

# **CoreAES128 v3.0**

*Handbook*

---

## **Actel Corporation, Mountain View, CA 94043**

© 2009 Actel Corporation. All rights reserved.

Printed in the United States of America

Part Number: 50200122-1

Release: June 2009

No part of this document may be copied or reproduced in any form or by any means without prior written consent of Actel.

Actel makes no warranties with respect to this documentation and disclaims any implied warranties of merchantability or fitness for a particular purpose. Information in this document is subject to change without notice. Actel assumes no responsibility for any errors that may appear in this document.

This document contains confidential proprietary information that is not to be disclosed to any unauthorized person without prior written consent of Actel Corporation.

### **Trademarks**

Actel and the Actel logo are registered trademarks of Actel Corporation.

Adobe and Acrobat Reader are registered trademarks of Adobe Systems, Inc.

All other products or brand names mentioned are trademarks or registered trademarks of their respective holders.

---

# Table of Contents

	Introduction . . . . .	5
	Core Overview . . . . .	5
	Design Security . . . . .	6
	Key Features . . . . .	6
	Core Version . . . . .	6
	Supported Tool Flows . . . . .	6
	Supported Families . . . . .	7
	Device Utilization and Performance . . . . .	7
<b>1</b>	<b>Design Description . . . . .</b>	<b>9</b>
	I/O Signals . . . . .	9
	Parameters/Generics . . . . .	10
<b>2</b>	<b>Functional Block Description . . . . .</b>	<b>11</b>
	Data Schedule Logic . . . . .	11
	State Correlator Logic . . . . .	11
	Key Schedule Logic . . . . .	11
	Key Expansion Logic . . . . .	11
<b>3</b>	<b>CoreAES128 Protocol Overview . . . . .</b>	<b>13</b>
	CoreAES128 Initialization . . . . .	13
	CoreAES128 Operation . . . . .	13
	Cipher Key Expansion . . . . .	13
	Encryption . . . . .	14
	Decryption . . . . .	15
	Pause/Resume . . . . .	16
	Clear/Abort . . . . .	17
	Modes of Operation . . . . .	18
<b>4</b>	<b>Tool Flows . . . . .</b>	<b>19</b>
	License . . . . .	19
	SmartDesign . . . . .	19
	Simulation Flows . . . . .	19
	Synthesis in Libero IDE . . . . .	20
	Place-and-Route in Libero IDE . . . . .	20
<b>5</b>	<b>Testbench Operation . . . . .</b>	<b>21</b>
	User Testbench . . . . .	21
<b>6</b>	<b>Ordering Information . . . . .</b>	<b>23</b>
	Ordering Codes . . . . .	23

7 Export Restrictions . . . . . 25

A Product Support . . . . . 27

    Customer Service . . . . . 27

    Actel Customer Technical Support Center . . . . . 27

    Actel Technical Support . . . . . 27

    Website . . . . . 27

    Contacting the Customer Technical Support Center . . . . . 27

B Index . . . . . 29

# Introduction

## Core Overview

The CoreAES128 macro implements the advanced encryption standard (AES), which provides a means of securing data. AES utilizes the Rijndael algorithm, which is described in detail in the Federal Information Processing Standards Publication (FIPS PUB) 197. The AES Rijndael algorithm (Figure 1) takes as inputs 128 bits of plaintext data and 128 bits of a cipher key. After several rounds of computation, the algorithm produces a 128-bit ciphered version of the original plaintext data as output<sup>1</sup>. During the rounds of the algorithm, data bits are subjected to byte substitution, data shift operations, data mixing operations, and addition (XOR) operations with an expanded version of the original 128-bit cipher key.

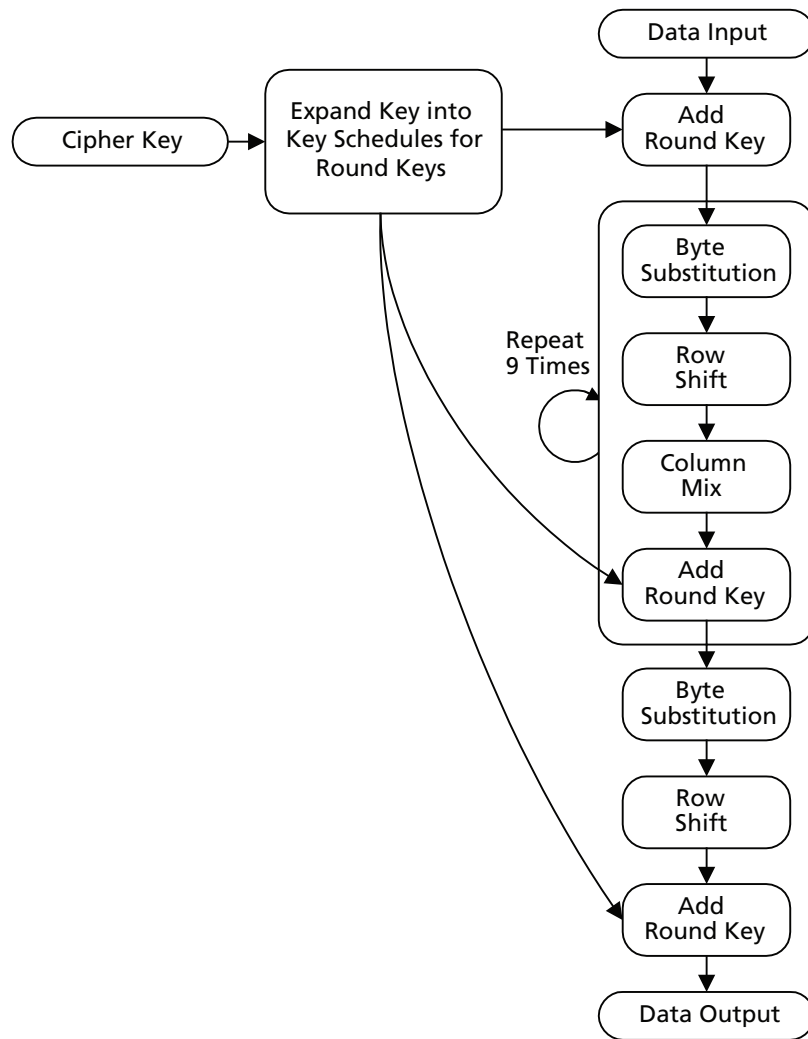


Figure 1 · AES Algorithm (128-bit cipher key)

1. FIPS PUB 197 allows for key sizes of 128, 192, and 256 bits. However, this implementation supports a cipher key size of 128 bits only.

