AES128IP Demo Instruction

AES128IP Demo Instruction

Rev1.02 25-May-2023

This document describes the instruction to demonstrate the operation of AES128IP on FPGA development boards. In the demonstration, AES128IP 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 Nios II Command Shell.

1 Environment Setup

To operate AES128IP demo, please prepare following test environment.

1) FPGA development board
   - Agilex F-series development kit
   - Arria10 SoC Development board
2) Test PC.
3) Micro USB cable for JTAG connection connecting between FPGA boards and Test PC.
4) Quartus programmer for programming FPGA and Nios II command shell, installed on PC.
5) SOF file named "AES128.sof" (To download these files, please visit our web site at www.design-gateway.com)

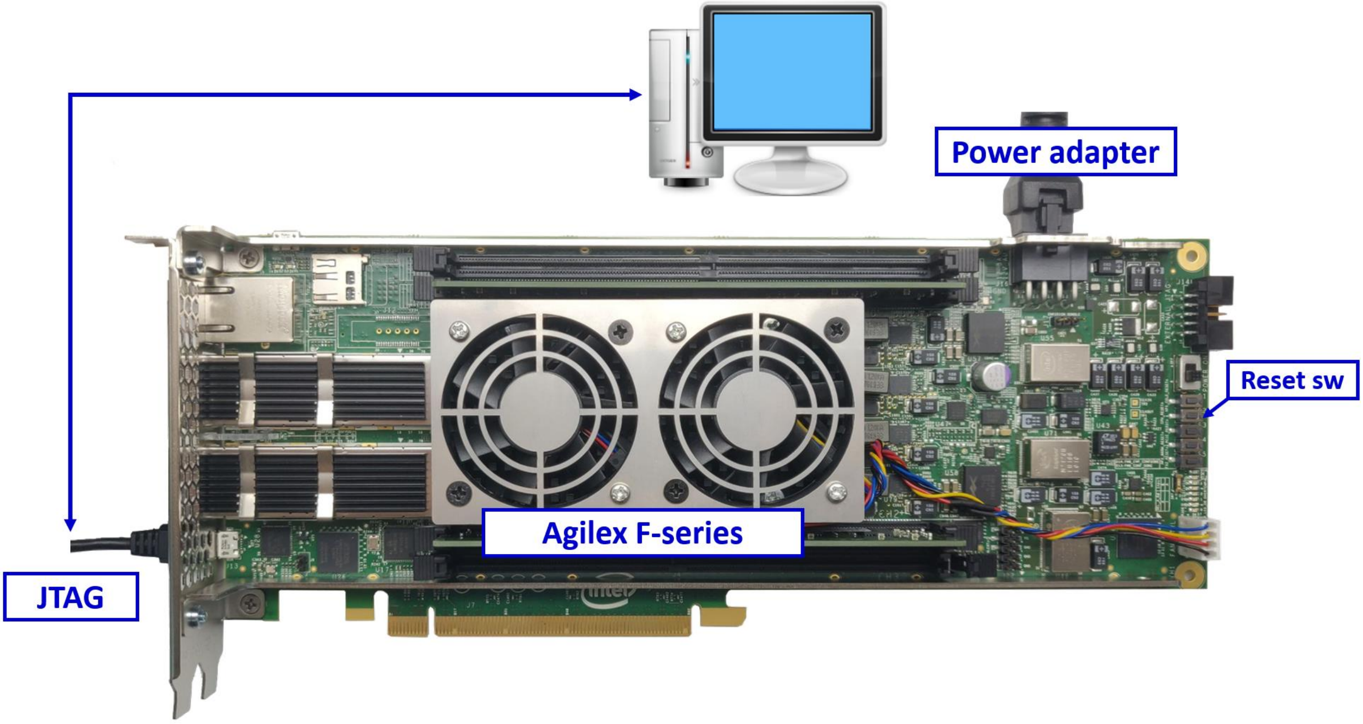


Figure 1-1 AES128IP demo environment on Agilex F-series board

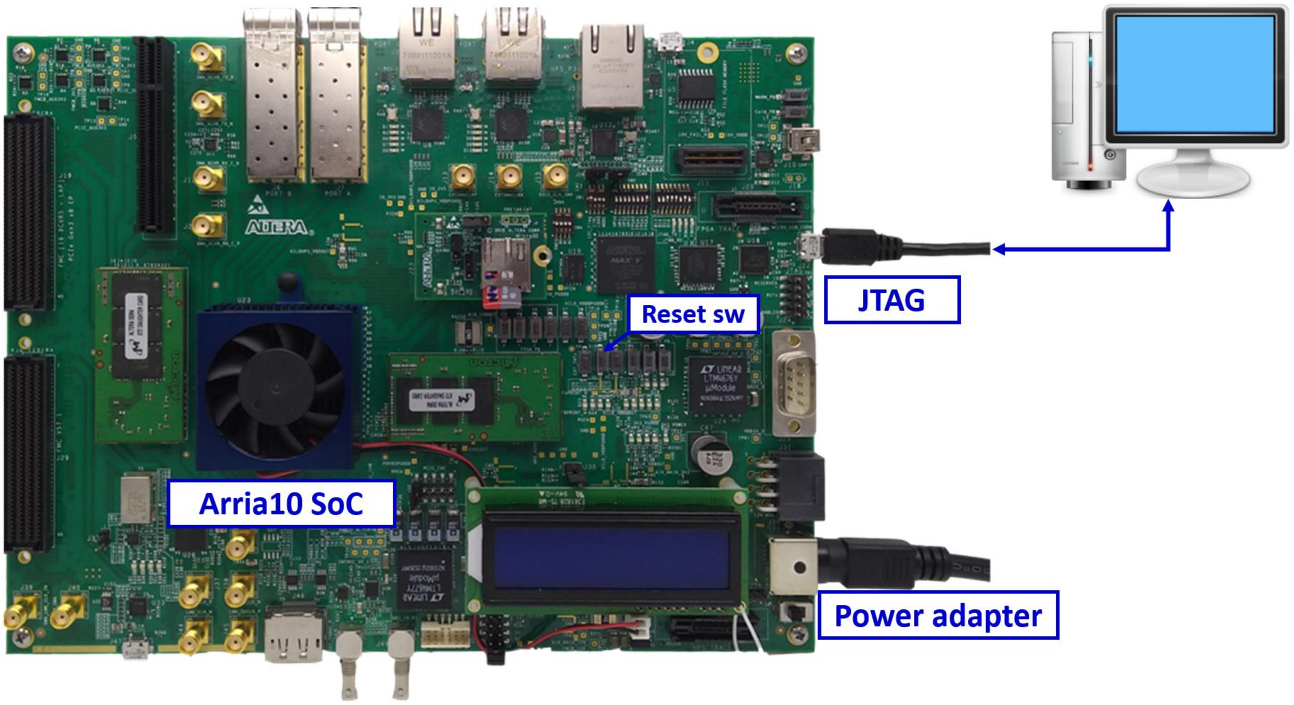
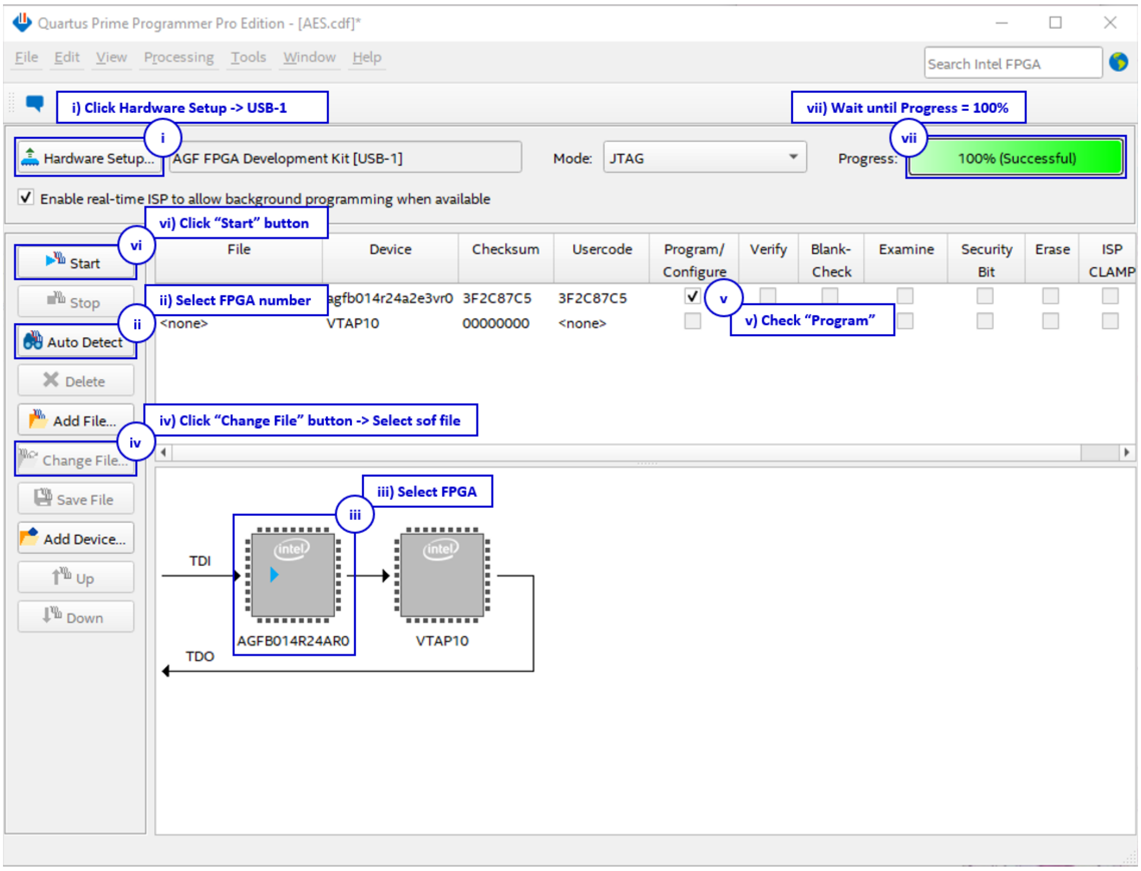


