AN-1394
APPLICATION NOTE

One Technology Way • P.O. Box 9106 • Norwood, MA 02062-9106, U.S.A. • Tel: 781.329.4700 • Fax: 781.461.3113 • www.analog.com

AES Encryption and Decryption for the ADF7023 and ADF7023-J

by Stephen Hinchy and Kalim Khan

INTRODUCTION

This application note describes the advanced encryption standard (AES) firmware module available for the ADF7023 and ADF7023-J transceivers (for the remainder of this application note, references to the ADF7023 also pertain to the ADF7023-J). The downloadable AES firmware module supports 128-bit block encryption and decryption with key sizes of 128 bits, 192 bits, and 256 bits. Two modes are supported: electronic codebook (ECB) mode and Cipher Block Chaining (CBC) Mode 1.

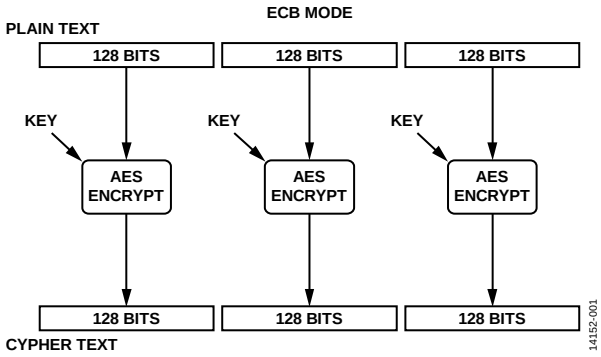

Figure 1. ECB Mode

ECB mode encrypts and decrypts on a 128-bit block by block with a single secret key as illustrated in Figure 1. CBC Mode 1 encrypts after first adding (via Modulo 2 arithmetic) a 128-bit user supplied initialization vector. The resulting cipher text is used as the initialization vector for the next block and so forth, as illustrated in Figure 2.

Decryption provides the inverse functionality. The firmware takes advantage of an on-chip hardware accelerator module to enhance throughput and minimize the latency of the AES processing.

The firmware module, which contains both Reed-Solomon (RS) forward error correction and AES encryption, is named rom_ram_7023_2_2_RS_AES.dat and can be found at www.analog.com/firmwaremodules-adf7023.

