources available. Then determine which AES key sizes you would like to support as well as any other special requirements your application may have.

Note: the 19-cycle version is available with a choice of “standard” or “fast” key expansion, which affects the overhead time of setting up a new key. The standard expansion is a particularly good choice when support for all three key sizes is required, as considerable area savings can be made.

If some of these choices are unclear, or you would just like to go over the options available, we are always happy to discuss the alternatives and help select the best solution for your application.



> **"Helion's products and services have consistently fit a variety of our needs.**
> **They are superbly easy to work with, answering questions in clear straightforward terms.**
> **The organization of their deliverables - particularly the concise and sensible documentation - provide for fast successful integration of their IP.**
> **I deal with many vendors, Helion is currently my favorite by every conceivable metric."**

Jeff Harms
IP Procurement Lead
Microchip Technology

About Helion

Founded in 1992, Helion is a long established British company based in Cambridge, England. We offer a range of product-proven Data Security and Lossless Compression IP cores, backed by a team of highly experienced engineers, proudly developing and supporting a world-class portfolio.

Our aim is to offer our customers…

Innovation
Helion works hard to anticipate, understand and then deliver great solutions for its customers. As an example, Helion offered the world's first commercial AES core back in 2001, even before the industry had fully adopted the algorithm. This process continues unabated today, with new products in development that will lead the field.

High Performance
Helion IP is specially designed and optimised for each target technology. This means lots of work for us, but this approach yields amazing results for our customers. We always aim for the best in class performance and lowest utilisation in any given ASIC or FPGA target.

High Quality
IP should be problem free, so we always go the extra mile to ensure a smooth and trouble free integration phase for our products. We realise that our customers are putting their faith in us, and want to repay that with an outstandingly easy deployment.

Ease of Use
Helion engineers have many years of real product development experience, and so our IP is designed to be used in realistic situations. It is flexible and well thought through - the result being that it is simple to drop into your system.

See how we achieve all this by visiting our Clients page at http://www.heliontech.com/clients.htm

More Information

For more detailed information on this or any of our other products and services, please contact Helion and we will be pleased to discuss how we can assist with your individual requirements.

HELION

Helion Technology Limited
Ash House, Breckenwood Road,
Fulbourn, Cambridge CB21 5DQ, England

tel: +44 (0)1223 500 924 email: info@heliontech.com
fax: +44 (0)1223 500 923 web: www.heliontech.com