Figure 1-2 AES128IP demo environment on Arria10 SoC board

2 FPGA development board setup

1) Make sure power switch is off and connect power supply to FPGA development board.
2) Connect USB cables between FPGA board and PC via micro-USB ports.
3) Turn on power switch for FPGA board.
4) Open Quartus Programmer to program FPGA through USB-1 by following step.
   i) Click “Hardware Setup…” to select
      - AGF FPGA Development Kit [USB-1] for Agilex F-series
      - USB-BlasterII [USB-1] for Arria10 SoC
   ii) Click “Auto Detect” and select FPGA number.
   iii) Select FPGA device icon (Agilex or A10SoC).
   iv) Click “Change File” button, select SOF file in pop-up window and click “open” button.
   v) Check “program”.
   vi) Click “Start” button to program FPGA.
   vii) Wait until Progress status is equal to 100%.



i) Click Hardware Setup -> USB-1

vii) Wait until Progress = 100%

Hardware Setup... AGF FPGA Development Kit [USB-1]

Mode: JTAG

Progress: 100% (Successful)

☑ Enable real-time ISP to allow background programming when available

vi) Click “Start” button

ii) Select FPGA number

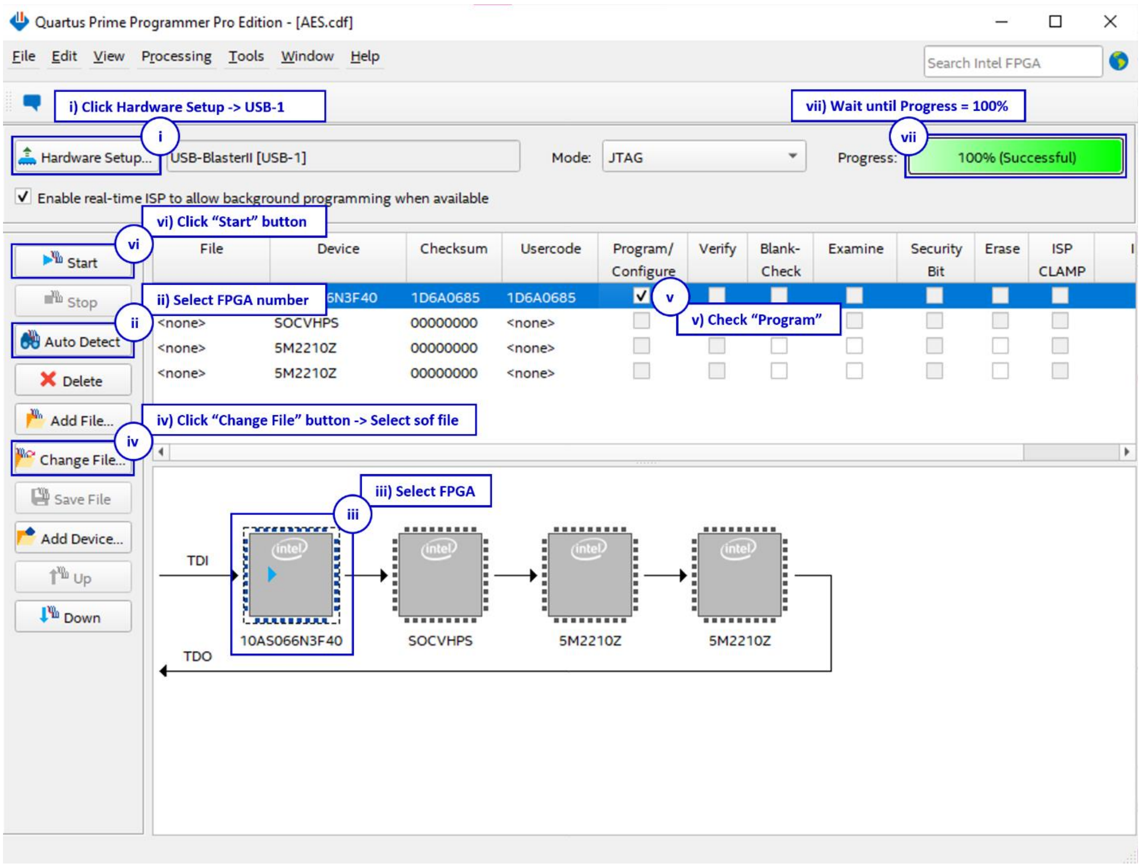
v) Check “Program”

iv) Click “Change File” button -> Select sof file

iii) Select FPGA

File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank-Check	Examine	Security Bit	Erase	ISP CLAMP
	agfb014r24a2e3vr0	3F2C87C5	3F2C87C5	☑	☐	☐	☐	☐	☐	☐
<none>	VTAP10	00000000	<none>	☐			☐	☐	☐	☐

Figure 2-1 FPGA Programmer for Agilex



Quartus Prime Programmer Pro Edition - [AES.cdf]

i) Click Hardware Setup -> USB-1

ii) Select FPGA number

iii) Select FPGA

iv) Click "Change File" button -> Select sof file

v) Check "Program"

vi) Click "Start" button

vii) Wait until Progress = 100%

Hardware Setup... USB-BlasterII [USB-1]