## Design Security

Figure 2 shows a typical system diagram. The cipher key, which is the “secret” key, can be made up of FPGA logic cells, preventing the possibility of design or data theft. Actel flash-based ProASIC<sup>PLUS</sup>® devices employ FlashLock® technology, and Actel antifuse-based Axcelerator® devices employ FuseLock™ technology; each of which keeps the cipher key and the rest of the logic secure. The output of the CoreAES128 macro should be connected to registers or FIFOs, as it is only valid for one clock cycle. Refer to the example used in the “Encryption” section on page 14 and “Decryption” section on page 15.

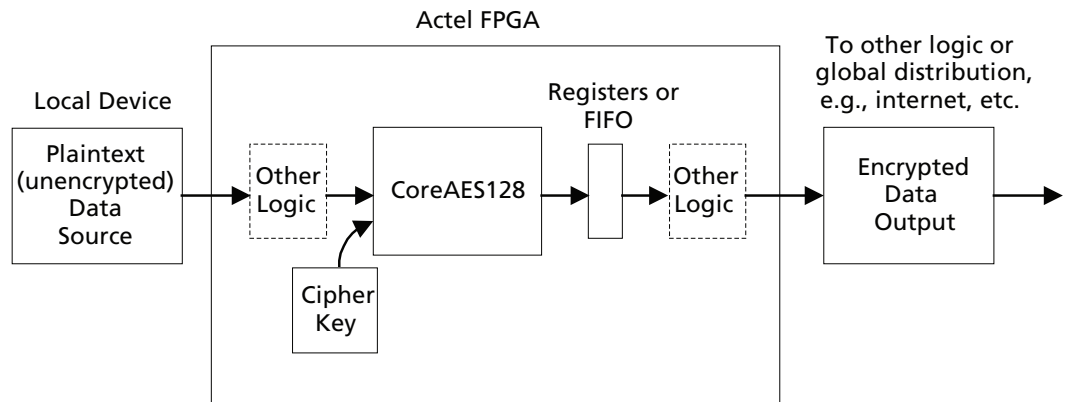


Figure 2 · Typical CoreAES128 System

## Key Features

- Compliant with FIPS PUB 197
- ECB implementation per NIST SP 800-38A
- Example source code provided for cipher block chaining (CBC), cipher feedback (CFB), output feedback (OFB), and counter (CTR) modes
- 128-bit cipher key
- Encryption and decryption possible with the same core
- 44-Clock cycle operation to encrypt or decrypt 128 bits of data
- Pause/resume functionality to continue encryption or decryption at will
- Provides redundant security

## Core Version

This handbook supports CoreAES128 version 3.0.

## Supported Tool Flows

CoreAES128 requires SmartDesign and Actel Libero® Integrated Design Environment (IDE) v8.4.

## Supported Families

- IGLOO®
- IGLOOe
- ProASIC®3
- ProASIC3E
- ProASIC3L
- Fusion
- ProASIC<sup>PLUS</sup>
- Axcelerator
- RTAX-S

## Device Utilization and Performance

Table 1 provides a summary of the implementation data for CoreAES128.

Table 1 · CoreAES128 Device Utilization and Performance

Family	Cells or Tiles			RAM Blocks	Utilization		Performance	Throughput
	Sequential	Combinatorial	Total		Device	Percentage		
IGLOO	366	3,683	4,049	8	AGL1000V5	17%	58 MHz	169 Mbps
IGLOOe	366	3,683	4,049	8	AGLE3000V5	6%	57 MHz	166 Mbps
ProASIC3	366	3,683	4,049	8	A3P1000	33%	88 MHz	256 Mbps
ProASIC3E	366	3,683	4,049	8	A3PE3000	33%	84 MHz	245 Mbps
ProASIC3L	366	3,683	4,049	8	A3PE3000L	6%	70 MHz	204 Mbps
Fusion	366	3,683	4,049	8	AFS600	38%	75 MHz	218 Mbps
ProASIC <sup>PLUS</sup>	427	4,507	4,934	24	APA1000	9%	36 MHz	104 Mbps
Axcelerator	380	2,222	2,602	10	AX1000	15%	90 MHz	262 Mbps
RTAX-S	380	2,222	2,602	10	RTAX1000S	15%	58 MHz	169 Mbps

*Note:* Data in this table were achieved using typical synthesis and layout settings.

Data throughput is computed by taking the bit width of the data (128 bits), dividing it by the number of cycles (44), and multiplying it by the clock rate (performance); the result is listed in Mbps<sup>1</sup>.

<sup>1</sup> Throughput does not include the time required to expand the cipher key.



# Design Description

## I/O Signals

The port signals for the CoreAES128 macro are illustrated in Figure 1-1 and described in Table 1-1. All signals are either “Input” (input only) or “Output” (output only).