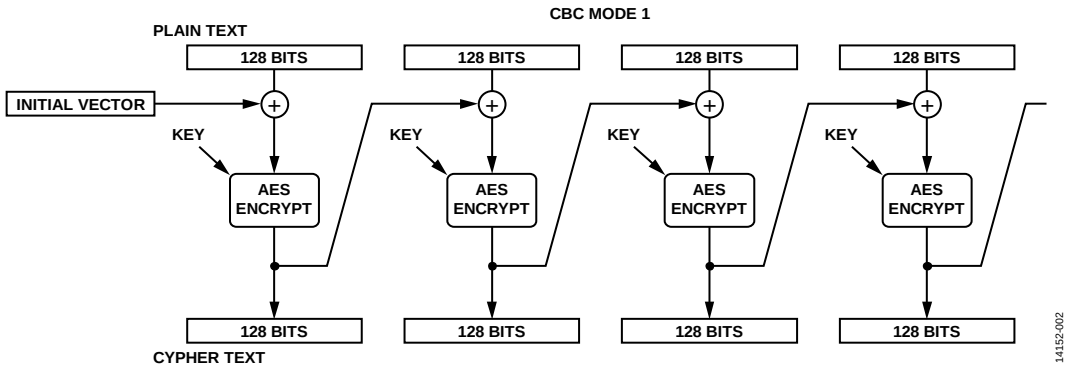

Figure 2. CBC Mode 1

TABLE OF CONTENTS

REVISION HISTORY

2/16—Revision 0: Initial Version

COMMANDS AND PACKET RANDOM ACCESS MEMORY REGISTER LOCATIONS

Table 1. Register Locations to Initialize Prior to AES Encryption or Decryption

Register Address[1]	Register Name	Description
0x001	VAR_NUM_BLOCKS	Number of 16-byte blocks to encrypt/decrypt
0x010	VAR_C_PTR	Pointer to the data to be encrypted/decrypted
0x011	VAR_W_PTR	Pointer to the 32-byte AES workspace
0x012	VAR_WINV_PTR	Pointer to the inverse key
0x013	VAR_WFOR_PTR	Pointer to the secret key
0x014	VAR_KEYSIZE	Set to 0x0C for a 128-bit key, 0x14 for a 192-bit key, or 0x1C for a 256-bit key
0x016	VAR_AES_MODE	Set to 0x00 for ECB mode or 0x01 for CBC Mode 1
0x017	VAR_ECV_PTR	Pointer to the 128-bit initialization vector used for encryption with CBC Mode 1
0x018	VAR_DCV_PTR	Pointer to the 128-bit initialization vector used for decryption with CBC Mode 1
0x019	VAR_CIPHERBUF_PTR	Pointer to the 128-bit storage location required when decrypting using CBC Mode 1

[1] These register definitions are specific to the firmware module and are not applicable to normal operation of the ADF7023.

AES configuration variables, keys, and data are stored in the packet random access memory (RAM).

The commands shown in Table 2 are necessary to perform an AES encryption, generate the inverse key, or perform an AES decryption. See the AES Procedures section for additional information regarding AES encryption and decryption procedures.

Due to the use of pointers, different key sizes, and two different modes, the implementation of AES on the ADF7023 is highly configurable. Figure 3 shows an example configuration.

Table 2. AES Commands

Command	Code	Description
CMD_AES_ENCRYPT	0xD0	Command used to encrypt a block of data
CMD_AES_DECRYPT_INIT	0xD1	Command used to generate the inverse key
CMD_AES_DECRYPT	0xD2	Command used to decrypt a block of data