Mode: JTAG

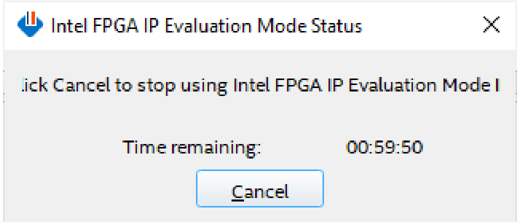
Progress: 100% (Successful)

☑ Enable real-time ISP to allow background programming when available

File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank- Check	Examine	Security Bit	Erase	ISP CLAMP
	6N3F40	1D6A0685	1D6A0685	☑	☐	☐	☐	☐	☐	☐
<none>	SOCVHPS	00000000	<none>	☐	☐	☐	☐	☐	☐	☐
<none>	5M2210Z	00000000	<none>	☐	☐	☐	☐	☐	☐	☐
<none>	5M2210Z	00000000	<none>	☐	☐	☐	☐	☐	☐	☐

Figure 2-2 FPGA Programmer for A10SoC

For A10SoC after program SOF file complete, Quartus Prime will show popup message of Intel FPGA IP Evaluation Mode Status as shown in Figure 2-3. Please do not press cancel button.



Intel FPGA IP Evaluation Mode Status

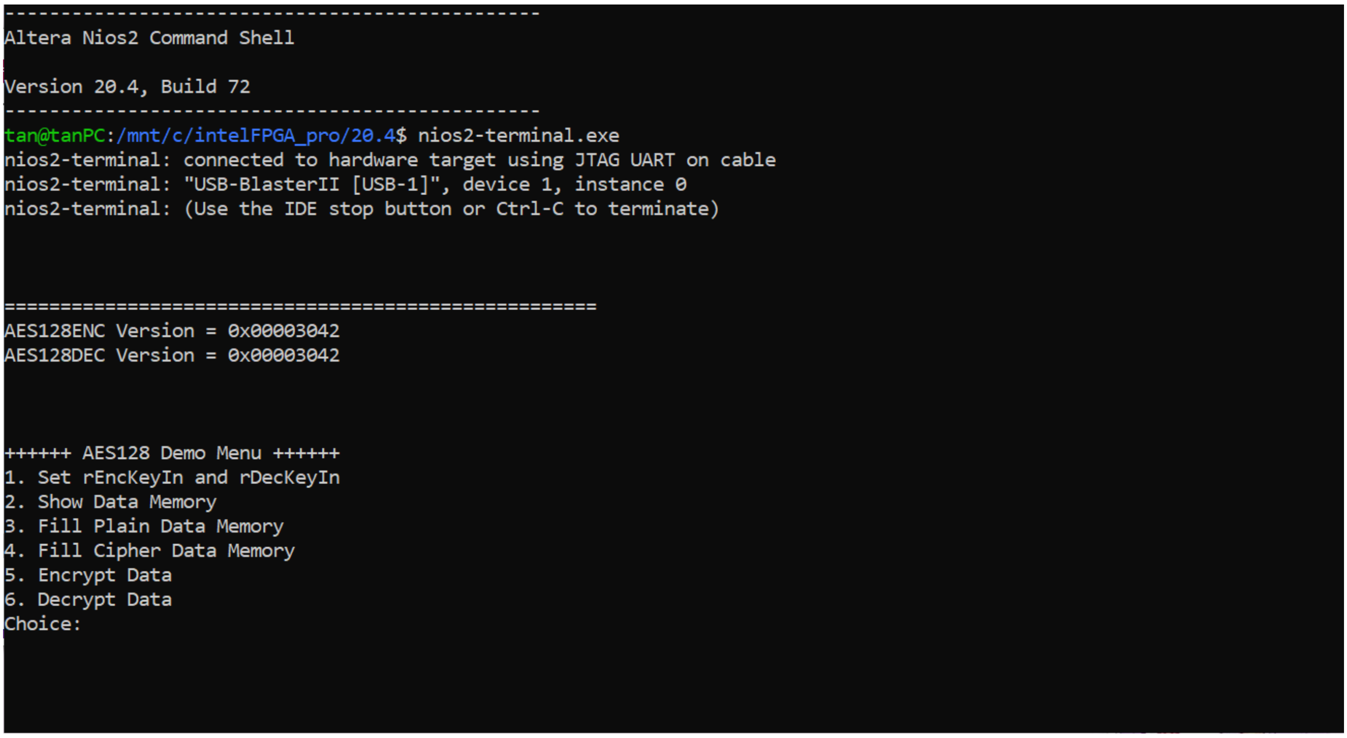
.ick Cancel to stop using Intel FPGA IP Evaluation Mode I

Time remaining: 00:59:50

Figure 2-3 Intel FPGA IP Evaluation Mode Status

3 Nios II Command Shell

User can fill RAMs with plain or cipher data patterns, set encryption/decryption key and control test operation via Nios II Command Shell. When configuration is completed, AES128demo command menu will be displayed as shown in Figure 3-1. The detailed information of each menu is described in topic 4.



```
------------------------------------------------
Altera Nios2 Command Shell

Version 20.4, Build 72
------------------------------------------------
tan@tanPC:/mnt/c/intelFPGA_pro/20.4$ nios2-terminal.exe
nios2-terminal: connected to hardware target using JTAG UART on cable
nios2-terminal: "USB-BlasterII [USB-1]", device 1, instance 0
nios2-terminal: (Use the IDE stop button or Ctrl-C to terminate)



=====================================================
AES128ENC Version = 0x00003042
AES128DEC Version = 0x00003042



++++++ AES128 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:
```