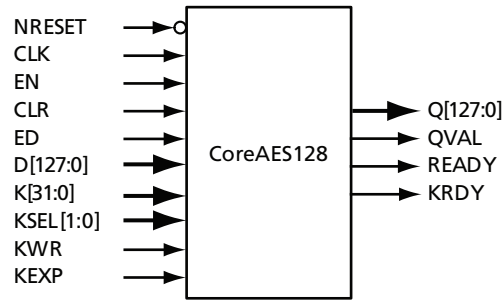


Figure 1-1 · CoreAES128 I/O Signal Diagram

Table 1-1 · CoreAES128 I/O Signals

Name	Type	Description
NRESET	Input	Active-low asynchronous reset
CLK	Input	System clock. Reference clock for all internal logic
EN	Input	Enable signal. Set to '1' for normal continuous encrypt/decrypt operation, set to '0' to pause.
CLR	Input	Synchronous clear signal. Set to '1' to clear logic at any time.
ED	Input	Encryption/decryption. '1' to encrypt, '0' to decrypt
D[127:0]	Input	Data in. 128-bit data input bus
K[31:0]	Input	Key. 32-bit cipher key input bus
KSEL[1:0]	Input	Key select. Selection bits to direct K[31:0] to one of the four 32-bit words comprising the internal 128-bit cipher key.
KWR	Input	Key write. Set to '1' to write K[31:0] to one of the four 32-bit words comprising the internal 128-bit cipher key.
KEXP	Input	Key expand. Set to '1' to expand the 128-bit internal key.
Q[127:0]	Output	Data out. 128-bit cipher text (encryption operation)/plaintext (decryption operation) output bus
QVAL	Output	Q Valid. '1' indicates that valid encryption/decryption data is available on Q[127:0].
READY	Output	Ready. '1' indicates that CoreAES128 has finished its initialization sequence 1,024 clock cycles after the rising edge of NRESET.
KRDY	Output	Key ready. '1' indicates that the internal 128-bit cipher key was expanded and the macro is ready for encryption/decryption.

## Parameters/Generics

CoreAES128 has parameters (Verilog) and generics (VHDL) for configuring the RTL code (Table 1-2). All parameters and generics are integer types and are mapped to configuration options in the CoreConsole configuration window.

Table 1-2 · CoreAES128 Configuration Parameters

Parameter	Values	Description
FAMILY	0 to 99	Must be set to match the supported FPGA family: 11 – Axcelerator 12 – RTAXS 14 – ProASIC <sup>PLUS</sup> 15 – ProASIC3 16 – ProASIC3E 17 – Fusion 20 – IGLOO 21 – IGLOOe 22 – ProASIC3L
CFG_MODE	1 to 3	Control encryption/decryption functionality: 1 – Enables encryption functionality only 2 – Enables decryption functionality only 3 – Encryption/decryption according to ED bit input

## Functional Block Description

CoreAES128 encrypts the input data by rounds of computations that produce the ciphered version of input data. Figure 2-1 shows the block diagram for CoreAES128.

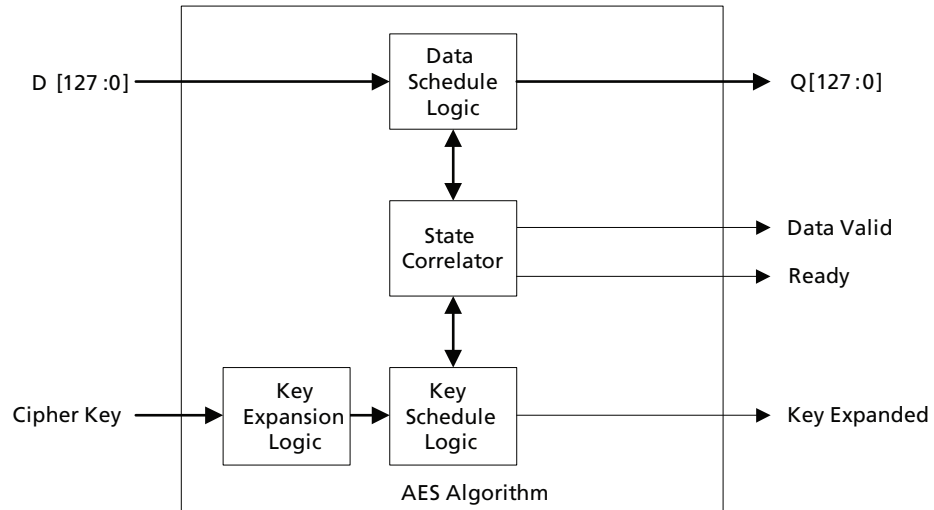


Figure 2-1 · CoreAES128 Algorithm Block Diagram

### Data Schedule Logic

The data schedule logic block of CoreAES128 computes the intermediate data values at each round of the AES algorithm.

### State Correlator Logic

The state correlator logic block for CoreAES128 maintains coherency between data and key schedule logic.

### Key Schedule Logic

The key schedule logic block of CoreAES128 controls the intermediate key schedule at each round of the AES algorithm.

### Key Expansion Logic

The key expansion logic block of CoreAES128 expands the original 128-bit key for use in encryption or decryption operations.



# CoreAES128 Protocol Overview

## CoreAES128 Initialization

After a reset condition, shown in [Figure 3-1](#), the CoreAES128 macro performs a self-initialization process. This initialization process takes 1,024 clock cycles, then the READY signal becomes active at logic '1'. Once READY is active, the CoreAES128 macro is ready for cipher key expansion, followed by encryption or decryption operations.

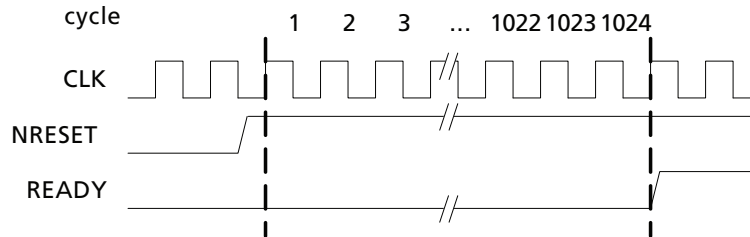


Figure 3-1 · CoreAES128 Initialization

## CoreAES128 Operation

As shown on the left side of [Figure 1 on page 5](#), the AES algorithm requires an expanded version of the original cipher key for use in encrypting or decrypting data. Upon a power-up condition, the cipher key and the expanded version of the cipher key are undefined. Therefore, they must be set up after the initialization process, described in the “[CoreAES128 Initialization](#)” section, and before encryption or decryption operations can take place. The procedures for writing and expanding the cipher key, described in the “[Cipher Key Expansion](#)” section, must be repeated any time a new 128-bit cipher key is required, such as after a reset or power-up condition.

