

## AES128 IP Core

September 10, 2021

Product Specification

Rev1.01



### Design Gateway Co.,Ltd

E-mail: ip-sales@design-gateway.com

URL: www.design-gateway.com

### Features

- Support AES ECB mode standard.
- Support 128-bit key size.
- Support input data width 128-bit.
- Throughput rate at 11.6 Mbits/MHz.
- Speed up to 2.9 Gbps @250MHz.
- 128-bit data calculation time is constant at 11 clock cycles.

### Core Facts

Provided with Core	
Documentation	User Guide, Design Guide
Design File Formats	Encrypted hdl File
Verification	A10SoC board @250MHz
Instantiation Templates	VHDL
Reference Designs & Application Notes	QuartusII Project, See Reference Design Manual
Additional Items	Arria10 SoC Development Board
Simulation Tool Used	
-	
Support	
Support Provided by Design Gateway Co., Ltd.	

**Table 1: Example Implementation Statistics for Encryption**

Family	Example Device	Fmax (MHz)	ALMs	Registers <sup>(1)</sup>	Design Tools
Arria10 SX	10AS066N3F40E2SGE2	250	1121.7	323	QuartusII 16.0

**Table 2: Example Implementation Statistics for Decryption**

Family	Example Device	Fmax (MHz)	ALMs	Registers <sup>(1)</sup>	Design Tools
Arria10 SX	10AS066N3F40E2SGE2	250	1295.0	473	QuartusII 16.0

Notes:

(1) Actual logic resource is dependent on percentage of unrelated logic.

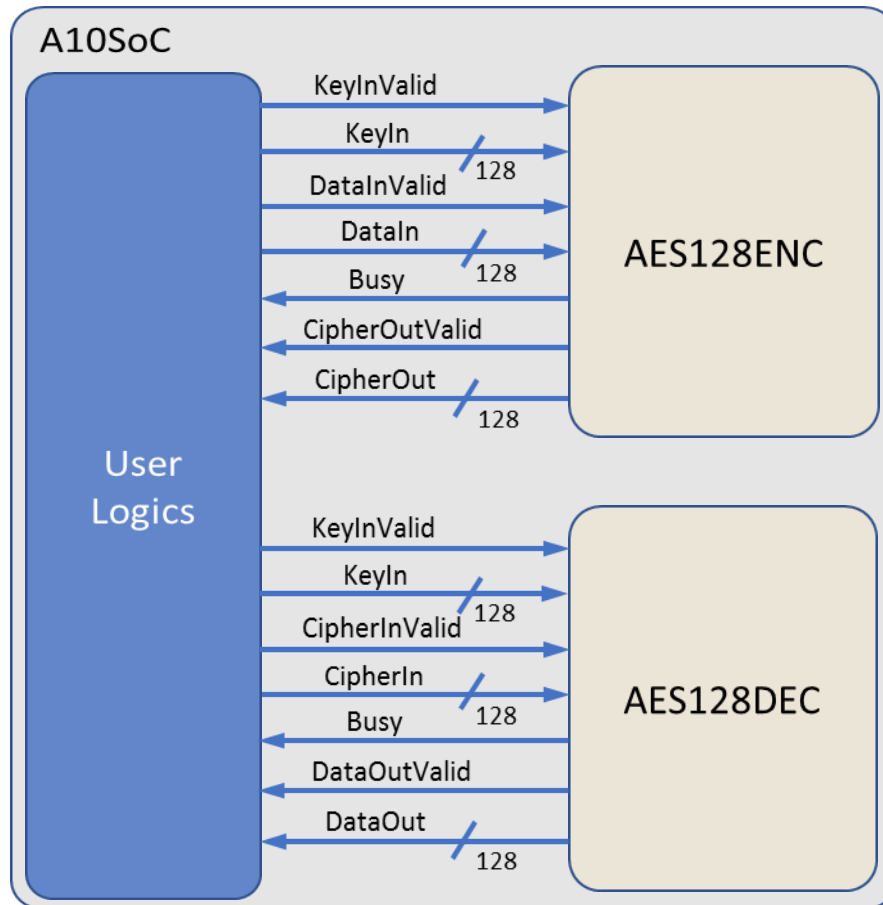


Figure 1: Block Diagram

## General Description

AES128 IP Core (AES128IP) implements the advanced encryption standard (AES) algorithm, which is widely used in many applications like file encryption, processor security, and secure file transfer protocol.