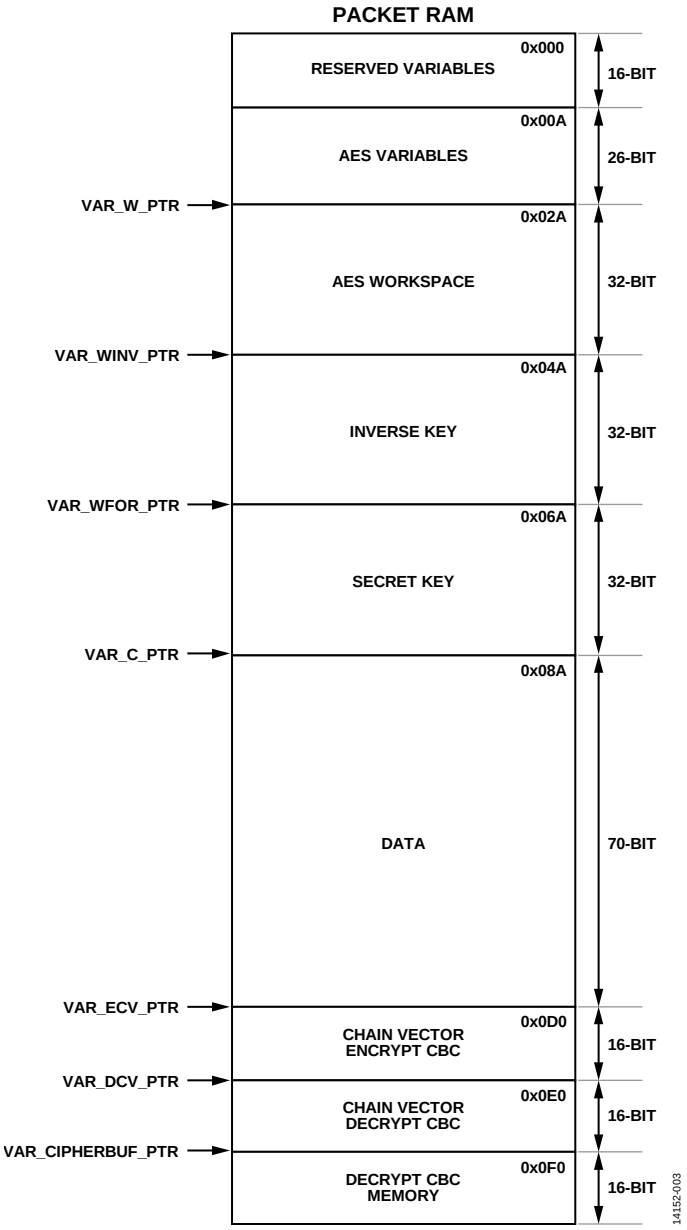

Figure 3. Example Packet RAM Memory Allocation for the AES Operation

AES PROCEDURES

WRITING THE AES FIRMWARE MODULE TO THE ADF7023

Prior to using the AES firmware module, the user must write the module to the program RAM of the ADF7023. The following steps detail how to write the AES firmware module to the program RAM:

1. Ensure that the ADF7023 is in the PHY_OFF state.
2. Issue the CMD_RAM_LOAD_INIT command (Address 0xBF).
3. Write the module to program RAM using a serial peripheral interface (SPI) memory block write (0x1E00 (firmware module); see the ADF7023 data sheet for more information on block writing).
4. Issue the CMD_RAM_LOAD_DONE command (Address 0xC7).

The firmware module is now stored in program RAM.

AES ENCRYPTION PROCEDURE

The following steps detail how to perform an AES encryption:

1. Write the start address of the AES workspace to VAR_W_PTR.
2. Write to VAR_KEYSIZE to set the size of the key.
3. Write to VAR_AES_MODE to select between ECB mode and CBC Mode 1.
4. If using CBC Mode 1 (skip this step if using ECB mode),
   a. Write the start address of the encryption initialization vector to VAR_ECV_PTR.
   b. Write the initialization vector to the location specified by VAR_ECV_PTR.
5. Write the address of the secret key to VAR_WFOR_PTR.
6. Write the secret key to the location specified by VAR_WFOR_PTR.
7. Write the number of 16-byte blocks to encrypt to VAR_NUM_BLOCKS.
8. Write the address of the data to be encrypted to VAR_C_PTR.
9. Write the data to be encrypted to the location specified by VAR_C_PTR.
10. Issue CMD_AES_ENCRYPT (0xD0). The data to be encrypted is overwritten with the encrypted data.
11. Wait for the command to finish.

Example of AES Encryption

In the following example of an AES encryption, the SPI commands are written to the ADF7023:

1. Write 0x18112A. VAR_W_PTR is set to 0x2A. The 32-byte workspace for the algorithm begins at Address 0x02A.
2. Write 0x18140C. A key size of 128 bits is selected via VAR_KEYSIZE.
3. Write 0x181600. ECB mode is selected via VAR_AES_MODE.
4. CBC Mode 1 is not being used; therefore, skip Step 4.
5. Write 0x18136A. VAR_WFOR_PTR is set to 0x6A. The secret key begins at Address 0x06A.
6. Write the secret key to the packet RAM starting at Address 0x06A.
7. Write 0x180101. VAR_NUM_BLOCKS is set to 0x01. One block of 16 bytes is then encrypted.
8. Write 0x18108A. Set VAR_C_PTR to 0x8A. The data to be encrypted begins at Address 0x08A.
9. Write the data to be encrypted to the packet RAM starting at Address 0x08A.
10. Write 0xD0. CMD_AES_ENCRYPT is issued.
11. Wait for the command to finish.