**Note:** If the same cipher key is to be used for all encryption and decryption operations, the procedures for writing and expanding the cipher key only need to be performed once.

## Cipher Key Expansion

Prior to any encryption or decryption operation, the 128-bit cipher key must be written to CoreAES128 and expanded ([Figure 3-2 on page 14](#)). Refer to FIPS PUB 197 for the algorithmic details of the key expansion process.

Follow the steps below to write the four 32-bit words that make up the 128-bit cipher key, and to expand the 128-bit cipher key:

1. Set EN to logic '0'.
2. Set KSEL[1:0] to '00' to select the lowest 32 bits (LSB word) of the internal 128-bit cipher key.
3. Set K[31:0] to the value of the lowest 32-bit word of the desired 128-bit cipher key.
4. Set KWR to logic '1' for one clock cycle.
5. Set KSEL[1:0] to '01' to select the second lowest 32 bits of the internal 128-bit cipher key.
6. Set K[31:0] to the value of the second lowest 32-bit word of the desired 128-bit cipher key.
7. Set KWR to logic '1' for one clock cycle.
8. Set KSEL[1:0] to '10' to select the second highest 32 bits of the internal 128-bit cipher key.
9. Set K[31:0] to the value of the second highest 32-bit word of the desired 128-bit cipher key.
10. Set KWR to logic '1' for one clock cycle.
11. Set KSEL[1:0] to '11' to select the highest 32 bits (MSB word) of the internal 128-bit cipher key.
12. Set K[31:0] to the value of the highest 32-bit word of the desired 128-bit cipher key.

13. Set KWR to logic '1' for one clock cycle.
14. Set KWR back to logic '0'.
15. Set KEXP to logic '1' for one clock cycle.
16. Set KEXP back to logic '0'.
17. Wait for 52 clock cycles.

The four 32-bit words which comprise the 128-bit cipher key can be written in any order. It is not necessary to write them in sequential order; i.e., lowest 32-bit word to highest 32-bit word.

If the KRDY signal was active at logic '1' prior to setting the KWR signal to logic '1' (from a previously expanded cipher key), it becomes inactive on the next rising clock edge after performing “Set KWR to logic '1' for one clock cycle.” After 52 clock cycles, the KRDY signal becomes active; (logic '1') to indicate that the 128-bit cipher key was expanded internally. The CoreAES128 macro is now ready for encryption or decryption operations. The KRDY signal initializes to the inactive state of logic '0' after a reset condition (Figure 3-2), prior to the key expansion process.

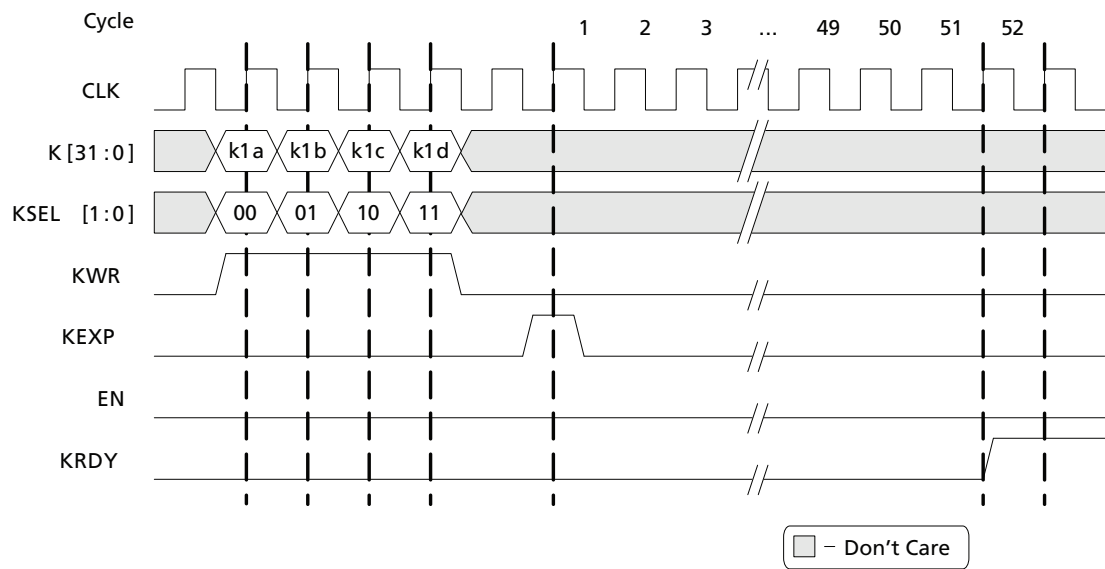


Figure 3-2 · Cipher Key Write and Expand

## Encryption

Follow the steps below to begin the encryption data process (Figure 3-3 on page 15):

1. Write and expand the 128-bit cipher key, if not already done (refer to “Cipher Key Expansion” on page 13.)
2. Set D[127:0] to the plaintext data to be encrypted (d1 in Figure 3-3).
3. Set ED to logic '1'.
4. Set EN to logic '1'.
5. Wait for 44 clock cycles.