AES is a symmetric block cipher algorithm of the Rijndael family. Please find more details in the Federal Information Processing Standards Publication (FIPS PUB) 197.

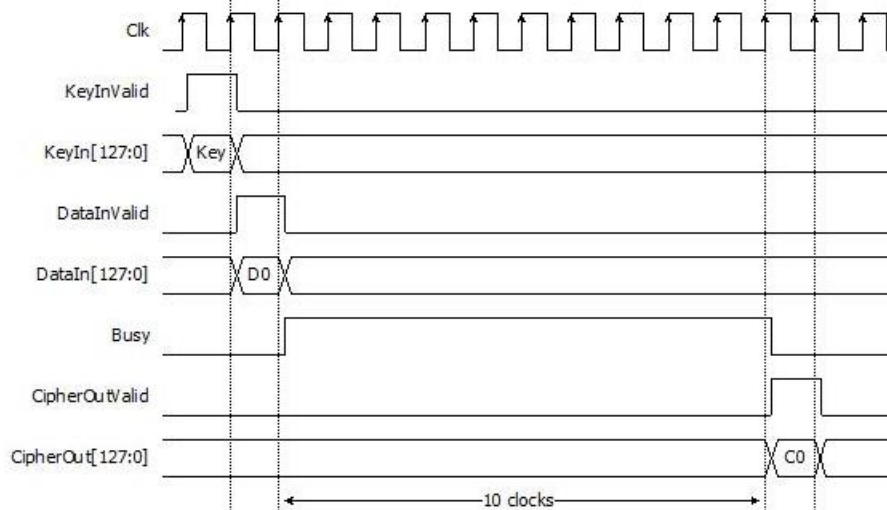
AES128IP consists of the AES128ENC module, which is the encryption module, and the AES128DEC module, which is the decryption module, as shown in Figure 1.

## Functional Description

### 1. AES128ENC

**Table 3: Interface signals of AES128ENC**

Signal name	Dir	Description
RstB	In	IP core system reset. Active low.
Clk	In	IP core system clock.
Key setting signals		
KeyInValid	In	KeyInValid is a user signal to specify data valid of KeyIn. Assert to '1' to set up KeyIn into AES128ENC.
KeyIn [127:0]	In	KeyIn is 128-bit key data of encryption. Valid when KeyInValid is asserted to '1'.
Encryption control signals		
DataInValid	In	DataInValid is a user signal to specify data valid of DataIn. Assert to '1' to indicate that DataIn is valid and start encryption process.
DataIn [127:0]	In	DataIn is 128-bit input data. Valid when DataInValid is asserted to '1'.
Busy	Out	AES128ENC Busy status. Assert to '1' when AES128ENC is busy for encryption process. AES128ENC will be ignored KeyInValid or DataInValid while Busy is '1'.
CipherOutValid	Out	CipherOutValid specify data valid for CipherOut. Assert to '1' when encryption finished.
CipherOut [127:0]	Out	CipherOut is 128-bit data output of encryption. Valid when CipherOutValid is asserted to '1'.



**Figure 2: AES128ENC operation timing diagram**

AES128ENC operation is as simple as 2 steps to use as below.