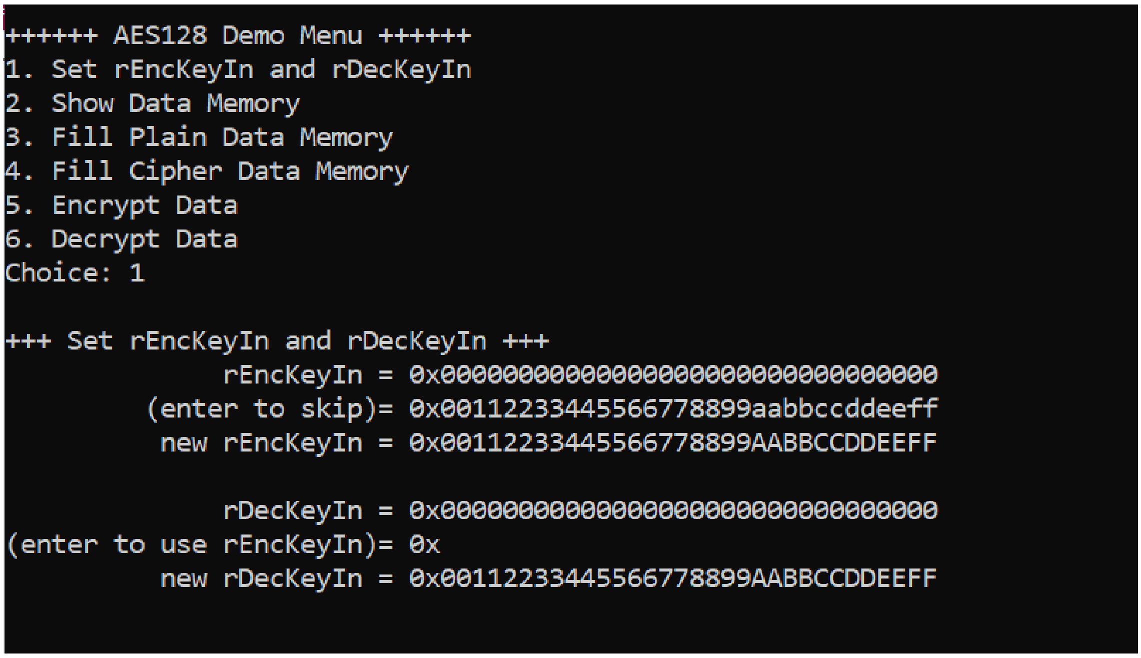
Figure 3-1 Nios II Command Shell

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 Nios II Command Shell 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 Nios II Command Shell.
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.

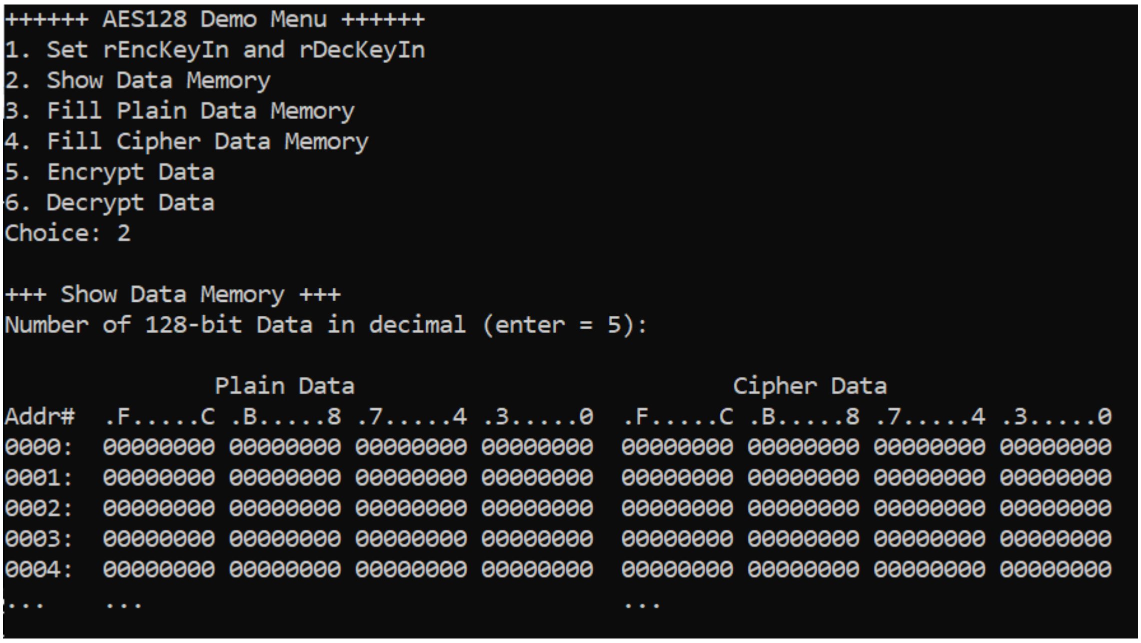


```
++++++ AES128 Demo Menu ++++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 1

+++ Set rEncKeyIn and rDecKeyIn +++
              rEncKeyIn = 0x00000000000000000000000000000000
         (enter to skip)= 0x00112233445566778899aabbccddeeff
          new rEncKeyIn = 0x00112233445566778899AABBCCDDEEFF

              rDecKeyIn = 0x00000000000000000000000000000000
(enter to use rEncKeyIn)= 0x
          new rDecKeyIn = 0x00112233445566778899AABBCCDDEEFF
```