After 44 clock cycles of the EN input being held continuously at logic '1', the QVAL signal will transition from logic '0' to logic '1' and remain valid for one clock cycle. This indicates that valid ciphered (encrypted) data (q1 in Figure 3-3) is available on the Q[127:0] outputs. The encrypted data is only available during clock cycle 44, so you must register or latch the data on Q[127:0], using the QVAL signal as a qualifying register enable or latch enable. As shown in

Figure 3-3, continuous encryption is possible. For example, the second 128-bit plaintext data word (d2 in Figure 3-3) can be immediately encrypted by setting the D[127:0] input to d2 on the rising clock edge of clock cycle 45.

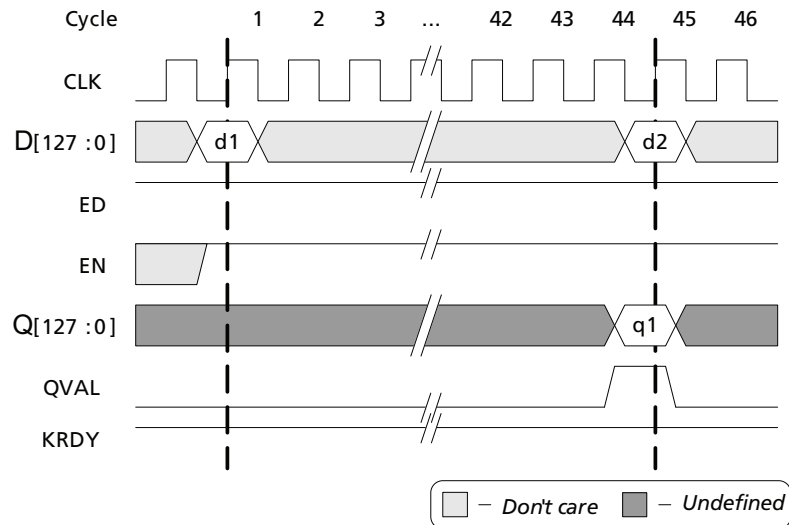


Figure 3-3 · Example Encryption Sequence

## Decryption

Follow the steps below to begin the decryption data process (Figure 3-4 on page 16):

1. Write and expand the 128-bit cipher key if not already done (refer to “Cipher Key Expansion” on page 13.)
2. Set D[127:0] to the ciphertext data to be decrypted (d1 in Figure 3-4).
3. Set ED to logic '0'.
4. Set EN to logic '1'.
5. Wait for 44 clock cycles.

After 44 clock cycles of the EN input being held continuously at logic '1', the QVAL signal will transition from logic '0' to logic '1' and remain valid for one clock cycle, indicating that valid plaintext (unencrypted data, shown as q1 in Figure 3-4) is available on the Q[127:0] outputs. The decrypted plaintext data is only available during clock cycle 44, so you must register or latch the data on Q[127:0] using the QVAL signal as a qualifying register enable or latch enable.

As shown in Figure 3-4, continuous decryption is possible. For example, the second 128-bit cipher text data word (d2 in Figure 3-4) can be immediately decrypted by setting the D[127:0] inputs to d2 on the rising clock edge of clock cycle 45.

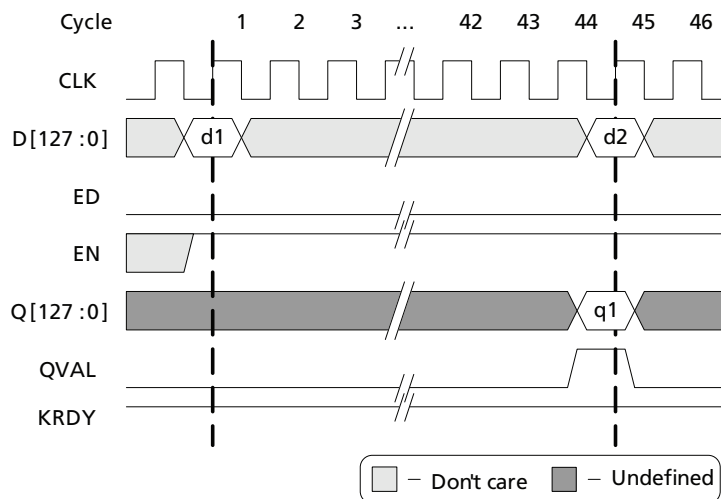


Figure 3-4 · Example Decryption Sequence

## Pause/Resume

For normal operation, the EN input is held at logic '1'. The core can be paused by holding the EN input at logic '0' indefinitely, as shown by the example in Figure 3-5 on page 17, where cycle 3 of an encryption operation is paused. To resume operation, bring the EN input back to logic '1'. This functionality applies to either encryption or decryption. Note that the ED input must remain at logic '1' throughout an entire encryption cycle or at logic '0' throughout an entire decryption cycle; otherwise, unpredictable results on the Q[127:0] outputs will occur. You can use the pause/resume functionality in the case where many blocks of data are encrypted one after another. For example, if the EN input is held statically at logic '1', the data inputs need to change every 44 clock cycles to encrypt the next block. After all blocks of data are encrypted, you would then need to hold the EN input at logic '0'. If it is left at logic '1', data will continue to be encrypted ad infinitum. When ready for the next blocks of data, resume the encryption process by holding the EN input at logic '1'. Another possibility occurs if there is an elastic buffer (FIFO) connected to the Q[127:0] output. If the FIFO is filling up with encrypted data faster than the encrypted data is being read out of the FIFO, you may want to pause the CoreAES128 macro by setting the EN input to a logic '0' when the full or almost-full flag logic from the FIFO is active.

When the FIFO full or almost-full flag logic clears, the CoreAES128 macro can then resume operation by again setting the EN input to logic '1'.

