ATAES132A Demo Walk-Through

Table of Contents

Overview

The demo provides a number of useful examples for MAC validation, hardware & software encryption/decryption, create key, temp key loading and usage, Key Authorizations and encrypt read and writes to the device.

Requirements

- Atmel Studio 7 (AS7)
- SAMD21 xplained Pro
- ATAES132a

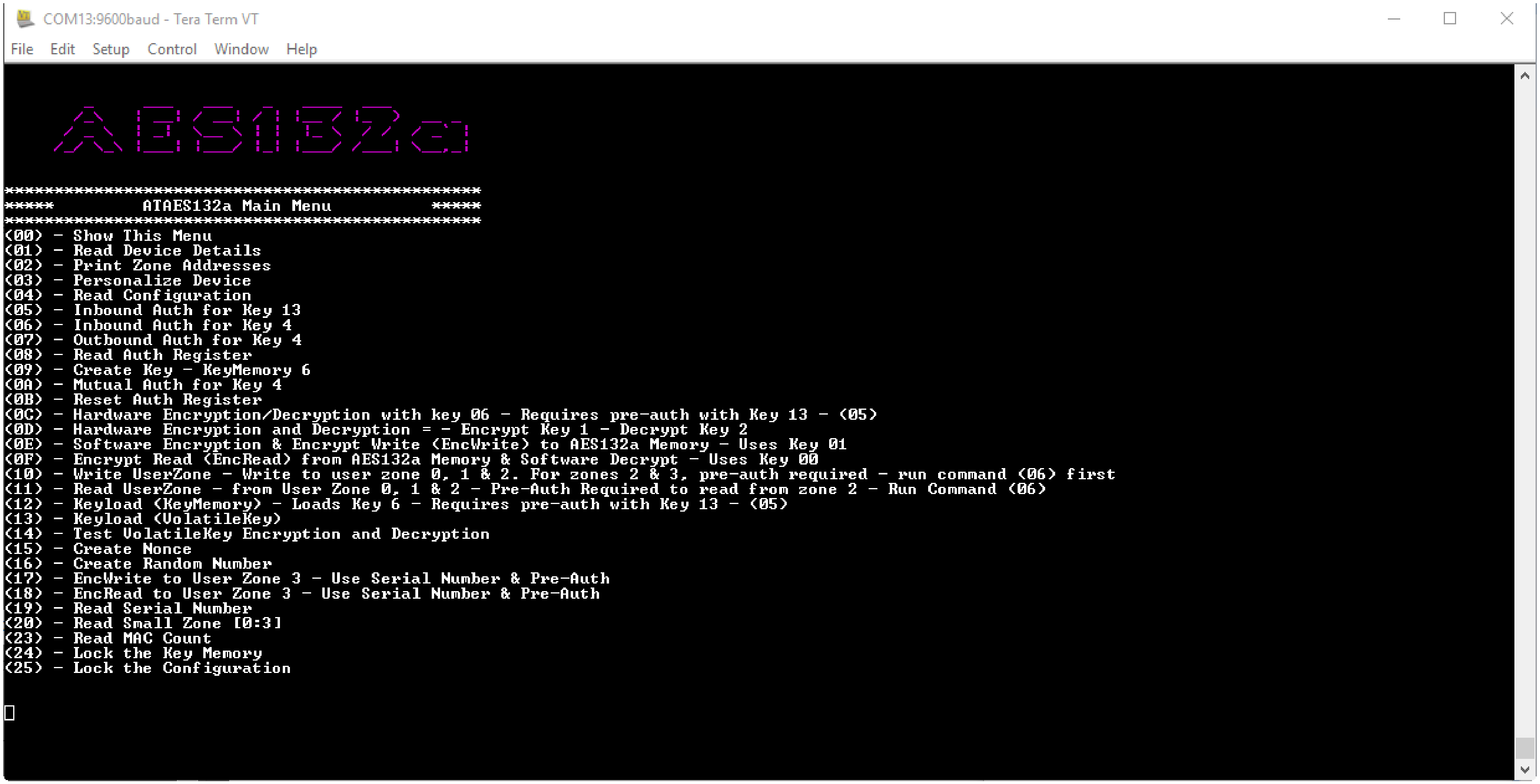
Contact

You can reach me at acmbug@gmail.com if you have any questions of comments.

Install Instructions

- Download the project
- Connect the ATAES132A to your samd21 xplained pro
- Load the example project on to the samd21
- Open up a terminal
- Type (00) for the Main Menu screen

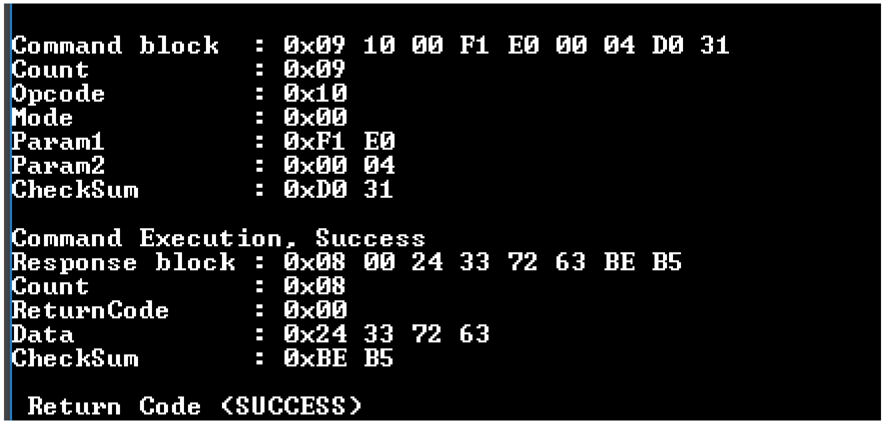
Main Menu


COM13:9600baud - Tera Term VT
File Edit Setup Control Window Help
AES132a

```
************************************************
*****         ATAES132a Main Menu          *****
************************************************
(00) - Show This Menu
(01) - Read Device Details
(02) - Print Zone Addresses
(03) - Personalize Device
(04) - Read Configuration
(05) - Inbound Auth for Key 13
(06) - Inbound Auth for Key 4
(07) - Outbound Auth for Key 4
(08) - Read Auth Register
(09) - Create Key - KeyMemory 6
(0A) - Mutual Auth for Key 4
(0B) - Reset Auth Register
(0C) - Hardware Encryption/Decryption with key 06 - Requires pre-auth with Key 13 - (05)
(0D) - Hardware Encryption and Decryption = - Encrypt Key 1 - Decrypt Key 2
(0E) - Software Encryption & Encrypt Write (EncWrite) to AES132a Memory - Uses Key 01
(0F) - Encrypt Read (EncRead) from AES132a Memory & Software Decrypt - Uses Key 00
(10) - Write UserZone - Write to user zone 0, 1 & 2. For zones 2 & 3, pre-auth required - run command (06) first
(11) - Read UserZone - from User Zone 0, 1 & 2 - Pre-Auth Required to read from zone 2 - Run Command (06)
(12) - Keyload (KeyMemory) - Loads Key 6 - Requires pre-auth with Key 13 - (05)
(13) - Keyload (VolatileKey)
(14) - Test VolatileKey Encryption and Decryption
(15) - Create Nonce
(16) - Create Random Number
(17) - EncWrite to User Zone 3 - Use Serial Number & Pre-Auth
(18) - EncRead to User Zone 3 - Use Serial Number & Pre-Auth
(19) - Read Serial Number
(20) - Read Small Zone [0:3]
(23) - Read MAC Count
(24) - Lock the Key Memory
(25) - Lock the Configuration
```

Command Block and Responses

Each command will print to the terminal the following:



```
Command block  : 0x09 10 00 F1 E0 00 04 D0 31
Count          : 0x09
Opcode         : 0x10
Mode           : 0x00
Param1         : 0xF1 E0
Param2         : 0x00 04
CheckSum       : 0xD0 31

Command Execution, Success
Response block : 0x08 00 24 33 72 63 BE B5
Count          : 0x08
ReturnCode     : 0x00
Data           : 0x24 33 72 63
CheckSum       : 0xBE B5

 Return Code (SUCCESS)
```

This allows you to easily see what is being sent to the ATAES132a, the response data and if the execution was successful or not.

Running the Demo

Before you do anything useful with the ATAES132a, you need to configure it aka Personalization

Step 1 - Personalize Device - Command (03)

Step 2 - Run Pre-Auth for Zone Write - Command (06)

Step 3 - Write to User Zone - Command (10)

Step 4 - Test the device!