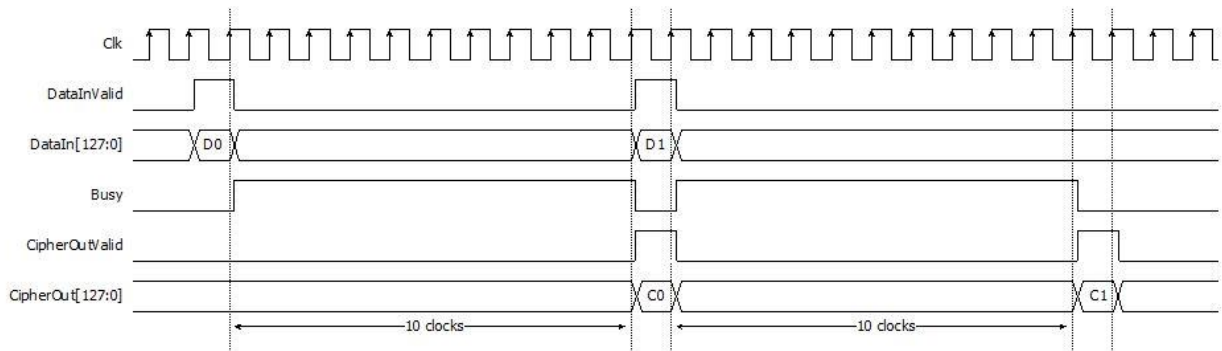
### 1.1. Key setting

As shown in Figure 2. Key setting is the first step to used AES128ENC. Which AES128ENC is set up KeyIn[127:0] when KeyInvalid='1'. After that, this key is used for every encryption. And key can be changed via KeyInvalid is asserted to '1' to set up KeyIn.

### 1.2. Encryption control

As shown in Figure 2. After key setting, AES128ENC starts the process when DataInvalid= '1'. Which Busy is set to be '1' in next cycle. And wait for 10 clock cycles, Process will finish when busy is set to be '0'. And encrypted data is set to CipherOut[127:0] signal while CipherOutValid signal is set to be '1' in this cycle.

For the best performance of encryption process, Figure 3 shows timing diagram of continuous and pipelining encryption.