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 Nios II Command Shell will display five rows (default value) of 128-bit plain data and 128-bit cipher data at address 0x0000-0x004F.



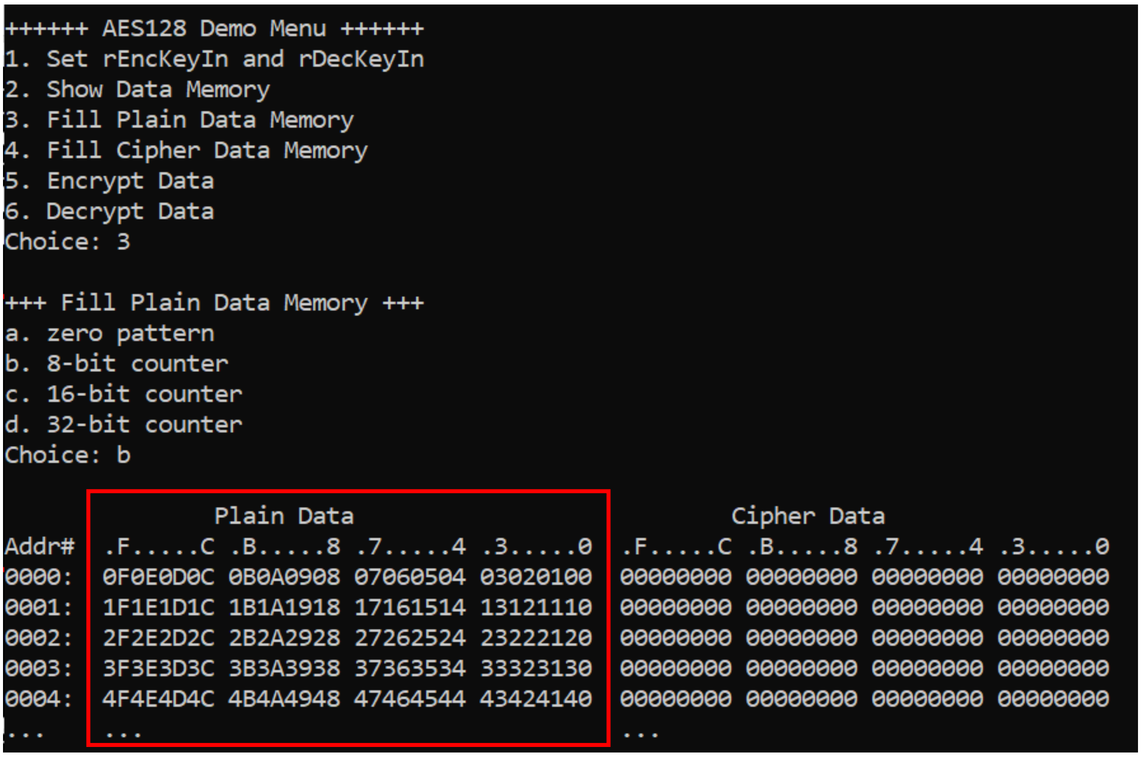
```
++++++ AES128 Demo Menu ++++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 2

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

               Plain Data                                Cipher Data
Addr#  .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
0001:  00000000 00000000 00000000 00000000  00000000 00000000 00000000 00000000
0002:  00000000 00000000 00000000 00000000  00000000 00000000 00000000 00000000
0003:  00000000 00000000 00000000 00000000  00000000 00000000 00000000 00000000
0004:  00000000 00000000 00000000 00000000  00000000 00000000 00000000 00000000
...    ...                                  ...
```

Figure 4-2 Displayed Data when press “enter” key

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.