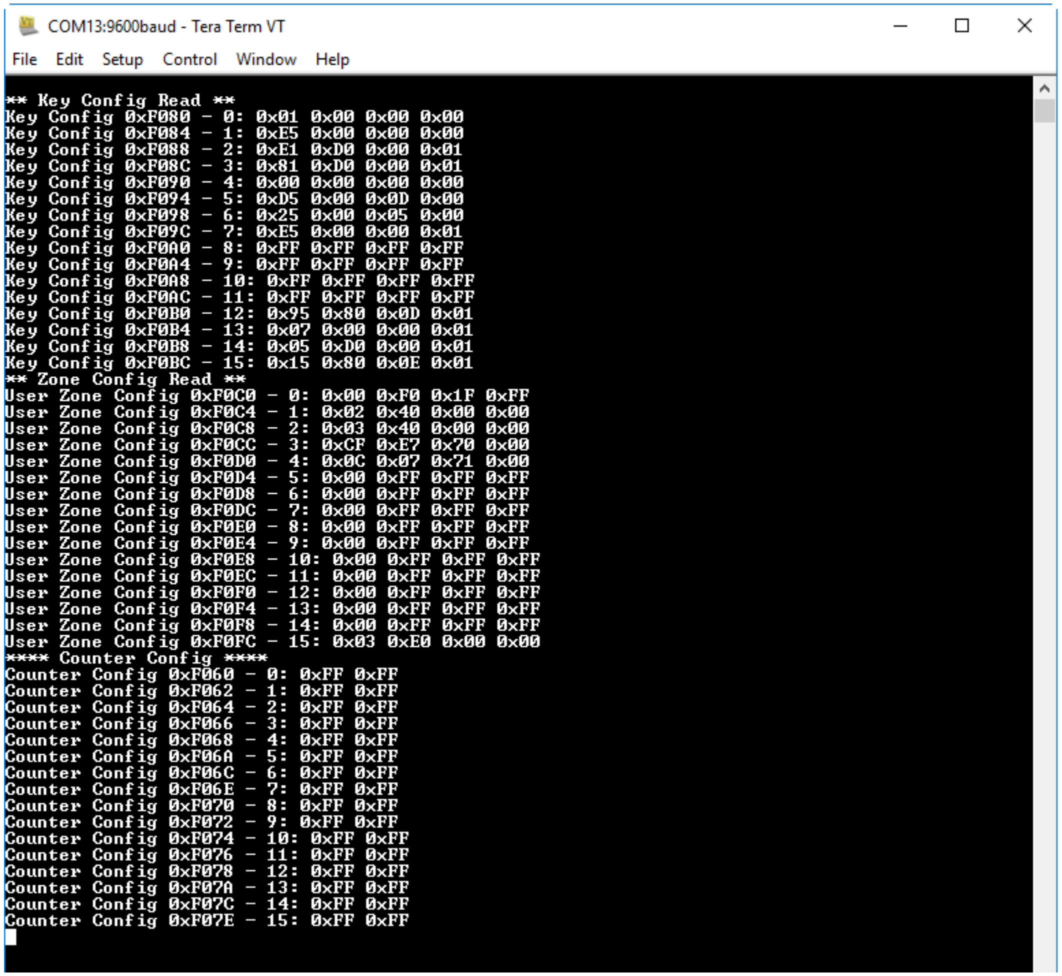
WARNING! Steps 5 & 6 are optional for testing. You do not need to lock the device to test. Keep in mind when the device is not locked; the Random Number Generator does not produce a random number. This means any keys generated on the device are going to be 16 bytes of 0xA5. You should lock the device after you are happy with the configuration.

Step 5 - Lock the Configuration - Command (25)

Step 6 - Lock the Key Memory - Command (24)

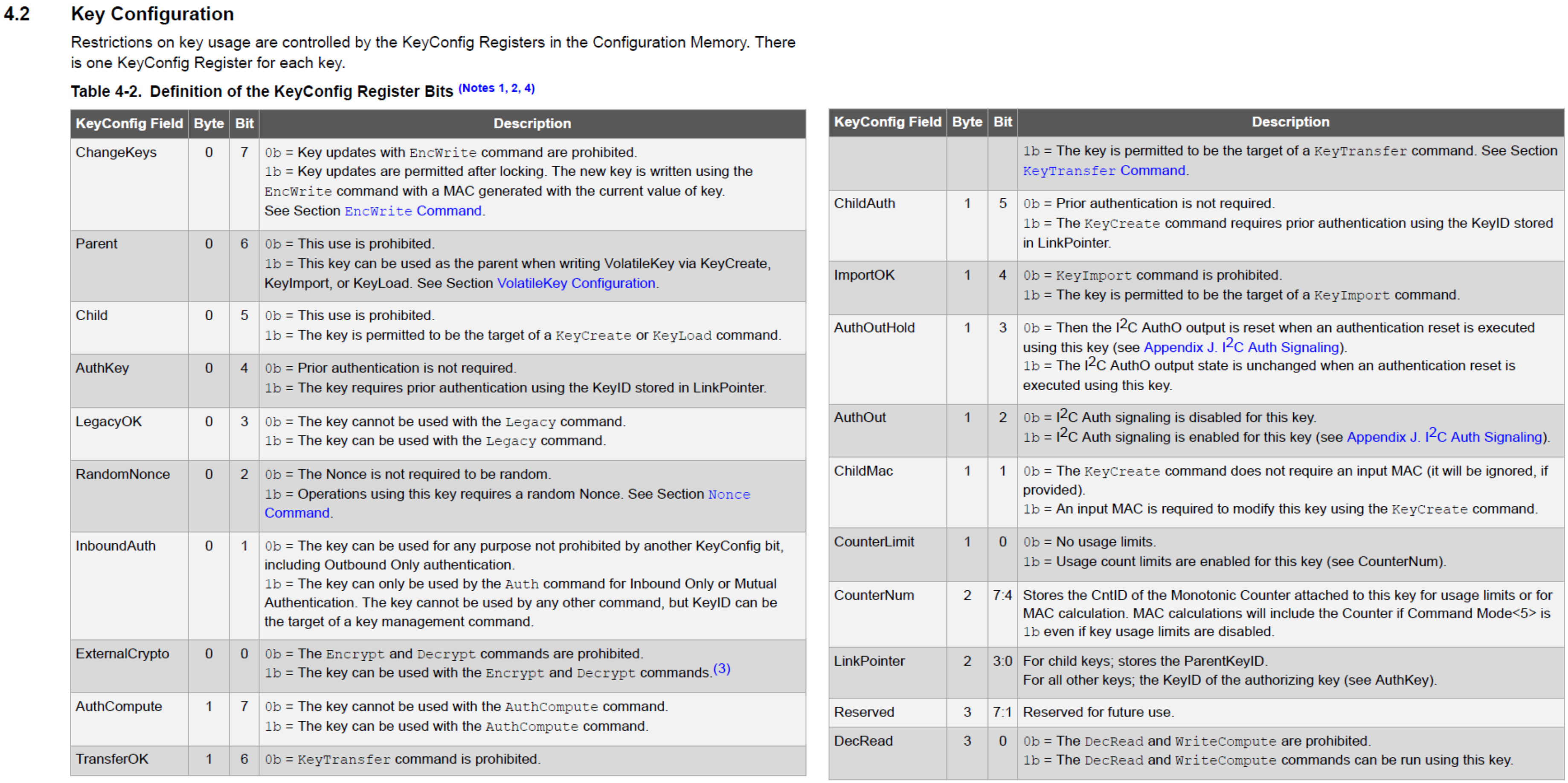
Key Configuration Registers

There are 16 key configuration registers, each are 4 bytes, 1 for each key (Figure 3). To help me configure the ATAES132a, I created a spreadsheet of the registers. You can read the devices configuration by typing the command (04) from the main menu. This will also print the Zone and Counter configuration.


COM13:9600baud - Tera Term VT
File Edit Setup Control Window Help

```
** Key Config Read **
Key Config 0xF080 - 0: 0x01 0x00 0x00 0x00
Key Config 0xF084 - 1: 0xE5 0x00 0x00 0x00
Key Config 0xF088 - 2: 0xE1 0xD0 0x00 0x01
Key Config 0xF08C - 3: 0x81 0xD0 0x00 0x01
Key Config 0xF090 - 4: 0x00 0x00 0x00 0x00
Key Config 0xF094 - 5: 0xD5 0x00 0x0D 0x00
Key Config 0xF098 - 6: 0x25 0x00 0x05 0x00
Key Config 0xF09C - 7: 0xE5 0x00 0x00 0x01
Key Config 0xF0A0 - 8: 0xFF 0xFF 0xFF 0xFF
Key Config 0xF0A4 - 9: 0xFF 0xFF 0xFF 0xFF
Key Config 0xF0A8 - 10: 0xFF 0xFF 0xFF 0xFF
Key Config 0xF0AC - 11: 0xFF 0xFF 0xFF 0xFF
Key Config 0xF0B0 - 12: 0x95 0x80 0x0D 0x01
Key Config 0xF0B4 - 13: 0x07 0x00 0x00 0x01
Key Config 0xF0B8 - 14: 0x05 0xD0 0x00 0x01
Key Config 0xF0BC - 15: 0x15 0x80 0x0E 0x01
** Zone Config Read **
User Zone Config 0xF0C0 - 0: 0x00 0xF0 0x1F 0xFF
User Zone Config 0xF0C4 - 1: 0x02 0x40 0x00 0x00
User Zone Config 0xF0C8 - 2: 0x03 0x40 0x00 0x00
User Zone Config 0xF0CC - 3: 0xCF 0xE7 0x70 0x00
User Zone Config 0xF0D0 - 4: 0x0C 0x07 0x71 0x00
User Zone Config 0xF0D4 - 5: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0D8 - 6: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0DC - 7: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0E0 - 8: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0E4 - 9: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0E8 - 10: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0EC - 11: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0F0 - 12: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0F4 - 13: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0F8 - 14: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0FC - 15: 0x03 0xE0 0x00 0x00
**** Counter Config ****
Counter Config 0xF060 - 0: 0xFF 0xFF
Counter Config 0xF062 - 1: 0xFF 0xFF
Counter Config 0xF064 - 2: 0xFF 0xFF
Counter Config 0xF066 - 3: 0xFF 0xFF
Counter Config 0xF068 - 4: 0xFF 0xFF
Counter Config 0xF06A - 5: 0xFF 0xFF
Counter Config 0xF06C - 6: 0xFF 0xFF
Counter Config 0xF06E - 7: 0xFF 0xFF
Counter Config 0xF070 - 8: 0xFF 0xFF
Counter Config 0xF072 - 9: 0xFF 0xFF
Counter Config 0xF074 - 10: 0xFF 0xFF
Counter Config 0xF076 - 11: 0xFF 0xFF
Counter Config 0xF078 - 12: 0xFF 0xFF
Counter Config 0xF07A - 13: 0xFF 0xFF
Counter Config 0xF07C - 14: 0xFF 0xFF
Counter Config 0xF07E - 15: 0xFF 0xFF
```

Figure 2 – Key Configurations



4.2 Key Configuration

Restrictions on key usage are controlled by the KeyConfig Registers in the Configuration Memory. There is one KeyConfig Register for each key.