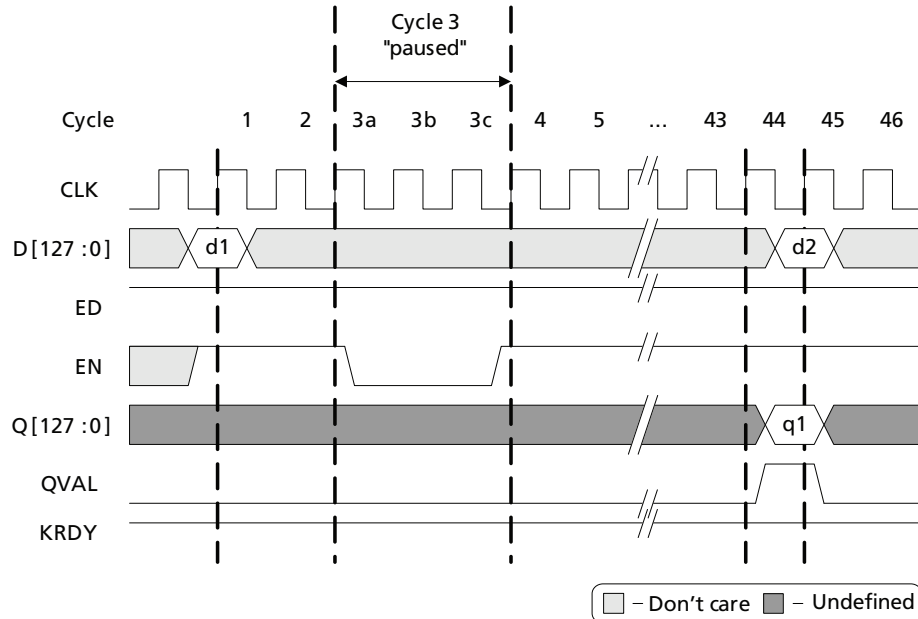


Figure 3-5 · Example Encryption Pause/Resume Sequence

## Clear/Abort

At any point in the process of encrypting or decrypting data, you can abort the current operation by setting the CLR input to logic '1'. This will clear all current calculations within the key schedule and data schedule logic. Then you can immediately begin to write and expand a different cipher key, as described in "Cipher Key Expansion" section, or use a different data input on the very next cycle, as shown in [Figure 3-6 on page 18](#), with d2 as the next 128-bit data block to be encrypted. Note that the CLR signal does not clear the 128-bit cipher key, the expanded version of the cipher key, or the KRDY signal. Only the signals NRESET, K[31:0], KWR, and KEXP affect the value of the 128-bit cipher key, the expanded version of the cipher key, and the KRDY output signal. The clear/abort functionality is provided as an additional tool. When you want to change the cipher key, possibly in the middle of an encryption or decryption sequence, you can stop the current operation immediately by holding the CLR input at logic '1' for at least one clock cycle new cipher key. After the new cipher key is expanded, new data can be encrypted. If the CoreAES128 macro is

integrated into a system containing a processor, the processor may abort the encryption or decryption operation for a specific event, such as low or failing power condition.

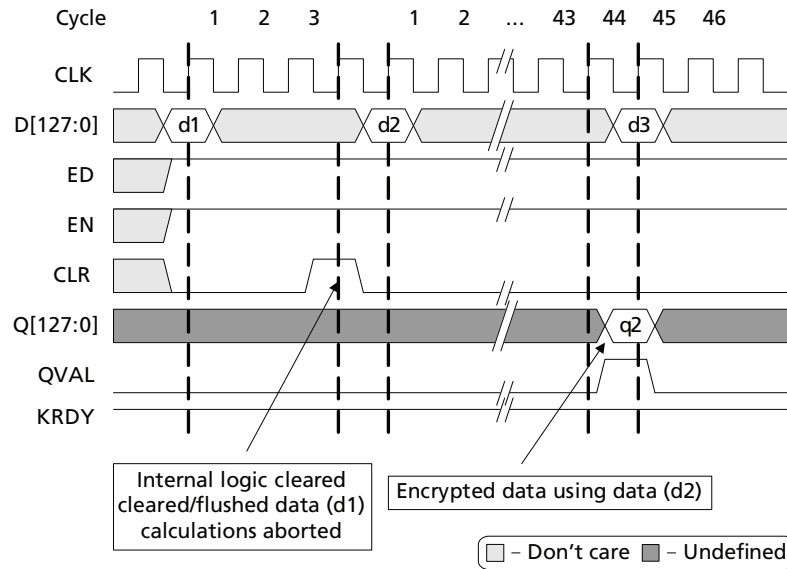


Figure 3-6 · Example Encryption Abort Sequence

## Modes of Operation

CoreAES128 is implemented using the ECB mode of operation, per NIST SP 800-38A. Depending on the application, other modes of operation for AES may be desirable. For this reason, Actel provides example VHDL and Verilog source code for the CBC, CFB, OFB, and CTR modes. Refer to the *modes* directory for the wrapper design for each mode. For detailed information on specific modes of operation, refer to NIST SP 800-38A.

## Tool Flows

### License

CoreAES128 is licensed in two ways. Depending on your license tool flow, functionality may be limited.

### Obfuscated

Complete RTL code is provided for the core, allowing the core to be instantiated with CoreConsole. Simulation, Synthesis, and Layout can be performed within Libero IDE. The RTL code for the core is obfuscated and some of the testbench source files are not provided; they are precompiled into the compiled simulation library instead.

### RTL

Complete RTL source code is provided for the core and testbenches.

### SmartDesign

CoreAES128 is preinstalled in the SmartDesign IP Deployment design environment. The core can be configured using the configuration GUI within SmartDesign, as shown in [Figure 4-1](#). For information on using SmartDesign to configure, connect, and generate cores, refer to the Libero IDE online help.