AES DECRYPTION PROCEDURE

The following steps detail how to perform an AES decryption:

1. Write the start address of the AES workspace to VAR_W_PTR.
2. Write to VAR_KEYSIZE to set the size of the key.
3. Write to VAR_AES_MODE to select between ECB mode and CBC Mode 1.
4. Write the address of the secret key to VAR_WFOR_PTR.
5. Write the secret key to the location specified by VAR_WFOR_PTR.
6. Write the address of the inverse key to VAR_WINV_PTR.
7. If using CBC Mode 1 (skip this step if using ECB mode),
   a. Write the address of the decryption initialization vector to VAR_DCV_PTR.
   b. Write the initialization vector to the location specified by VAR_DCV_PTR.
   c. Write the address of the reserved storage required when decrypting to VAR_CIPHERBUF_PTR.
8. Issue CMD_AES_DECRYPT_INIT (0xD1). This command generates and saves the inverse key.
9. Wait for the command to finish.
10. Write the number of 16-byte blocks to decrypt to VAR_NUM_BLOCKS.
11. Write the address of the data to be decrypted to VAR_C_PTR.
12. Write the data to be decrypted to the location specified by VAR_C_PTR.
13. Issue CMD_AES_DECRYPT (0xD2). The data to be decrypted is overwritten with the decrypted data.
14. Wait for the command to finish.

Example of AES Decryption

In the following example of an AES decryption, the SPI commands are written to the ADF7023:

1. Write 0x18112A. VAR_W_PTR is set to 0x2A. The 32-byte workspace for the algorithm begins at Address 0x02A.
2. Write 0x18140C. A key size of 128 bits is selected via VAR_KEYSIZE.
3. Write 0x181600. ECB mode is selected via VAR_AES_MODE.
4. Write 0x18136A. VAR_WFOR_PTR is set to 0x6A. The secret key begins at Address 0x06A.
5. Write the secret key to the packet RAM starting at Address 0x06A.
6. Write 0x18124A. VAR_WINV_PTR is set to 0x4A. The inverse key begins at Address 0x04A.
7. CBC Mode 1 is not being used; therefore, skip Step 7.

8. Write 0xD1. CMD_AES_DECRYPT_INIT is issued. This command generates and saves the inverse key starting at Address 0x04A.
9. Wait for the command to finish.
10. Write 0x180101. VAR_NUM_BLOCKS is set to 0x01. One block of 16 bytes is then decrypted.
11. Write 0x18108A. Set VAR_C_PTR to 0x8A. The data to be decrypted begins at Address 0x08A.
12. Write the data to be decrypted to the packet RAM starting at Address 0x08A.
13. Write 0xD2. CMD_AES_DECRYPT is issued.
14. Wait for the command to finish.

DETERMINING WHEN AES COMMANDS ARE COMPLETE

Use the CMD_FINISHED interrupt to determine when the CMD_AES_ENCRYPT, CMD_AES_DECRYPT_INIT, and CMD_AES_DECRYPT commands are complete. To enable this interrupt, set Bit 0 (CMD_FINISHED) of the INTERRUPT_MASK_1 register (Address 0x101). When this mask bit is set, the interrupt pin (IRQ_GP3) of the ADF7023 is asserted upon completion of any command. The interrupt is cleared by writing Logic 1 to Bit 0 of INTERRUPT_SOURCE_1 (Address 0x337). See the ADF7023 data sheet for more information on interrupt generation.

AES ENCRYPTION AND DECRYPTION TIMES

Typical AES execution times are listed in Table 3.

Table 3. AES Initialization, Encryption, and Decryption Times

Data Length (Bytes)	Key Size (Bits)	Initialize Decryption (ms)	Encryption (ms)	Decryption (ms)
16	128	1.08	1.07	1.22
	192	1.27	1.27	1.47
	256	1.47	1.46	1.69
32	128	1.08	2.13	2.42
	192	1.27	2.51	2.88
	256	1.46	2.87	3.37
48	128	1.08	3.19	3.61
	192	1.27	3.76	4.63
	256	1.46	4.3	5.05
64	128	1.08	4.24	4.82
	192	1.27	5.02	5.82
	256	1.46	5.76	6.72