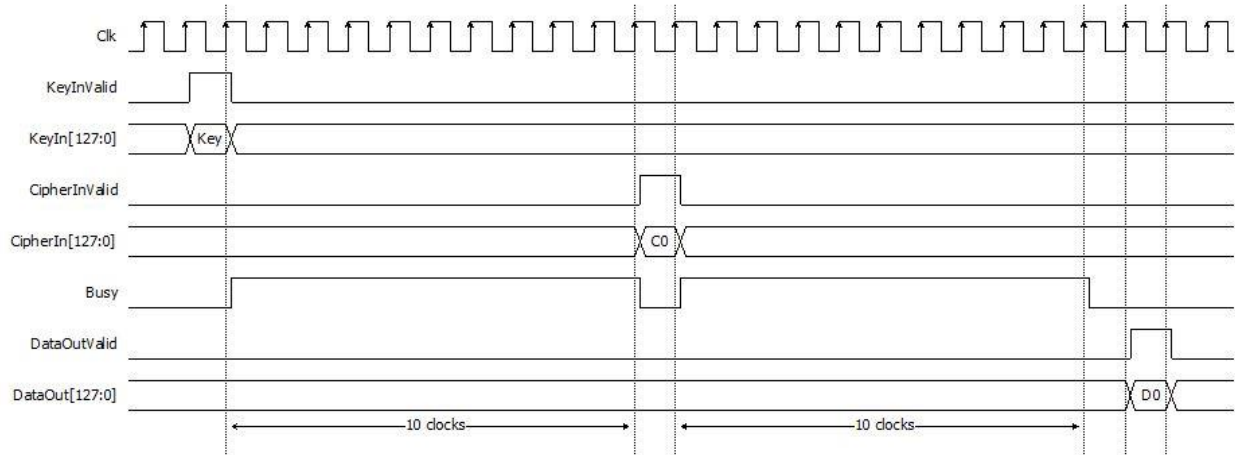


**Figure 3: Continuous and pipelining encryption**

## 2. AES128DEC

**Table 4: Interface signals of AES128DEC**

Signal name	Dir	Description
RstB	In	IP core system reset. Active low.
Clk	In	IP core system clock.
<b>Key initialize signals</b>		
KeyInValid	In	KeyInValid is a user signal to specify data valid of KeyIn. Assert to '1' to indicate that KeyIn is valid and start key initialize process.
KeyIn [127:0]	In	KeyIn is 128-bit key data of decryption. Valid when KeyInValid is asserted to '1'.
<b>Decryption control signals</b>		
CipherInValid	In	CipherInValid is a user signal to specify data valid of CipherIn. Assert to '1' to indicate that CipherIn is valid and start decryption process.
CipherIn [127:0]	In	CipherIn is 128-bit encrypted input data. Valid when CipherInValid is asserted to '1'.
Busy	Out	AES128DEC Busy status. Assert to '1' when AES128DEC is busy for key initialization and encryption process. AES128DEC will be ignored KeyInValid and CypherInValid while Busy is '1'.
DataOutValid	Out	DataOutValid specify data valid for DataOut. Assert to '1' when decryption finished.
DataOut [127:0]	Out	DataOut is 128-bit data output of decryption. Valid when DataOutValid is asserted to '1'.



**Figure 4: AES128DEC operation timing diagram**

AES128DEC operation is as simple as 2 steps to use as below.

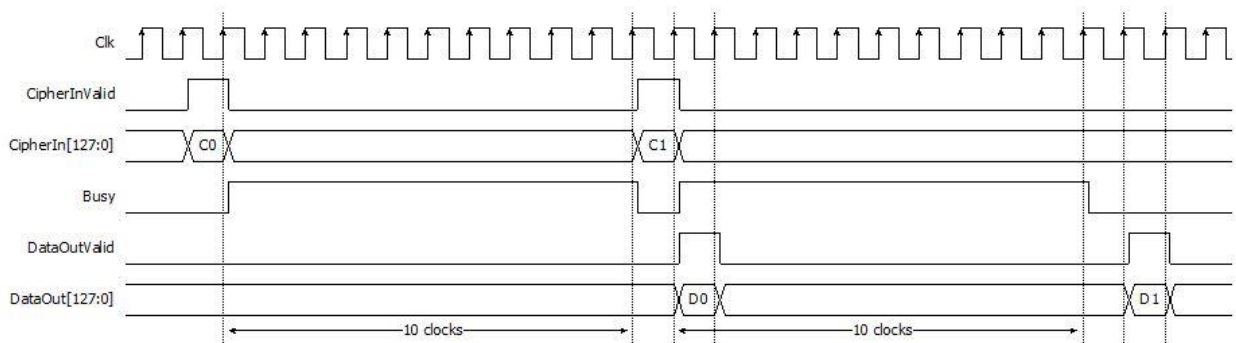
### 2.1. Key initialize

As shown in Figure 4. Key initialize is the first step to used AES128DEC. Which AES128DEC is started key initialize process when KeyInvalid='1'. KeyIn[127:0] will be set up in AES128DEC after the initialize process is finished (Busy='0'). After that, this key is used for every decryption. And key can be changed via KeyInvalid is asserted to '1'.

### 2.2. Decryption control

As shown in Figure 4. After key initializes, AES128DEC starts the process when CipherInvalid= '1'. Which Busy is set to be '1' in next cycle. And wait for 10 clock cycles, Process will finish when Busy is set to be '0'. After that, decrypted data is set to DataOut[127:0] signal while DataOutValid signal is set to be '1' in next cycle.

For the best performance of decryption process, Figure 5 shows timing diagram of continuous and pipelining decryption.



**Figure 5: Continuous and pipelining decryption**

## Verification Methods

AES128 IP Core functionality were verified on real board design by using Arria10 SoC development board.

## Recommended Design Experience

The user must be familiar with HDL design methodology to integrate this IP into system.

## Ordering Information

This product is available directly from Design Gateway Co., Ltd. Please contact Design Gateway Co., Ltd. For pricing and additional information about this product using the contact information on the front page of this datasheet.

## Revision History

Revision	Date	Description
1.01	10/Sep/2021	Revised description information.
1.00	30/Apr/2021	New release