



AES256SS IP Demo Instruction

1	Environment Setup.....	1
2	ZCU106 board setup.....	2
3	Serial Console.....	2
4	Command detail and testing result.....	3
4.1	Set encryption/decryption key.....	3
4.2	Show Data Memory.....	4
4.3	Fill Plain Data Memory.....	5
4.4	Fill Cipher Data Memory.....	6
4.5	Encrypt.....	7
4.6	Decrypt.....	8
5	Revision History.....	9

AES256SS IP Demo Instruction

Rev1.02 2-Jun-2023

This document describes the instruction to demonstrate the operation of AES256SSIP on ZCU106 Evaluation Board. In the demonstration, AES256SSIP are used to encrypt and decrypt data between two memories in FPGA. User can fill memory with plain or cipher data patterns, set encryption/decryption key and control test operation via serial console.

1 Environment Setup

To operate AES256SSIP demo, please prepare following test environment.

- 1) FPGA development boards (ZCU106 board).
- 2) Test PC.
- 3) Micro USB cable for JTAG connection connecting between ZCU106 board and Test PC.
- 4) Micro USB cable for UART connection connecting between ZCU106 board and Test PC.
- 5) Vivado tool for programming FPGA installed on Test PC.
- 6) Serial console software such as TeraTerm installed on PC. The setting on the console is
- 7) Baudrate=115200, Data=8-bit, Non-parity and Stop=1.
- 8) Batch file named "AES256SSIPTest_ZCU106.bat" (To download these files, please visit our web site at www.design-gateway.com)

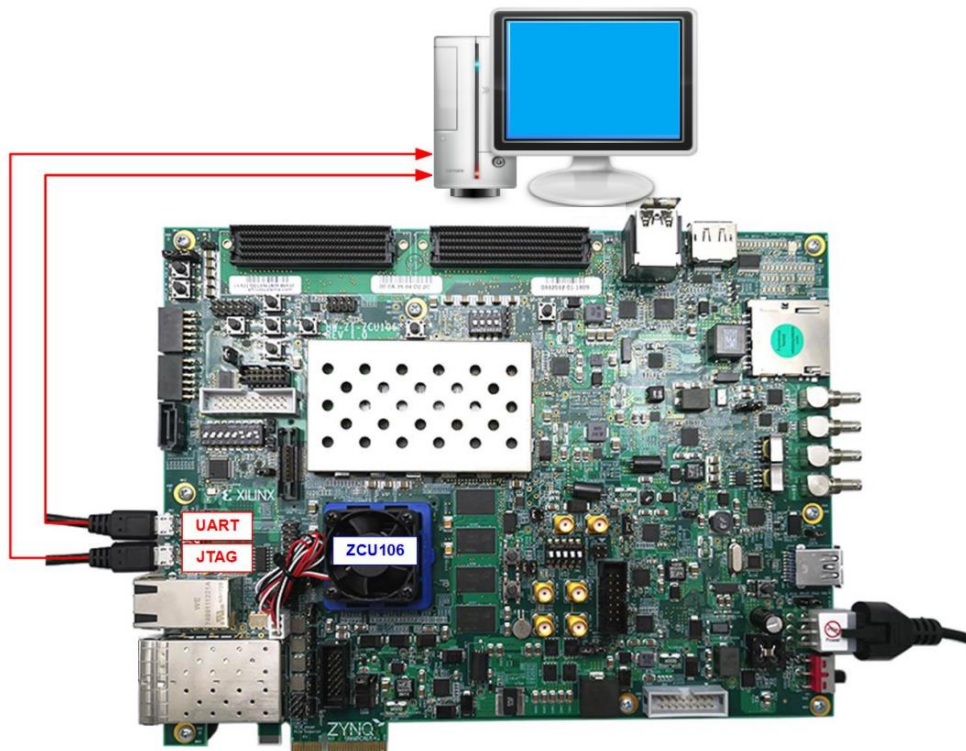
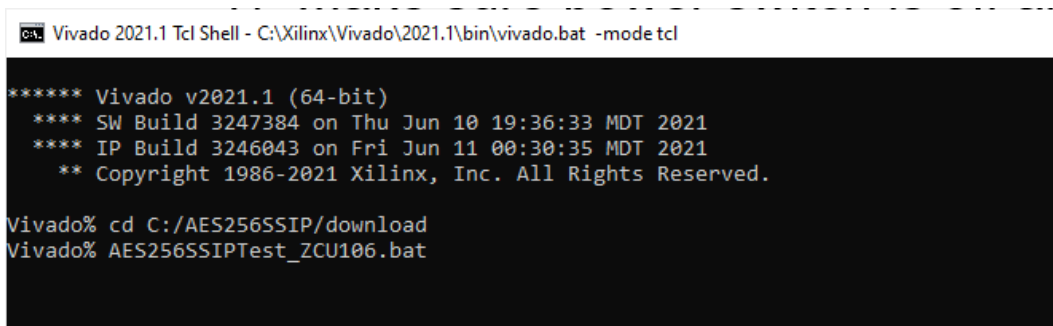