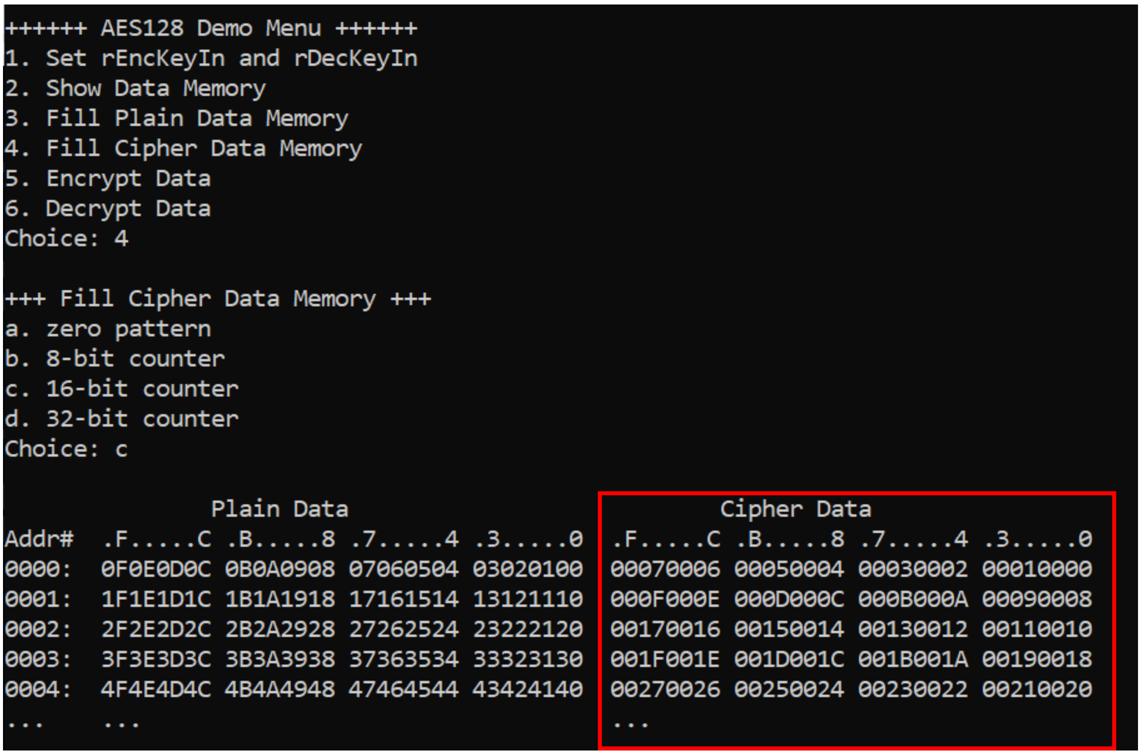
```
++++++ AES128 Demo Menu ++++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 3

+++ 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
0001:  1F1E1D1C 1B1A1918 17161514 13121110  00000000 00000000 00000000 00000000
0002:  2F2E2D2C 2B2A2928 27262524 23222120  00000000 00000000 00000000 00000000
0003:  3F3E3D3C 3B3A3938 37363534 33323130  00000000 00000000 00000000 00000000
0004:  4F4E4D4C 4B4A4948 47464544 43424140  00000000 00000000 00000000 00000000
...    ...                                  ...
```

Figure 4-3 Displayed Data when select pattern b

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.

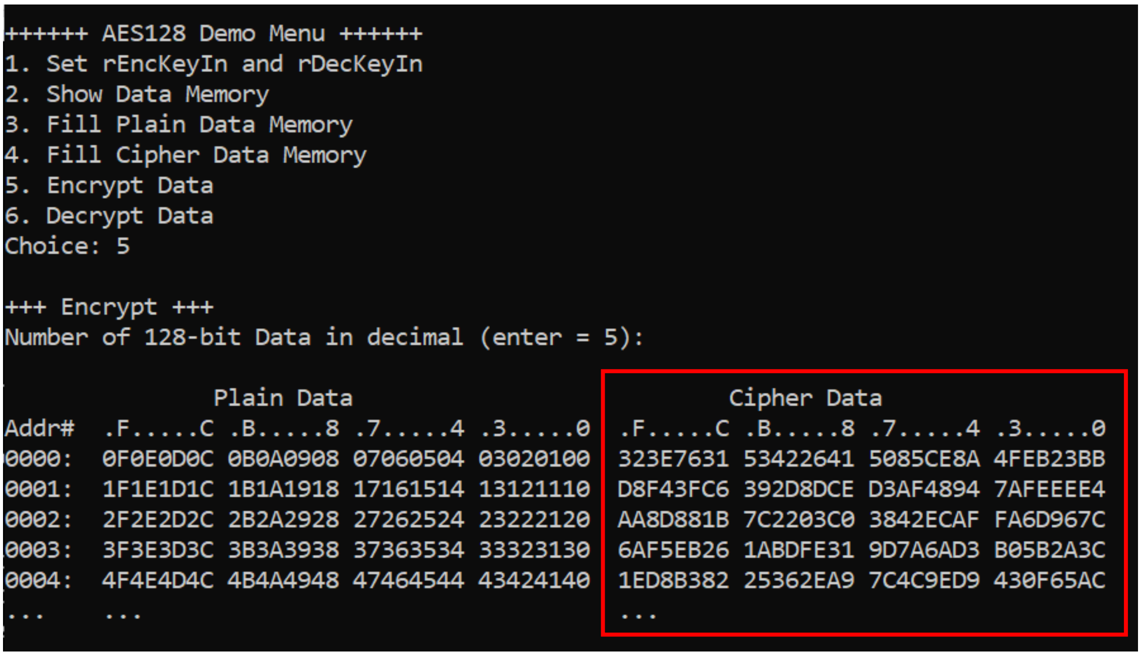


```
++++++ AES128 Demo Menu ++++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 4

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

               Plain Data                             Cipher Data
Addr#  .F.....C .B.....8 .7.....4 .3.....0  .F.....C .B.....8 .7.....4 .3.....0
0000:  0F0E0D0C 0B0A0908 07060504 03020100  00070006 00050004 00030002 00010000
0001:  1F1E1D1C 1B1A1918 17161514 13121110  000F000E 000D000C 000B000A 00090008
0002:  2F2E2D2C 2B2A2928 27262524 23222120  00170016 00150014 00130012 00110010
0003:  3F3E3D3C 3B3A3938 37363534 33323130  001F001E 001D001C 001B001A 00190018
0004:  4F4E4D4C 4B4A4948 47464544 43424140  00270026 00250024 00230022 00210020
...    ...                                  ...
```

Figure 4-4 Displayed Data when select pattern c

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 AES128ENC.