Table 4-2. Definition of the KeyConfig Register Bits (Notes 1, 2, 4)

KeyConfig Field	Byte	Bit	Description
ChangeKeys	0	7	`0b` = Key updates with `EncWrite` command are prohibited. `1b` = Key updates are permitted after locking. The new key is written using the `EncWrite` command with a MAC generated with the current value of key. See Section `EncWrite` Command.
Parent	0	6	`0b` = This use is prohibited. `1b` = This key can be used as the parent when writing VolatileKey via KeyCreate, KeyImport, or KeyLoad. See Section VolatileKey Configuration.
Child	0	5	`0b` = This use is prohibited. `1b` = The key is permitted to be the target of a `KeyCreate` or `KeyLoad` command.
AuthKey	0	4	`0b` = Prior authentication is not required. `1b` = The key requires prior authentication using the KeyID stored in LinkPointer.
LegacyOK	0	3	`0b` = The key cannot be used with the `Legacy` command. `1b` = The key can be used with the `Legacy` command.
RandomNonce	0	2	`0b` = The Nonce is not required to be random. `1b` = Operations using this key requires a random Nonce. See Section `Nonce` Command.
InboundAuth	0	1	`0b` = The key can be used for any purpose not prohibited by another KeyConfig bit, including Outbound Only authentication. `1b` = The key can only be used by the `Auth` command for Inbound Only or Mutual Authentication. The key cannot be used by any other command, but KeyID can be the target of a key management command.
ExternalCrypto	0	0	`0b` = The `Encrypt` and `Decrypt` commands are prohibited. `1b` = The key can be used with the `Encrypt` and `Decrypt` commands.(3)
AuthCompute	1	7	`0b` = The key cannot be used with the `AuthCompute` command. `1b` = The key can be used with the `AuthCompute` command.
TransferOK	1	6	`0b` = `KeyTransfer` command is prohibited. `1b` = The key is permitted to be the target of a `KeyTransfer` command. See Section `KeyTransfer` Command.
ChildAuth	1	5	`0b` = Prior authentication is not required. `1b` = The `KeyCreate` command requires prior authentication using the KeyID stored in LinkPointer.
ImportOK	1	4	`0b` = `KeyImport` command is prohibited. `1b` = The key is permitted to be the target of a `KeyImport` command.
AuthOutHold	1	3	`0b` = Then the I^{2}C AuthO output is reset when an authentication reset is executed using this key (see Appendix J. I^{2}C Auth Signaling). `1b` = The I^{2}C AuthO output state is unchanged when an authentication reset is executed using this key.
AuthOut	1	2	`0b` = I^{2}C Auth signaling is disabled for this key. `1b` = I^{2}C Auth signaling is enabled for this key (see Appendix J. I^{2}C Auth Signaling).
ChildMac	1	1	`0b` = The `KeyCreate` command does not require an input MAC (it will be ignored, if provided). `1b` = An input MAC is required to modify this key using the `KeyCreate` command.
CounterLimit	1	0	`0b` = No usage limits. `1b` = Usage count limits are enabled for this key (see CounterNum).
CounterNum	2	7:4	Stores the CntID of the Monotonic Counter attached to this key for usage limits or for MAC calculation. MAC calculations will include the Counter if Command Mode<5> is `1b` even if key usage limits are disabled.
LinkPointer	2	3:0	For child keys; stores the ParentKeyID. For all other keys; the KeyID of the authorizing key (see AuthKey).
Reserved	3	7:1	Reserved for future use.
DecRead	3	0	`0b` = The `DecRead` and `WriteCompute` are prohibited. `1b` = The `DecRead` and `WriteCompute` commands can be run using this key.

Figure 3 - KeyConfig

Menu Commands

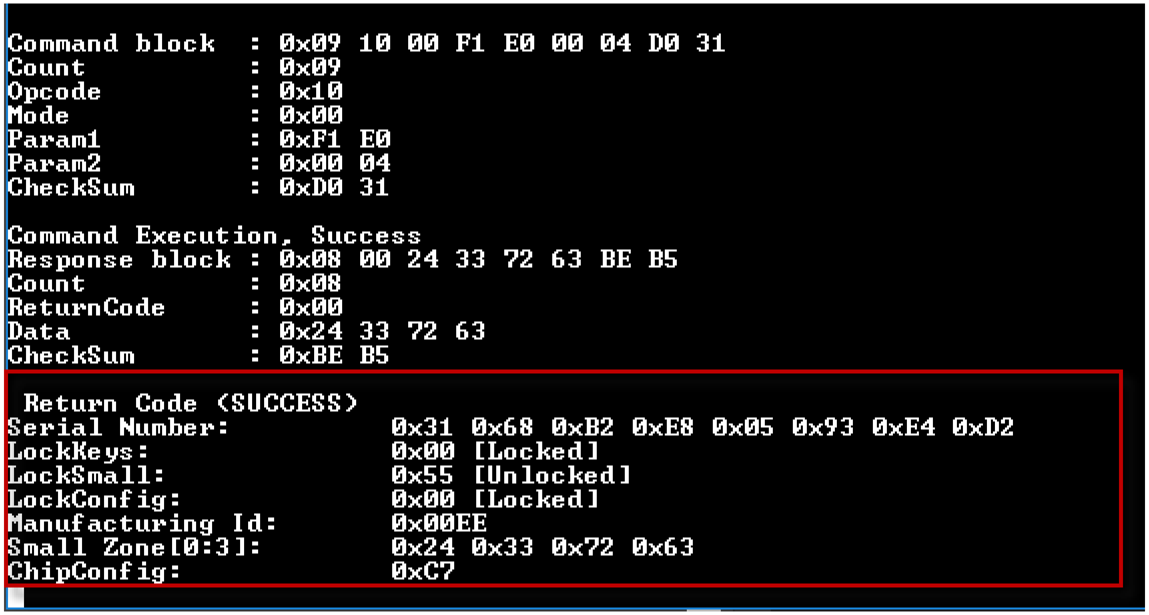
The demo has a number of useful examples. Some examples may require a number of steps, which is indicated in the instructions.

Command (00) - Main Menu

Prints the Command Menu.

Command (01) - Print Device Details

Reads the Device Details.



```
Command block   : 0x09 10 00 F1 E0 00 04 D0 31
Count           : 0x09
Opcode          : 0x10
Mode            : 0x00
Param1          : 0xF1 E0
Param2          : 0x00 04
CheckSum        : 0xD0 31

Command Execution, Success
Response block : 0x08 00 24 33 72 63 BE B5
Count           : 0x08
ReturnCode      : 0x00
Data            : 0x24 33 72 63
CheckSum        : 0xBE B5

 Return Code (SUCCESS)
Serial Number:          0x31 0x68 0xB2 0xE8 0x05 0x93 0xE4 0xD2
LockKeys:               0x00 [Locked]
LockSmall:              0x55 [Unlocked]
LockConfig:             0x00 [Locked]
Manufacturing Id:       0x00EE
Small Zone[0:3]:        0x24 0x33 0x72 0x63
ChipConfig:             0xC7
```

Command (02) – Print Zone Addresses

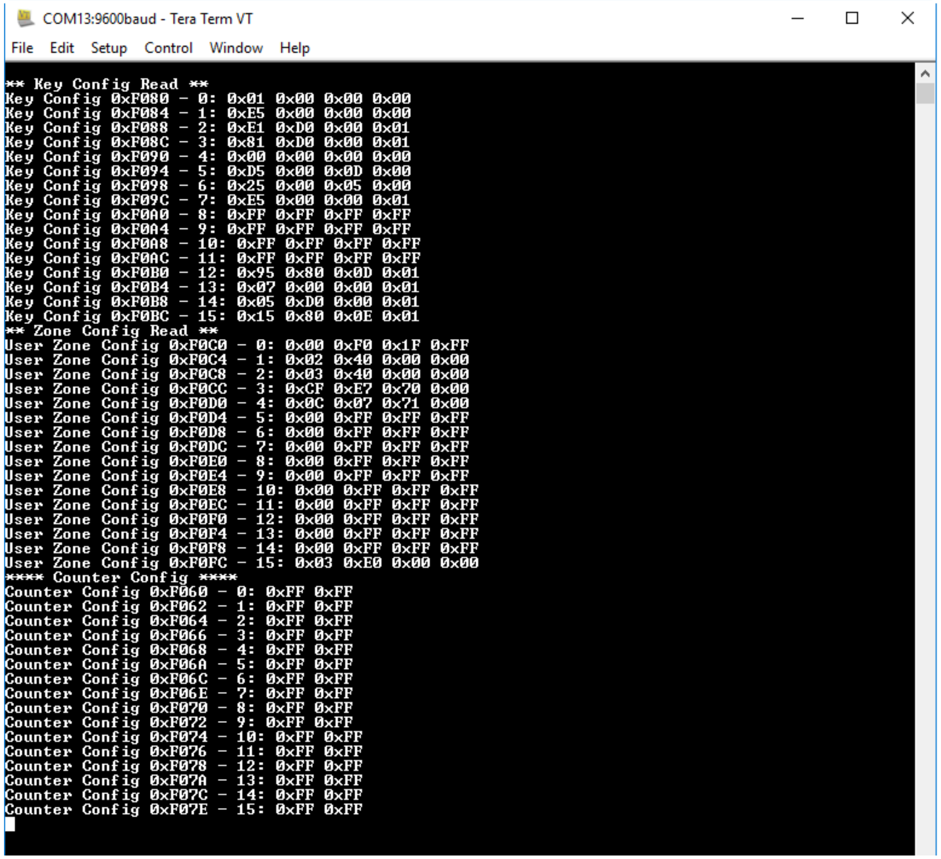
Prints the Zone Addresses. Useful for when you need to read or write to a zone.