Figure 1-1 AES256SSIP demo environment on ZCU106 board

2 ZCU106 board setup

- 1) Make sure power switch is off and connect power supply to FPGA development board.
- 2) Connect two USB cables between FPGA board and PC via micro-USB ports.
- 3) Power on system.
- 4) Download configuration file and firmware to FPGA board by following step,
 - a) open Vivado TCL shell.
 - b) change current directory to download folder which includes demo configuration file.
 - c) Type “AES256SSIPTest_ZCU106.bat”, as shown in Figure 2-1.



```

C:\Xilinx\Vivado\2021.1\bin\vivado.bat -mode tcl

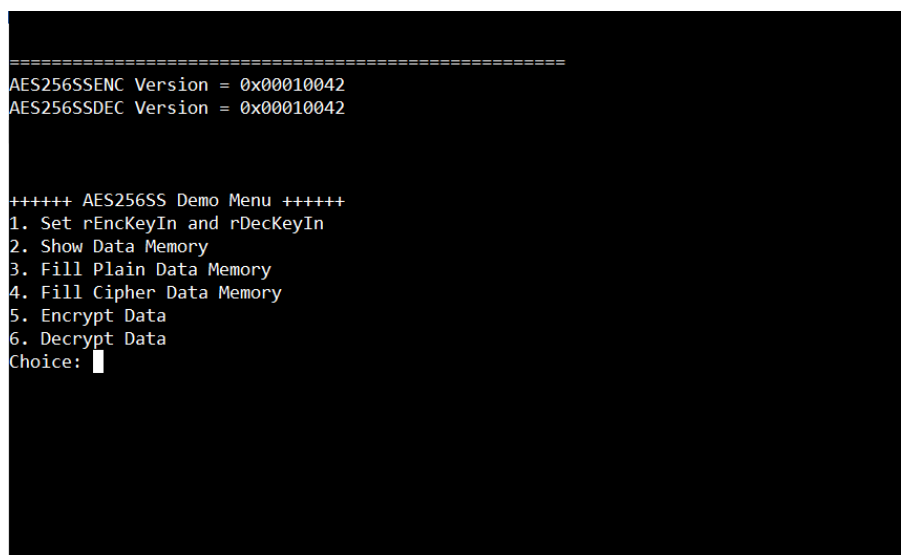
***** Vivado v2021.1 (64-bit)
**** SW Build 3247384 on Thu Jun 10 19:36:33 MDT 2021
**** IP Build 3246043 on Fri Jun 11 00:30:35 MDT 2021
** Copyright 1986-2021 Xilinx, Inc. All Rights Reserved.

Vivado% cd C:/AES256SSIP/download
Vivado% AES256SSIPTest_ZCU106.bat
  
```

Figure 2-1 Example command script for download configuration file

3 Serial Console

User can fill RAMs with plain or cipher data patterns, set encryption/decryption key and control test operation via serial console. When configuration is completed, AES256SSdemo command menu will be displayed as shown in Figure 3-1. The detailed information of each menu is described in topic 4.



```

=====
AES256SSENC Version = 0x00010042
AES256SSDEC Version = 0x00010042

++++++ AES256SS Demo Menu ++++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: █
  