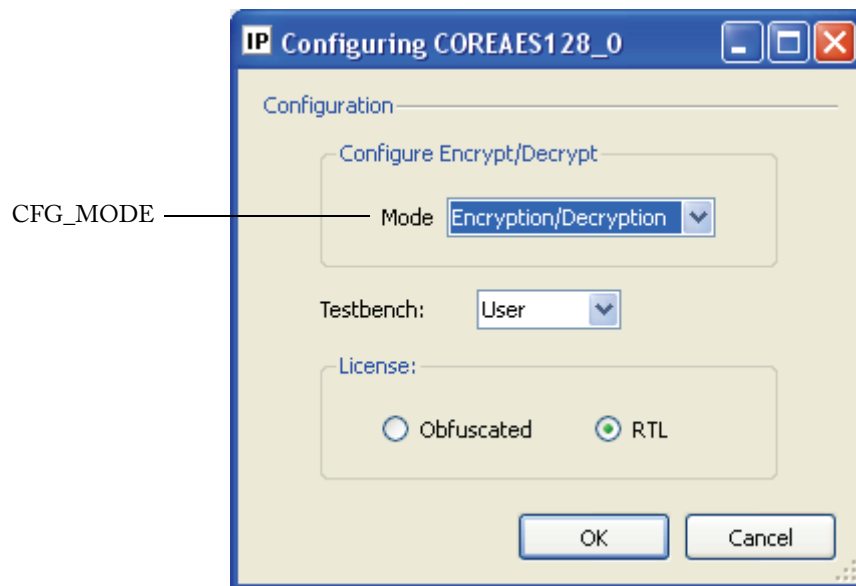


Figure 4-1 · CoreAES128 Configuration Window

### Simulation Flows

The user testbench for CoreAES128 is included in all releases. To run simulation, select the user testbench flow within CoreConsole and click **Save & Generate** on the Generate pane. The user testbench is selected through the Core Testbench Configuration GUI. When CoreConsole generates the Libero IDE project, it installs the user testbench files. To run the user testbench, set the design root to the CoreAES128 instantiation in the Libero IDE design hierarchy pane, and click the Simulation icon in the Libero IDE Design Flow window. This will invoke ModelSim® and automatically run the simulation.

## Synthesis in Libero IDE

Click the Synthesis icon in Libero IDE. The Synthesis window appears, displaying the Synplicity® project. Set Synplicity to use the Verilog 2001 standard if Verilog is being used. To run Synthesis, select the Run icon.

## Place-and-Route in Libero IDE

After running Synthesis, click the Layout icon in the Libero IDE to invoke Designer. CoreAES128 requires no special place-and-route settings.

# Testbench Operation

## User Testbench

An example user testbench is included with the obfuscated and RTL releases of CoreAES128. The obfuscated and RTL releases provide the precompiled ModelSim format, as well as the source code for the user testbench to ease the process of integrating and verifying the CoreAES128 macro into a design. A block diagram of the example user design and testbench is shown in Figure 5-1.

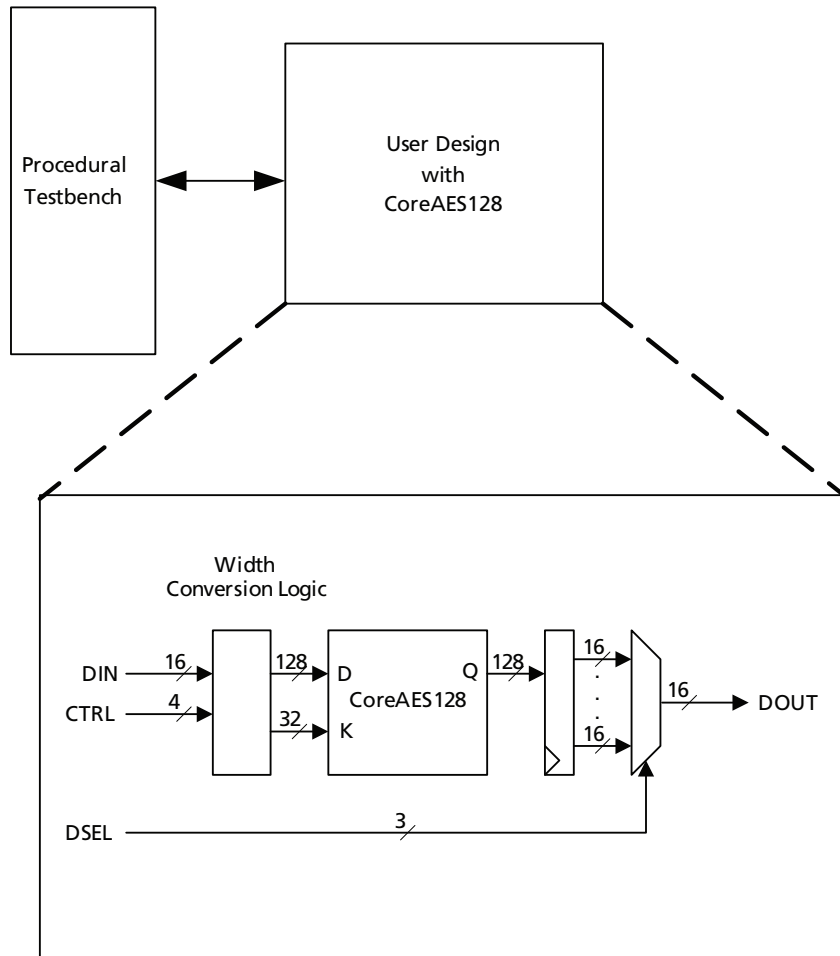


Figure 5-1 · Example User Design and User Testbench

The user testbench includes a simple example design that serves as a reference for users that want to implement their own designs. RTL source code for the example design and user testbench, shown in Figure 5-1, "wraps" around the CoreAES128 macro.

The source code for each user testbench includes example support routines to aid the user in testing an embedded system that contains the CoreAES128 macro. Refer to the comments in the user testbench source code for details on the support routines (tasks for Verilog testbenches, functions and procedures for VHDL testbenches.)



## Ordering Information

### Ordering Codes

CoreAES128 can be ordered through your local Actel sales representative. It should be ordered using the following number scheme: CoreAES128-XX, where XX is listed in [Table 6-1](#).