Command (03) – Configure Device

Personalize the device. This is perhaps the most important step for this demo and should be run before locking the device. This step configures the Key Memory, Zones , writes all the keys to the device, write to the zones 0,1,2 & small zone.

Command (04) – Print Configuration

Reads the configuration for the Key Memory Config, Zone Config and Counter Config


COM13:9600baud - Tera Term VT
File Edit Setup Control Window Help

```
** Key Config Read **
Key Config 0xF080 - 0: 0x01 0x00 0x00 0x00
Key Config 0xF084 - 1: 0xE5 0x00 0x00 0x00
Key Config 0xF088 - 2: 0xE1 0xD0 0x00 0x01
Key Config 0xF08C - 3: 0x81 0xD0 0x00 0x01
Key Config 0xF090 - 4: 0x00 0x00 0x00 0x00
Key Config 0xF094 - 5: 0xD5 0x00 0x0D 0x00
Key Config 0xF098 - 6: 0x25 0x00 0x05 0x00
Key Config 0xF09C - 7: 0xE5 0x00 0x00 0x01
Key Config 0xF0A0 - 8: 0xFF 0xFF 0xFF 0xFF
Key Config 0xF0A4 - 9: 0xFF 0xFF 0xFF 0xFF
Key Config 0xF0A8 - 10: 0xFF 0xFF 0xFF 0xFF
Key Config 0xF0AC - 11: 0xFF 0xFF 0xFF 0xFF
Key Config 0xF0B0 - 12: 0x95 0x80 0x0D 0x01
Key Config 0xF0B4 - 13: 0x07 0x00 0x00 0x01
Key Config 0xF0B8 - 14: 0x05 0xD0 0x00 0x01
Key Config 0xF0BC - 15: 0x15 0x80 0x0E 0x01
** Zone Config Read **
User Zone Config 0xF0C0 - 0: 0x00 0xF0 0x1F 0xFF
User Zone Config 0xF0C4 - 1: 0x02 0x40 0x00 0x00
User Zone Config 0xF0C8 - 2: 0x03 0x40 0x00 0x00
User Zone Config 0xF0CC - 3: 0xCF 0xE7 0x70 0x00
User Zone Config 0xF0D0 - 4: 0x0C 0x07 0x71 0x00
User Zone Config 0xF0D4 - 5: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0D8 - 6: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0DC - 7: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0E0 - 8: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0E4 - 9: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0E8 - 10: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0EC - 11: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0F0 - 12: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0F4 - 13: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0F8 - 14: 0x00 0xFF 0xFF 0xFF
User Zone Config 0xF0FC - 15: 0x03 0xE0 0x00 0x00
**** Counter Config ****
Counter Config 0xF060 - 0: 0xFF 0xFF
Counter Config 0xF062 - 1: 0xFF 0xFF
Counter Config 0xF064 - 2: 0xFF 0xFF
Counter Config 0xF066 - 3: 0xFF 0xFF
Counter Config 0xF068 - 4: 0xFF 0xFF
Counter Config 0xF06A - 5: 0xFF 0xFF
Counter Config 0xF06C - 6: 0xFF 0xFF
Counter Config 0xF06E - 7: 0xFF 0xFF
Counter Config 0xF070 - 8: 0xFF 0xFF
Counter Config 0xF072 - 9: 0xFF 0xFF
Counter Config 0xF074 - 10: 0xFF 0xFF
Counter Config 0xF076 - 11: 0xFF 0xFF
Counter Config 0xF078 - 12: 0xFF 0xFF
Counter Config 0xF07A - 13: 0xFF 0xFF
Counter Config 0xF07C - 14: 0xFF 0xFF
Counter Config 0xF07E - 15: 0xFF 0xFF
```

Command (05) – Inbound Auth

Sets the Inbound Auth Register to Key 13.

Command (06) - Inbound Auth

Sets the Inbound Auth Register to Key 4.

Command (07) - Outbound Auth

Sets the Outbound Auth Register to Key 4.

Command (08) – Read Auth Register

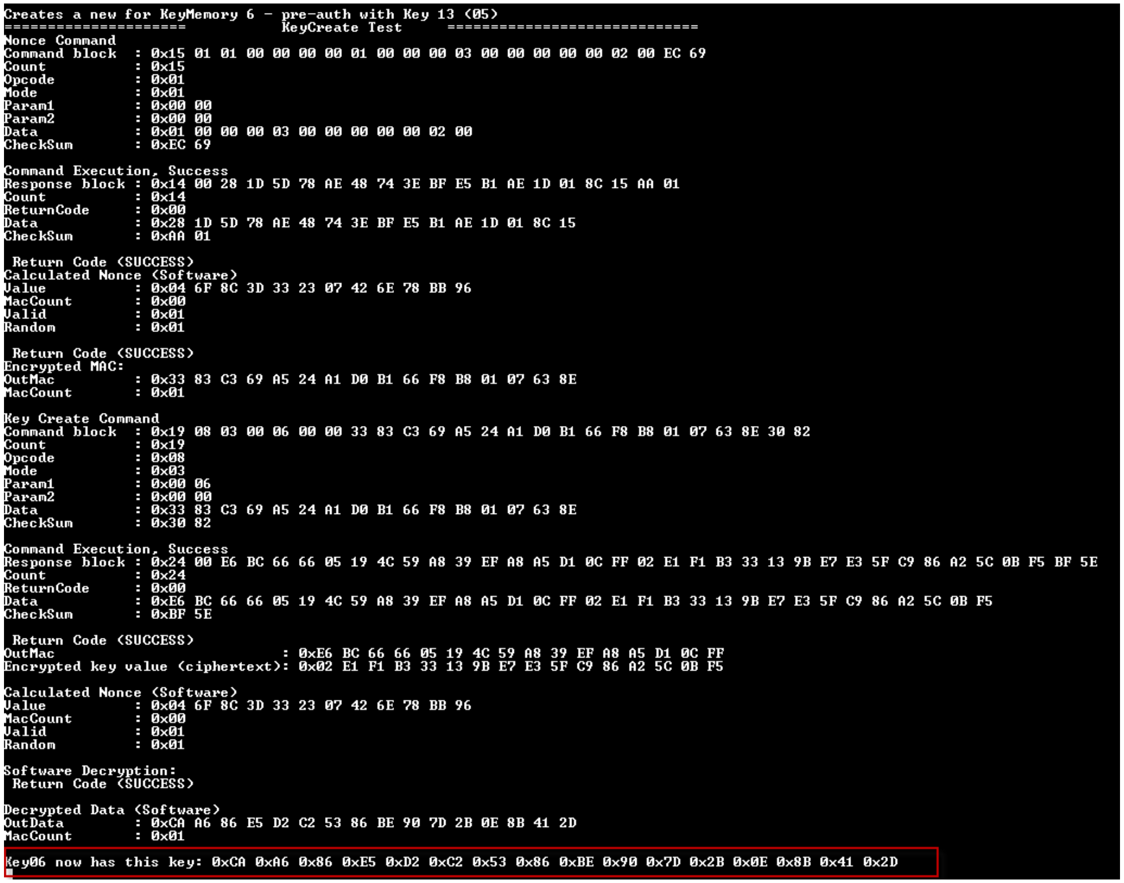
Reads the Auth Register. This will tell you what key was last used for Authorization.

Command (09) - KeyCreate

Creates a new for Key for Key Memory 06. Pre-auth is required with Key 13.

Step 1 – Run Command (05) – Pre-Auth with key 13

Step 2 – Run Command (09) – Creates a key and loads the new key in to Key Memory 06. The new key is returned to the terminal window.