```

Figure 3-1 Serial console

4 Command detail and testing result

4.1 Set encryption/decryption key

Step to set encryption key and decryption key as follows

- a) Select “1. Set rEncKeyIn and rDecKeyIn”.
- b) Current encryption key will be displayed on serial console as shown in Figure 4-1.
- c) Set new encryption key: User is allowed to input new key in hex format or press “enter” to skip setting new key. Then the current encryption key is printed again.
- d) Current decryption key will be displayed on serial console.
- e) Set new decryption key: User is allowed to input new key in hex format or press “enter” to use rEncKeyIn as rDecKeyIn. Then the current decryption key is printed again.

```

+++++ AES256SS Demo Menu +++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 1

+++ Set rEncKeyIn and rDecKeyIn +++
   rEncKeyIn = 0x0000000000000000000000000000000000000000000000000000000000000000
   (enter to skip)= 0x00112233445566778899aabbccddeeff00112233445566778899aabbccddeeff
   new rEncKeyIn = 0x00112233445566778899AABBCCDDEEFF00112233445566778899AABBCCDDEEFF

   rDecKeyIn = 0x0000000000000000000000000000000000000000000000000000000000000000
   (enter to use rEncKeyIn)= 0x
   new rDecKeyIn = 0x00112233445566778899AABBCCDDEEFF00112233445566778899AABBCCDDEEFF

```

Figure 4-1 Set rEncKeyIn and rDecKeyIn example

4.2 Show Data Memory

To show data in memory, user can select “2. Show Data Memory” and input the desired number of 128-bit data to show. Both plain data and cipher data will be displayed in table-form as shown in Figure 4-2. User can press “enter” key to skip putting the number of data, then serial console will display five rows of 128-bit plain data and 128-bit cipher data at address 0x0000-0x004F.

```

+++++ AES256SS Demo Menu +++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 2

+++ Show Data Memory +++
Number of 128-bit Data in decimal (enter = 5): 20

Addr#   Plain Data                               Cipher Data
.F....C .B....8 .7....4 .3....0 .F....C .B....8 .7....4 .3....0
0000: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0010: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0020: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0030: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0040: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0050: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0060: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0070: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0080: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0090: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00A0: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00B0: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00C0: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00D0: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00E0: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00F0: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0100: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0110: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0120: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0130: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
...   ...

```

Figure 4-2 Displayed Data example

4.3 Fill Plain Data Memory

Step to fill plain data in memory as follows

- a) Select “3. Fill Plain Data Memory”.
- b) There are four pattern to fill memory.
 - a. zero pattern
 - b. 8-bit counter
 - c. 16-bit counter
 - d. 32-bit counter
- c) Whole plain data memory is filled with selected data pattern.

```

+++++ AES256SS Demo Menu +++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 3

+++ Fill Plain Data Memory +++
a. zero pattern
b. 8-bit counter
c. 16-bit counter
d. 32-bit counter
Choice: b

          Plain Data                      Cipher Data
Addr#    .F.....C .B.....8 .7.....4 .3.....0 .F.....C .B.....8 .7.....4 .3.....0
0000:    0F0E0D0C 0B0A0908 07060504 03020100 00000000 00000000 00000000 00000000
0010:    1F1E1D1C 1B1A1918 17161514 13121110 00000000 00000000 00000000 00000000
0020:    2F2E2D2C 2B2A2928 27262524 23222120 00000000 00000000 00000000 00000000
0030:    3F3E3D3C 3B3A3938 37363534 33323130 00000000 00000000 00000000 00000000
0040:    4F4E4D4C 4B4A4948 47464544 43424140 00000000 00000000 00000000 00000000
...      ...