Table 6-1 · Ordering Codes

XX	Description
OM	OM for Obfuscated RTL source – multiple-use license
RM	RTL for RTL source – multiple-use license

*Note:* CoreAES128-OM is included free in the Libero IDE Catalog if you have a valid Libero IDE license



---

## Export Restrictions

CoreAES128 is subject to strict export controls and is licensable under the U.S. Department of Commerce Export Administration Regulations, the U.S. Department of State International Traffic in Arms Regulations, or other laws, government regulations, or restrictions. Actel is in the process of obtaining additional permissions to ship CoreAES128 to a wider audience. The licensee will not import, export, re-export, divert, transfer, or disclose CoreAES128 without complying strictly with the export control laws and all legal requirements in the relevant jurisdictions, including, without limitation, obtaining the prior approval of the U.S. Department of Commerce or the U.S. Department of State, as applicable.



---

## Product Support

Actel backs its products with various support services including Customer Service, a Customer Technical Support Center, a web site, an FTP site, electronic mail, and worldwide sales offices. This appendix contains information about contacting Actel and using these support services.

### Customer Service

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

From Northeast and North Central U.S.A., call 650.318.4480

From Southeast and Southwest U.S.A., call 650.318.4480

From South Central U.S.A., call 650.318.4434

From Northwest U.S.A., call 650.318.4434

From Canada, call 650.318.4480

From Europe, call 650.318.4252 or +44 (0) 1276 401 500

From Japan, call 650.318.4743

From the rest of the world, call 650.318.4743

Fax, from anywhere in the world 650.318.8044

### Actel Customer Technical Support Center

Actel staffs its Customer Technical Support Center with highly skilled engineers who can help answer your hardware, software, and design questions. The Customer Technical Support Center spends a great deal of time creating application notes and answers to FAQs. So, before you contact us, please visit our online resources. It is very likely we have already answered your questions.

### Actel Technical Support

Visit the [Actel Customer Support website \(www.actel.com/custsup/search.html\)](http://www.actel.com/custsup/search.html) for more information and support. Many answers available on the searchable web resource include diagrams, illustrations, and links to other resources on the Actel web site.

### Website

You can browse a variety of technical and non-technical information on Actel's [home page](http://www.actel.com), at [www.actel.com](http://www.actel.com).

### Contacting the Customer Technical Support Center

Highly skilled engineers staff the Technical Support Center from 7:00 A.M. to 6:00 P.M., Pacific Time, Monday through Friday. Several ways of contacting the Center follow:

#### Email

You can communicate your technical questions to our email address and receive answers back by email, fax, or phone. Also, if you have design problems, you can email your design files to receive assistance. We constantly monitor the email account throughout the day. When sending your request to us, please be sure to include your full name, company name, and your contact information for efficient processing of your request.

The technical support email address is [tech@actel.com](mailto:tech@actel.com).

## Phone

Our Technical Support Center answers all calls. The center retrieves information, such as your name, company name, phone number and your question, and then issues a case number. The Center then forwards the information to a queue where the first available application engineer receives the data and returns your call. The phone hours are from 7:00 A.M. to 6:00 P.M., Pacific Time, Monday through Friday. The Technical Support numbers are:

**650.318.4460**  
**800.262.1060**

Customers needing assistance outside the US time zones can either contact technical support via email ([tech@actel.com](mailto:tech@actel.com)) or contact a local sales office. [Sales office listings](http://www.actel.com/contact/offices/index.html) can be found at [www.actel.com/contact/offices/index.html](http://www.actel.com/contact/offices/index.html).

# Index

## A

- abort 17
- Actel
  - electronic mail 27
  - telephone 28
  - web-based technical support 27
  - website 27

## C

- cipher key 6
- cipher key expansion 13
- clear 17
- contacting Actel
  - customer service 27
  - electronic mail 27
  - telephone 28
  - web-based technical support 27
- CoreAES128
  - block diagram 11
  - I/O signals 9
  - initialization 13
  - key features 6
  - operation 13
  - overview 5
  - parameters/generics 10
  - version 6
- customer service 27

## D

- decryption 15
- design description 9
- design security 6

## E

- encryption 14

## L

- license 19, 21
  - obfuscated 19

- RTL 19

## O

- operation modes 18
- ordering code 23

## P

- pause 16
- product support 27–28
  - customer service 27
  - electronic mail 27
  - technical support 27
  - telephone 28
  - website 27

## R

- resume 16
- Rijndael algorithm 5

## S

- supported families 7

## T

- technical support 27
- testbenches
  - operation 23, 25
- tool flow
  - place-and-route 20
  - simulation 19
  - SmartDesign 19
  - synthesis 20
  - user testbench 21

## U

- utilization and performance 7

## W

- web-based technical support 27



**Actel is the leader in low-power and mixed-signal FPGAs and offers the most comprehensive portfolio of system and power management solutions. Power Matters. Learn more at [www.actel.com](http://www.actel.com).**

**Actel Corporation** • 2061 Stierlin Court • Mountain View, CA 94043 • USA

Phone 650.318.4200 • Fax 650.318.4600 • Customer Service: 650.318.1010 • Customer Applications Center: 800.262.1060

**Actel Europe Ltd.** • River Court, Meadows Business Park • Station Approach, Blackwater • Camberley Surrey GU17 9AB • United Kingdom

Phone +44 (0) 1276 609 300 • Fax +44 (0) 1276 607 540

**Actel Japan** • EXOS Ebisu Building 4F • 1-24-14 Ebisu Shibuya-ku • Tokyo 150 • Japan

Phone +81.03.3445.7671 • Fax +81.03.3445.7668 • <http://jp.actel.com>

**Actel Hong Kong** • Room 2107, China Resources Building • 26 Harbour Road • Wanchai • Hong Kong

Phone +852 2185 6460 • Fax +852 2185 6488 • [www.actel.com.cn](http://www.actel.com.cn)