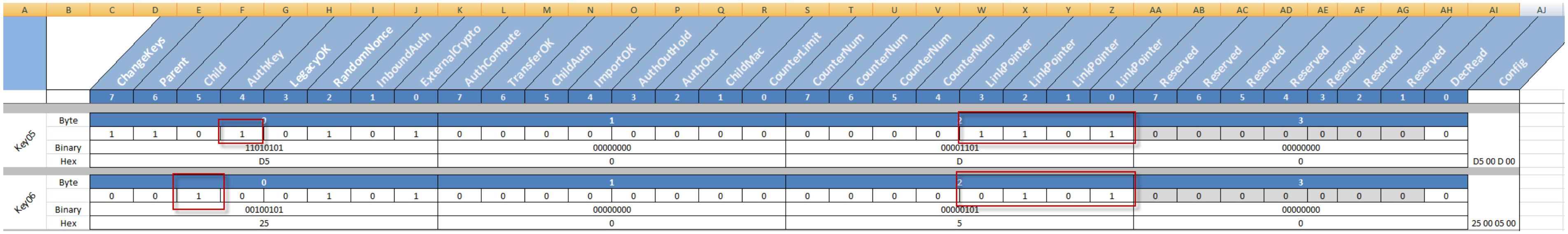


```
Creates a new for KeyMemory 6 - pre-auth with Key 13 (05)
=====================           KeyCreate Test      ==============================
Nonce Command
Command block  : 0x15 01 01 00 00 00 00 01 00 00 00 03 00 00 00 00 00 02 00 EC 69
Count          : 0x15
Opcode         : 0x01
Mode           : 0x01
Param1         : 0x00 00
Param2         : 0x00 00
Data           : 0x01 00 00 00 03 00 00 00 00 00 02 00
CheckSum       : 0xEC 69

Command Execution, Success
Response block : 0x14 00 28 1D 5D 78 AE 48 74 3E BF E5 B1 AE 1D 01 8C 15 AA 01
Count          : 0x14
ReturnCode     : 0x00
Data           : 0x28 1D 5D 78 AE 48 74 3E BF E5 B1 AE 1D 01 8C 15
CheckSum       : 0xAA 01

 Return Code (SUCCESS)
Calculated Nonce (Software)
Value          : 0x04 6F 8C 3D 33 23 07 42 6E 78 BB 96
MacCount       : 0x00
Valid          : 0x01
Random         : 0x01

 Return Code (SUCCESS)
Encrypted MAC:
OutMac         : 0x33 83 C3 69 A5 24 A1 D0 B1 66 F8 B8 01 07 63 8E
MacCount       : 0x01

Key Create Command
Command block  : 0x19 08 03 00 06 00 00 33 83 C3 69 A5 24 A1 D0 B1 66 F8 B8 01 07 63 8E 30 82
Count          : 0x19
Opcode         : 0x08
Mode           : 0x03
Param1         : 0x00 06
Param2         : 0x00 00
Data           : 0x33 83 C3 69 A5 24 A1 D0 B1 66 F8 B8 01 07 63 8E
CheckSum       : 0x30 82

Command Execution, Success
Response block : 0x24 00 E6 BC 66 66 05 19 4C 59 A8 39 EF A8 A5 D1 0C FF 02 E1 F1 B3 33 13 9B E7 E3 5F C9 86 A2 5C 0B F5 BF 5E
Count          : 0x24
ReturnCode     : 0x00
Data           : 0xE6 BC 66 66 05 19 4C 59 A8 39 EF A8 A5 D1 0C FF 02 E1 F1 B3 33 13 9B E7 E3 5F C9 86 A2 5C 0B F5
CheckSum       : 0xBF 5E

 Return Code (SUCCESS)
OutMac                          : 0xE6 BC 66 66 05 19 4C 59 A8 39 EF A8 A5 D1 0C FF
Encrypted key value (ciphertext): 0x02 E1 F1 B3 33 13 9B E7 E3 5F C9 86 A2 5C 0B F5

Calculated Nonce (Software)
Value          : 0x04 6F 8C 3D 33 23 07 42 6E 78 BB 96
MacCount       : 0x00
Valid          : 0x01
Random         : 0x01

Software Decryption:
 Return Code (SUCCESS)

Decrypted Data (Software)
OutData        : 0xCA A6 86 E5 D2 C2 53 86 BE 90 7D 2B 0E 8B 41 2D
MacCount       : 0x01

Key06 now has this key: 0xCA 0xA6 0x86 0xE5 0xD2 0xC2 0x53 0x86 0xBE 0x90 0x7D 0x2B 0x0E 0x8B 0x41 0x2D
```

The command will also test the new key. Keep in mind, that after running this command, if you run Command (0C), you will get a Return Code (MAC_ERROR). This is because we have changed the keys for Key Memory 06. You can return the original key with Command (12) and then run Command (0C).

For this command we need to take a look at KeyConfig06 configuration:

- The Nonce must be random for computing which is indicated by KeyConfig[RandomNonce] bit
- The key can be used for External Crypto Functions which is indicated by KeyConfig[ExternalCrypto] bit
- Key 6 is a child key, indicated by KeyConfig[Child] bit – which means it's the target of a KeyCreate function. When this bit is set, and the KeyCreate command is run, the devices looks at the 4 bits in the KeyConfig[LinkPointer] to know what key is used in encryption and decryption. This is known as the Parent key. In this configuration, Key05 is the parent key. Please note, when a key is created, the KeyCreate command generates a 16-byte random number, and stores it in the Key Memory, in this case key06. The newly generated key is then encrypted with the parent key (key05) and returned to the Host along with a MAC. The Key is then decrypted in the software and returned to the terminal.

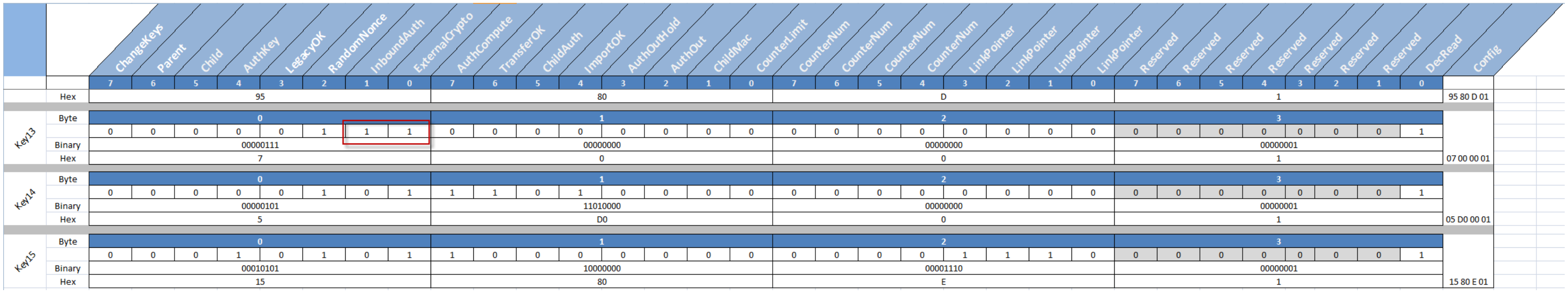


A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ
		ChangeKeys	Parent	Child	AuthKey	LegacyOK	RandomNonce	InboundAuth	ExternalCrypto	AuthCompute	TransferOK	ChildAuth	ImportOK	AuthOutHold	AuthOut	ChildMac	CounterLimit	CounterNum	CounterNum	CounterNum	CounterNum	LinkPointer	LinkPointer	LinkPointer	LinkPointer	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	DecRead	Config	
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0		
Key05	Byte	0								1								2								3									
		1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0		
	Binary	11010101								00000000								00001101								00000000									
	Hex	D5								0								D								0								D5 00 D 00	
Key06	Byte	0								1								2								3									
		0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0		
	Binary	00100101								00000000								00000101								00000000									
	Hex	25								0								5								0								25 00 05 00	

Since key05 is the parent key and will be used for the encryption/decryption, we need look at its configuration.

- Key05 can be used as an AuthKey as indicated by the KeyConfig[AuthKey] bit. When this bit is set, the device looks at the LinkPointer bits in KeyConfig[LinkPointer] to see what key the Auth Command must be run against. For Key05, the Auth Command must be run against Key13. So now we must make sure Key13 is allowed to be used as target of the Auth Command.

There are 2 important bits that need to be set. The KeyConfig[InboundAuth] and KeyConfig[ExternalCrypto] bits. This permits key13 to be used as the target of an inbound Auth.



		ChangeKeys	Parent	Child	AuthKey	LegacyOK	RandomNonce	InboundAuth	ExternalCrypto	AuthCompute	TransferOK	ChildAuth	ImportOK	AuthOutHold	AuthOut	ChildMac	CounterLimit	CounterNum	CounterNum	CounterNum	CounterNum	LinkPointer	LinkPointer	LinkPointer	LinkPointer	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	DecRead	Config
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
	Hex	95								80								D								1								95 80 D 01
Key13	Byte	0								1								2								3								
		0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
	Binary	00000111								00000000								00000000								00000001								
	Hex	7								0								0								1								07 00 00 01
Key14	Byte	0								1								2								3								
		0	0	0	0	0	1	0	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
	Binary	00000101								11010000								00000000								00000001								
	Hex	5								D0								0								1								05 D0 00 01
Key15	Byte	0								1								2								3								
		0	0	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	1	
	Binary	00010101								10000000								00001110								00000001								
	Hex	15								80								E								1								15 80 E 01

Command (0A) – Mutual Auth

A Mutual Auth is performed on Key 4.

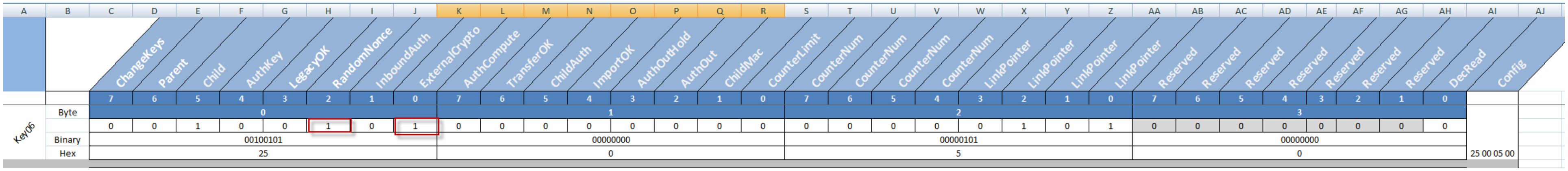
Command (0B) – Reset Auth Register

Resets Auth Register.