```

Figure 4-3 Displayed Data when select pattern

4.4 Fill Cipher Data Memory

Step to fill Cipher data in memory as follows

- a) Select “4. Fill Cipher Data Memory”.
- b) There are four pattern to fill memory.
 - a. zero pattern
 - b. 8-bit counter
 - c. 16-bit counter
 - d. 32-bit counter
- c) Whole cipher data memory is filled with selected data pattern.

```

+++++ AES256SS Demo Menu +++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 4

+++ Fill Cipher Data Memory +++
a. zero pattern
b. 8-bit counter
c. 16-bit counter
d. 32-bit counter
Choice: c

          Plain Data
Addr#   .F.....C .B.....8 .7.....4 .3.....0
0000:  0F0E0D0C 0B0A0908 07060504 03020100
0010:  1F1E1D1C 1B1A1918 17161514 13121110
0020:  2F2E2D2C 2B2A2928 27262524 23222120
0030:  3F3E3D3C 3B3A3938 37363534 33323130
0040:  4F4E4D4C 4B4A4948 47464544 43424140
...     ...

          Cipher Data
Addr#   .F.....C .B.....8 .7.....4 .3.....0
0000:  00070006 00050004 00030002 00010000
0010:  000F000E 000D000C 000B000A 00090008
0020:  00170016 00150014 00130012 00110010
0030:  001F001E 001D001C 001B001A 00190018
0040:  00270026 00250024 00230022 00210020
...     ...

```

Figure 4-4 Displayed Data when select pattern

4.5 Encrypt

Select “5. Encrypt” to encrypt plain data in memory. User can input the desired number of plain data to encrypt or press “enter” key to encrypt five 128-bit plain data. When the encryption process is finished, DpRam2 will be filled with cipher data from AES256SSENC.

In this example, Number of 128-bit Plain Data was ‘5’ (default value). When AES256SSENC finished encryption process, five rows of plain data and cipher data were displayed on serial console as shown in Figure 4-5.

```

+++++ AES256SS Demo Menu +++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 5

+++ Encrypt +++
Number of 128-bit Data in decimal (enter = 5):

          Plain Data          Cipher Data
Addr#  .F.....C .B.....8 .7.....4 .3.....0  .F.....C .B.....8 .7.....4 .3.....0
0000: 0F0E0D0C 0B0A0908 07060504 03020100 0BB89F0 DCAE168A AFEC764 A225AE59
0010: 1F1E1D1C 1B1A1918 17161514 13121110 22588148 85F97E0E 539CC061 3346D8F4
0020: 2F2E2D2C 2B2A2928 27262524 23222120 E9C22525 90A0049E 8DFB21F9 83DFA464
0030: 3F3E3D3C 3B3A3938 37363534 33323130 D0697778 3D8BEAE9 191AD8B6 8043082B
0040: 4F4E4D4C 4B4A4948 47464544 43424140 E156BF4B 26D865EC 6CBFFD4E EC10F657
...    ...

```

Figure 4-5 Serial console after finished encryption process

4.6 Decrypt

Select “6. Decrypt” to decrypt cipher data in memory. User can input the desired number of cipher data to decrypt or press “enter” key to decrypt five 128-bit Cipher data. When the decryption process is finished, DpRam1 will be filled with plain data from AES256SSDEC.

In this example, Number of 128-bit Cipher Data was ‘5’ (default value). When AES256SSDEC finished decryption process, five rows of plain data and cipher data were displayed on serial console as shown in Figure 4-6.

```

+++++ AES256SS Demo Menu +++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 6

+++ Decrypt +++
Number of 128-bit Data in decimal (enter = 5):

      Plain Data                                Cipher Data
Addr# .F.....C .B.....8 .7.....4 .3.....0 .F.....C .B.....8 .7.....4 .3.....0
0000: 0F0E0D0C 0B0A0908 07060504 03020100 0BBA89F0 DCAE168A AFCD764 A225AE59
0010: 1F1E1D1C 1B1A1918 17161514 13121110 22588148 85F97E0E 539CC061 3346D8F4
0020: 2F2E2D2C 2B2A2928 27262524 23222120 E9C22525 90A0049E 8DFB21F9 83DFA464
0030: 3F3E3D3C 3B3A3938 37363534 33323130 D0697778 3D8BEAE9 191AD8B6 8043082B
0040: 4F4E4D4C 4B4A4948 47464544 43424140 E156BF4B 26D865EC 6CBFFD4E EC10F657
...     ...

```

Figure 4-6 Serial console after finished decryption process

5 Revision History

Revision	Date	Description
1.00	1-Sep-2022	Initial version release
1.02	16-Sep-2022	Update description for new design