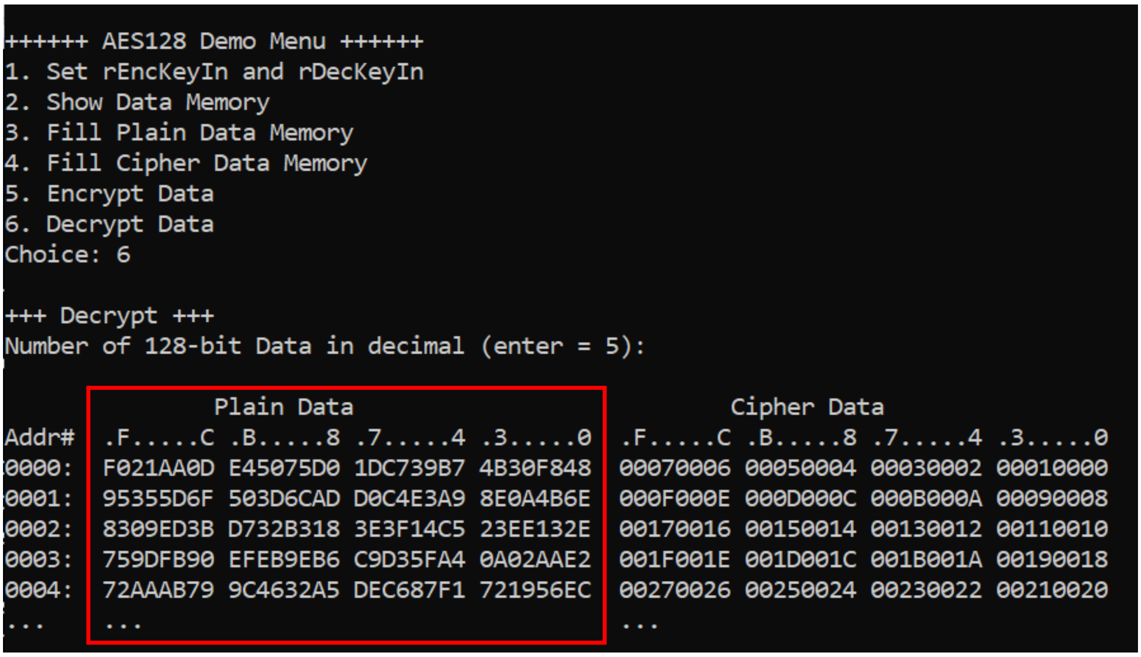


```
++++++ AES128 Demo Menu ++++++
1. Set rEncKeyIn and rDecKeyIn
2. Show Data Memory
3. Fill Plain Data Memory
4. Fill Cipher Data Memory
5. Encrypt Data
6. Decrypt Data
Choice: 5

+++ 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  323E7631 53422641 5085CE8A 4FEB23BB
0001:  1F1E1D1C 1B1A1918 17161514 13121110  D8F43FC6 392D8DCE D3AF4894 7AFEEEE4
0002:  2F2E2D2C 2B2A2928 27262524 23222120  AA8D881B 7C2203C0 3842ECAF FA6D967C
0003:  3F3E3D3C 3B3A3938 37363534 33323130  6AF5EB26 1ABDFE31 9D7A6AD3 B05B2A3C
0004:  4F4E4D4C 4B4A4948 47464544 43424140  1ED8B382 25362EA9 7C4C9ED9 430F65AC
...    ...                                  ...
```

Figure 4-5 Nios II Command Shell 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 AES128DEC.



```
++++++ AES128 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:  F021AA0D E45075D0 1DC739B7 4B30F848   00070006 00050004 00030002 00010000
0001:  95355D6F 503D6CAD D0C4E3A9 8E0A4B6E   000F000E 000D000C 000B000A 00090008
0002:  8309ED3B D732B318 3E3F14C5 23EE132E   00170016 00150014 00130012 00110010
0003:  759DFB90 EFEB9EB6 C9D35FA4 0A02AAE2   001F001E 001D001C 001B001A 00190018
0004:  72AAAB79 9C4632A5 DEC687F1 721956EC   00270026 00250024 00230022 00210020
...    ...                                   ...
```

Figure 4-6 Nios II Command Shell after finished decryption process

5 Revision History

Revision	Date	Description
1.00	5-May-2021	Initial version release
1.01	10-Sep-2021	Revised description information
1.02	27-Oct-2022	Update description for new design