Command (0C) – Encryption/Decryption

Hardware Encryption/Decryption with key 06.

The ExternCrypto bit is set and a Random Nonce is required.

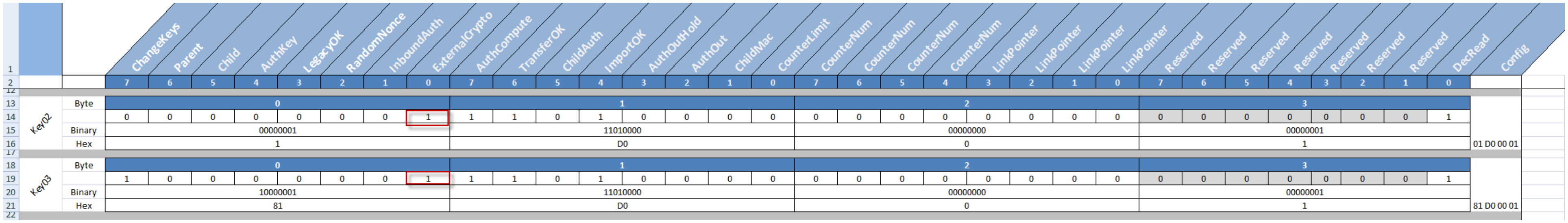


A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
		ChangeKeys	Parent	Child	AuthKey	LegacyOK	RandomNonce	InboundAuth	ExternalCrypto	AuthCompute	TransferOK	ChildAuth	ImportOK	AuthOutHold	AuthOut	ChildMac	CounterLimit
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Key06	Byte	0								1							
		0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0
	Binary	00100101								00000000							
	Hex	25								0							

A	B	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ
		CounterNum	CounterNum	CounterNum	CounterNum	LinkPointer	LinkPointer	LinkPointer	LinkPointer	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	DecRead	Config	
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0		
Key06	Byte	2								3									
		0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0		
	Binary	00000101								00000000									
	Hex	5								0								25 00 05 00	

Command (0D) - Hardware Encryption and Decryption

Hardware Encryption and Decryption with Encryption using Key 1 and decryption using Decrypt Key 2. Both Key 1 and 2 are identical and demonstrates encrypting with one key and decryption with another. To do this the ExternalCrypto bits must be set for each key



1			ChangeKeys	Parent	Child	AuthKey	LegacyOK	RandomNonce	InboundAuth	ExternalCrypto	AuthCompute	TransferOK	ChildAuth	ImportOK	AuthOutHold	AuthOut	ChildMac	CounterLimit
2			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
12																		
13	Key02	Byte	0								1							
14			0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0
15		Binary	00000001								11010000							
16		Hex	1								D0							
17																		
18	Key03	Byte	0								1							
19			1	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0
20		Binary	10000001								11010000							
21		Hex	81								D0							
22																		

1			CounterNum	CounterNum	CounterNum	CounterNum	LinkPointer	LinkPointer	LinkPointer	LinkPointer	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	DecRead	Config
2			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
12																			
13	Key02	Byte	2								3								
14			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
15		Binary	00000000								00000001								
16		Hex	0								1								01 D0 00 01
17																			
18	Key03	Byte	2								3								
19			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
20		Binary	00000000								00000001								
21		Hex	0								1								81 D0 00 01
22																			

Command (0E) - Software Encryption & Encrypt Write

Software Encryption & Encrypt Write (EncWrite) to AES132a Memory - Uses Key 01. See Command (0D) for key configuration.

Command (0F) - EncRead

Encrypt Read (EncRead) from AES132a Memory & Software Decrypt - Uses Key 00. See Command (0D) for key configuration

Command (10) – Write to Zone

Writes to UserZone user zones 0, 1 & 2. For zones 2 & 3, pre-auth required with Key04 is required

Step 1 – Run Command (06) – Pre-Auth for key 4

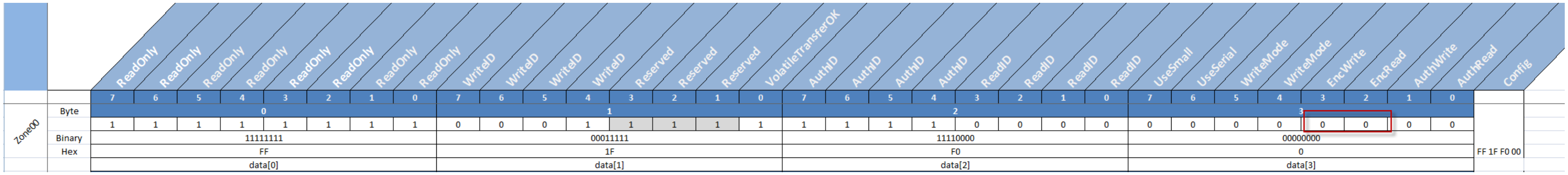
Step 2 – Run Command (10) – Writes to zones, 0, 1 & 2.

For this example, we need to see how the Zone 00 is configured. Reference Table 14-11. Definition of the ZoneConfig Register Bits of the Datasheet for more information

User Zone 00

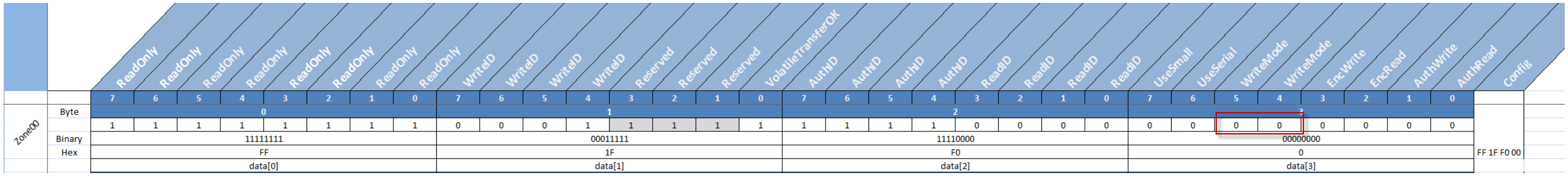
User Zone 00 configuration (Address 0xF0C0): 0x00 0xF0 0x1F 0xFF

The ZoneConfig[EncRead] & ZoneConfig[EncWrite] are not set, so this zone can be written with Plain Text.



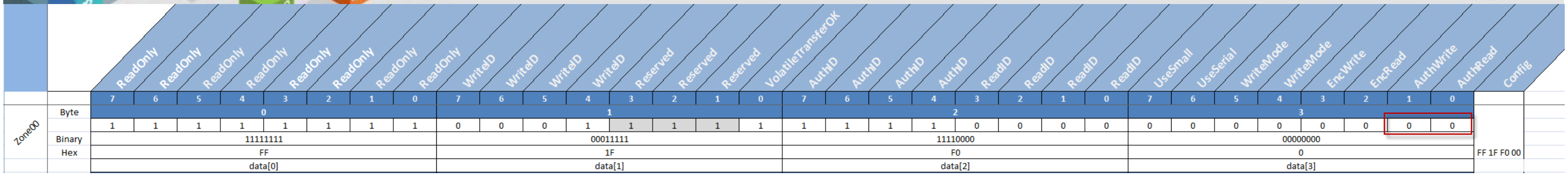
		ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	WriteID	WriteID	WriteID	WriteID	Reserved	Reserved	Reserved	VolatileTransferOK	AuthID	AuthID	AuthID	AuthID	ReadID	ReadID	ReadID	ReadID	UseSmall	UseSerial	WriteMode	WriteMode	EncWrite	EncRead	AuthWrite	AuthRead	Config
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
Zone00	Byte	0								1								2								3								
		1	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	
	Binary	11111111								00011111								11110000								00000000								
	Hex	FF								1F								F0								0								FF 1F F0 00
		data[0]								data[1]								data[2]								data[3]								

ZoneConfig[WriteMode] is set to permanently allow Read/Writes



		ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	WriteID	WriteID	WriteID	WriteID	Reserved	Reserved	Reserved	VolatileTransferOK	AuthID	AuthID	AuthID	AuthID	ReadID	ReadID	ReadID	ReadID	UseSmall	UseSerial	WriteMode	WriteMode	EncWrite	EncRead	AuthWrite	AuthRead	Config
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
Zone00	Byte	0								1								2								3								
		1	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	
	Binary	11111111								00011111								11110000								00000000								
	Hex	FF								1F								F0								0								FF 1F F0 00
		data[0]								data[1]								data[2]								data[3]								

ZoneConfig[AuthWrite] and ZoneConfig[AuthRead] are not set, so no authorization is required to read or write to zone 00.



		ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	WriteID	WriteID	WriteID	WriteID	Reserved	Reserved	Reserved	VolatileTransferOK	AuthID	AuthID	AuthID	AuthID	ReadID	ReadID	ReadID	ReadID	UseSmall	UseSerial	WriteMode	WriteMode	EncWrite	EncRead	AuthWrite	AuthRead	Config
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
Zone00	Byte	0								1								2								3								
		1	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	
	Binary	11111111								00011111								11110000								00000000								
	Hex	FF								1F								F0								0								FF 1F F0 00
		data[0]								data[1]								data[2]								data[3]								

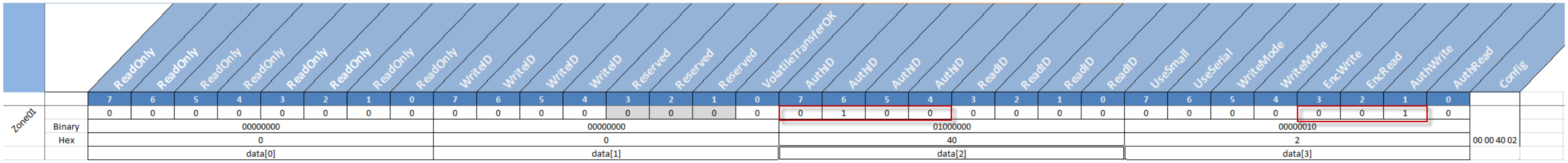
User Zone 01

User Zone Config 01 configuration (Address 0xF0C4): 0x02 0x40 0x00 0x00

The ZoneConfig[EncRead] & ZoneConfig[EncWrite] are not set, so this zone can be written with Plain Text.

The ZoneConfig[AuthWrite] bit is set, so we must run the Auth Command for key04 before writing to zone 01.

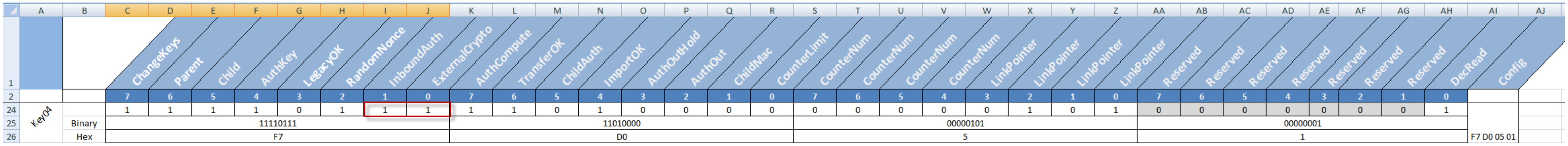
The ZoneConfig[AuthId] bits indicates what key must be used for Pre-Auth (Key 04).



		ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	WriteID	WriteID	WriteID	WriteID	Reserved	Reserved	Reserved	VolatileTransferOK	AuthID	AuthID	AuthID	AuthID	ReadID	ReadID	ReadID	ReadID	UseSmall	UseSerial	WriteMode	WriteMode	EncWrite	EncRead	AuthWrite	AuthRead	Config
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
Zone01		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
	Binary	00000000								00000000								01000000								00000010								
	Hex	0								0								40								2								00 00 40 02
		data[0]								data[1]								data[2]								data[3]								

Since the Auth key is key 04, we need to look at the Key04 Configuration.

KeyConfig[InboundAuth] and KeyConfig[ExternalCrypto] are set, so this key is authorized to be used as an Auth Key

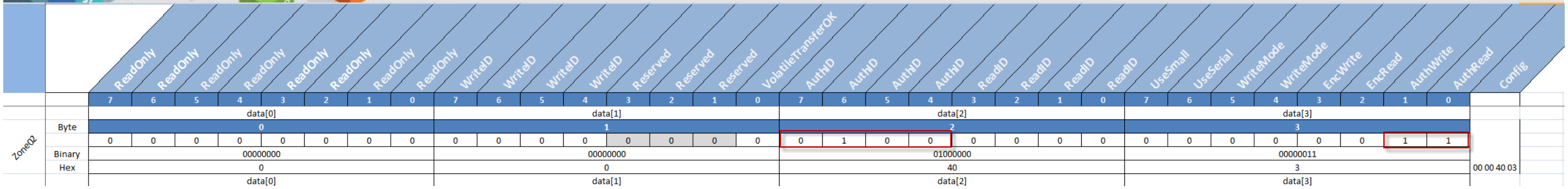


	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ
1			ChangeKeys	Parent	Child	AuthKey	LegacyOK	RandomNonce	InboundAuth	ExternalCrypto	AuthCompute	TransferOK	ChildAuth	ImportOK	AuthOutHold	AuthOut	ChildMac	CounterLimit	CounterNum	CounterNum	CounterNum	CounterNum	LinkPointer	LinkPointer	LinkPointer	LinkPointer	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	DecRead	Config	
2			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0		
24	Key04		1	1	1	1	0	1	1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1		
25		Binary	11110111								11010000								00000101								00000001									
26		Hex	F7								D0								5								1								F7 D0 05 01	

User Zone 02

As with UserZone 0 and 1, we need to see how the zone is configured.

User Zone Config 02 configuration (Address 0xF0C8): 0x03 0x40 0x00 0x00



		ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	WriteID	WriteID	WriteID	WriteID	Reserved	Reserved	Reserved	VolatileTransferOK
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
		data[0]								data[1]							
Zone02	Byte	0								1							
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Binary	00000000								00000000							
	Hex	0								0							
		data[0]								data[1]							

		AuthID	AuthID	AuthID	AuthID	ReadID	ReadID	ReadID	ReadID	UseSmall	UseSerial	WriteMode	WriteMode	EncWrite	EncRead	AuthWrite	AuthRead	Config
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
		data[2]								data[3]								
Zone02	Byte	2								3								
		0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	
	Binary	01000000								00000011								
	Hex	40								3								00 00 40 03
		data[2]								data[3]								

The ZoneConfig[EncRead] & ZoneConfig[EncWrite] are not set, so this zone can be written with Plain Text.

The ZoneConfig[AuthWrite] bit is set, so we must run the Auth Command before writing to zone 02.

The ZoneConfig[AuthRead] bit is set, so we must run the Auth Command before reading from zone 02.

The ZoneConfig[AuthId] bits indicate what key must be used for Pre-Auth (Key 04).

Command (11) - Zone Reads

Read User Zones 0, 1 & 2 - Pre-Auth required to read from zone 2 - Run Command (06)

See Command (10) for Zone and Key configuration

Command (12) - Keyload

Keyload Command (KeyMemory) - Loads Key 6 - Requires pre-auth with Key 13 - (05). Loads a key in to Key Memory 06. Uses Key05 for Encryption and MAC generation. Use this command after running Command (09) – KeyCreate Command– This puts back the original Key

Step 1 – Run Command (06) – Pre-Auth for key 13

Step 2 – Run Command (12) – Load a key in to Key06

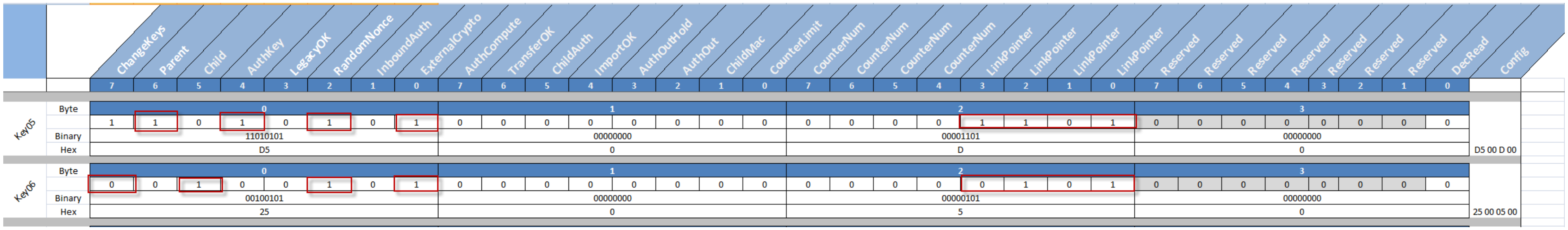
Key 5 and 6 are designated as a Parent-Child pair through the configuration. The parent key is defined as the key that will be used for MAC creation, and Encryption. The parent key is key05 and key06 is the target key, the key to be changed.

KeyConfig05[ChangeKeys] when set, indicates Key updates are permitted after locking for the child key. This is confusing as I thought that is bit setting would indicate that the key changes would apply to the key itself, not the target. In this case, key 05, the parent.

KeyConfig05[AuthKey] indicates that the Key requires Pre-Auth with the value in KeyConfig05[LinkPointer] – Key13

KeyConfig[ExternalCrypto] is set, so this key is authorized to be used for external crypto functions

KeyConfig[RandomNonce] is set, so when executing a nonce, it must always be random



		ChangeKeys	Parent	Child	AuthKey	LegacyOK	RandomNonce	InboundAuth	ExternalCrypto	AuthCompute	TransferOK	ChildAuth	ImportOK	AuthOutHold	AuthOut	ChildMac	CounterLimit
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Key05	Byte	0								1							
		1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0
	Binary	11010101								00000000							
	Hex	D5								0							
Key06	Byte	0								1							
		0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0
	Binary	00100101								00000000							
	Hex	25								0							

		CounterNum	CounterNum	CounterNum	CounterNum	LinkPointer	LinkPointer	LinkPointer	LinkPointer	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	DecRead	Config
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
Key05	Byte	2								3								
		0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	
	Binary	00001101								00000000								
	Hex	D								0								D5 00 D 00
Key06	Byte	2								3								
		0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	
	Binary	00000101								00000000								
	Hex	5								0								25 00 05 00

Command (13) - Keyload (VolatileKey)

Keyload (VolatileKey) – Loads a key in to VolatileKey Memory. For the VolatileKey, when you load the key you must set the ParentKeyID for the encryption/decryption as well as Usage Restrictions (ref 4.3 VolatileKey Configuration of the datasheet).

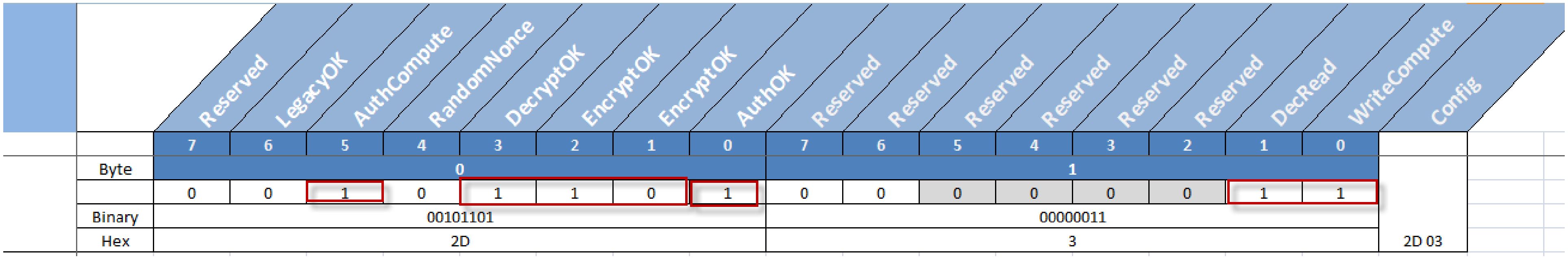
Here we set VolatileKey key with 0x032D. This configures the VolatileKey as follows:

VolatileKey[AuthCompute] - AuthCompute command can be run using this key.

VolatileKey[DecryptOK] - Decrypt command can be run using this key.

VolatileKey[EncryptOK] - Encrypt command can be run using this key without a prior authentication.

VolatileKey[WriteCompute] - WriteCompute command can be run using this key.



	Reserved	LegacyOK	AuthCompute	RandomNonce	DecryptOK	EncryptOK	EncryptOK	AuthOK	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	DecRead	WriteCompute	Config
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
Byte	0								1								
	0	0	1	0	1	1	0	1	0	0	0	0	0	0	1	1	
Binary	00101101								00000011								
Hex	2D								3								2D 03

Use Command (14) to test the VolatileKey

Command (14) - Test the VolatileKey

Test the VolatileKey that was loaded with command (13)

Command (15) - Nonce

This simple show how to create a Nonce from a random number

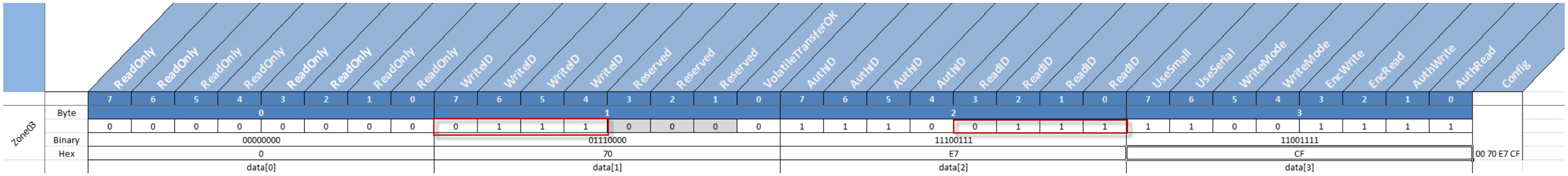
Command (16) - Random Number

Create a Random Number – If the device is not locked, the Key will be 16 bytes of 0xA5

Command (17) - EncWrite

EncWrite to User Zone 3 - Uses Serial Number for encryption. For this command both the Key7 and Zone 3 must be configured.

Here the zone is configured to require the data to be encrypted when read and will require to be encrypted for writing using Key 07.



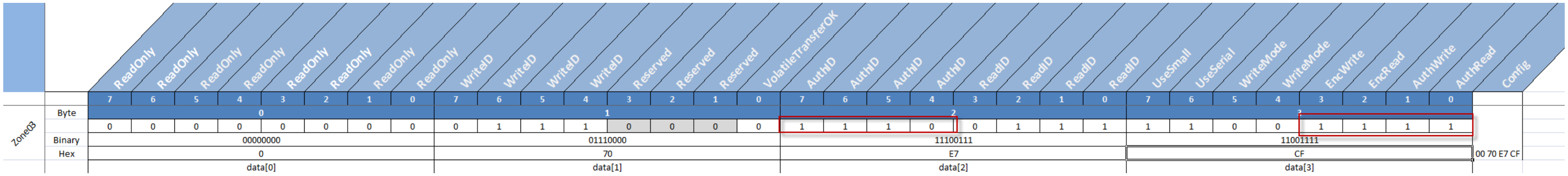
		ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly
Zone03		7	6	5	4	3	2	1	0
	Byte	0							
		0	0	0	0	0	0	0	0
	Binary	00000000							
	Hex	0							
		data[0]							

		WriteID	WriteID	WriteID	WriteID	Reserved	Reserved	Reserved	VolatileTransferOK
Zone03		7	6	5	4	3	2	1	0
	Byte	1							
		0	1	1	1	0	0	0	0
	Binary	01110000							
	Hex	70							
		data[1]							

		AuthID	AuthID	AuthID	AuthID	ReadID	ReadID	ReadID	ReadID
Zone03		7	6	5	4	3	2	1	0
	Byte	2							
		1	1	1	0	0	1	1	1
	Binary	11100111							
	Hex	E7							
		data[2]							

		UseSmall	UseSerial	WriteMode	WriteMode	EncWrite	EncRead	AuthWrite	AuthRead	Config
Zone03		7	6	5	4	3	2	1	0	
	Byte	3								
		1	1	0	0	1	1	1	1	
	Binary	11001111								
	Hex	CF								00 70 E7 CF
		data[3]								

The AuthRead and AuthWrite bits are set, so authorization with key 14 is required before reading or writing (KeyConfig[AuthId])



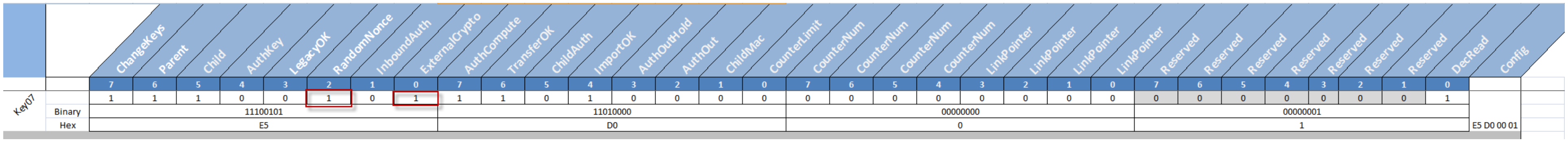
		ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly	ReadOnly
Zone03		7	6	5	4	3	2	1	0
	Byte	0							
		0	0	0	0	0	0	0	0
	Binary	00000000							
	Hex	0							
		data[0]							

		WriteID	WriteID	WriteID	WriteID	Reserved	Reserved	Reserved	VolatileTransferOK
Zone03		7	6	5	4	3	2	1	0
	Byte	1							
		0	1	1	1	0	0	0	0
	Binary	01110000							
	Hex	70							
		data[1]							

		AuthID	AuthID	AuthID	AuthID	ReadID	ReadID	ReadID	ReadID
Zone03		7	6	5	4	3	2	1	0
	Byte	2							
		1	1	1	0	0	1	1	1
	Binary	11100111							
	Hex	E7							
		data[2]							

		UseSmall	UseSerial	WriteMode	WriteMode	EncWrite	EncRead	AuthWrite	AuthRead	Config
Zone03		7	6	5	4	3	2	1	0	
	Byte	3								
		1	1	0	0	1	1	1	1	
	Binary	11001111								
	Hex	CF								00 70 E7 CF
		data[3]								

Key 07, must allow external Crypto. Also note, the nonce generated must be random



		ChangeKeys	Parent	Child	AuthKey	LegacyOK	RandomNonce	InboundAuth	ExternalCrypto
Key07		7	6	5	4	3	2	1	0
		1	1	1	0	0	1	0	1
	Binary	11100101							
	Hex	E5							

		AuthCompute	TransferOK	ChildAuth	ImportOK	AuthOutHold	AuthOut	ChildMac	CounterLimit
Key07		7	6	5	4	3	2	1	0
		1	1	0	1	0	0	0	0
	Binary	11010000							
	Hex	D0							

		CounterNum	CounterNum	CounterNum	CounterNum	LinkPointer	LinkPointer	LinkPointer	LinkPointer
Key07		7	6	5	4	3	2	1	0
		0	0	0	0	0	0	0	0
	Binary	00000000							
	Hex	0							

		Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	DecRead	Config
Key07		7	6	5	4	3	2	1	0	
		0	0	0	0	0	0	0	1	
	Binary	00000001								
	Hex	1								E5 D0 00 01

Command (18) - EncRead

EncRead of User Zone 3 - Uses Serial Number. See Command (17) for configuration settings.

Command (19)- Read Serial Number

Reads the devices Serial Number.

Command (20) – Read Small Zone

Reads the first 4 bytes of the small zone.

Command (21) – Mac Count

Reads the MAC count after encryption.

Command (23) - Lock the Key Memory

Lock the Key Memory - Warning, once locked, you cannot unlock it.

Command (24) - Lock the Configuration

Lock the Configuration - Warning, once locked, you cannot unlock it.