Harris Corporation

Harris AES Software Load Module

Software Version: R04A01

FIPS 140-2 Non-Proprietary Security Policy

FIPS Security Level: 1
Document Version: 1.1

Prepared for:

Harris Corporation
1680 University Avenue
Rochester, NY 14610
United States of America

Phone: +1 (585) 244-5830
http://www.harris.com

Prepared by:

Corsec Security, Inc.
13135 Lee Jackson Memorial Hwy., Suite 220
Fairfax, VA 22033
United States of America

Phone: +1 703 267-6050
http://www.corsec.com

Table of Contents

Table of Figures

List of Tables

1 Introduction

1.1 Purpose

This is a non-proprietary Cryptographic Module Security Policy for the Harris AES Software Load Module (HALM) from Harris Corporation (also referred to as Harris throughout this document). This Security Policy describes how the Harris AES Software Load Module meets the security requirements of FIPS 140-2 and how to run the module in a secure FIPS 140-2 mode. This policy was prepared as part of the Level 1 FIPS 140-2 validation of the module.

FIPS 140-2 (Federal Information Processing Standards Publication 140-2 – *Security Requirements for Cryptographic Modules*) details the U.S. and Canadian Government requirements for cryptographic modules. More information about the FIPS 140-2 standard and validation program is available on the Cryptographic Module Validation Program (CMVP) website, which is maintained by National Institute of Standards and Technology (NIST) and the Canadian Centre for Cyber Security (CCCS): https://csrc.nist.gov/projects/cryptographic-module-validation-program.

The Harris AES Software Load Module is referred to in this document as the HALM, the cryptographic module, or the module.

1.2 References

This document deals only with operations and capabilities of the module in the technical terms of a FIPS 140-2 cryptographic module security policy. More information is available on the module from the following sources:

- The Harris website (www.l3harris.com) contains information on the full line of products from Harris.
- The search page on the CMVP website (https://csrc.nist.gov/Projects/cryptographic-module-validation-program/Validated-Modules/Search) can be used to locate and obtain vendor contact information for technical or sales-related questions about the module.

1.3 Document Organization

The Security Policy document is one document in a FIPS 140-2 Submission Package. In addition to this document, the Submission Package contains:

- Validation Submission Summary
- Vendor Evidence document
- Finite State Model
- Other supporting documentation as additional references

This Security Policy and the other validation submission documents were produced by Corsec Security, Inc., under contract with Harris. With the exception of this Non-Proprietary Security Policy, the FIPS 140-2 Validation Documentation is proprietary to Harris and is releasable only under appropriate non-disclosure agreements. For access to these documents, please contact Harris.

2 HALM Overview

2.1 Overview

Harris is a leading supplier of systems and equipment for public safety, federal, utility, commercial, and transportation markets. Their products range from the most advanced IP[1] voice and data networks, to industry-leading multiband/multimode radios, and even public safety-grade broadband video and data solutions. Their comprehensive line of software-defined radio products and systems support the critical missions of countless public and private agencies, federal and state agencies, and government, defense, and peacekeeping organizations throughout the world. This Security Policy documents the security features of the Harris AES Software Load Module (HALM) incorporated into the Harris 5300 (Mobile 800Mhz only), 5400 (Portable only), 5500 (Portable only), 7200, 7300, Unity, XG-25P, XG-25M, XG-75 UHF-L, XG-75 VHF, XG-75 (800 MHz) and other terminal products, which are single and multi-band, multi-mode radios that deliver end-to-end encrypted digital voice and data communications, and are Project 25 Phase 2 upgradable[2]. Figure 1 and Figure 2 display some of the many radio terminals the Harris AES Software Load Module is incorporated into.

Figure 1 – HALM Portable Terminals (Left to Right: 5400, 7200, 7300, and Unity)

Figure 2 – HALM Mobile Terminals (Left to Right: 5300, 7200, 7300, and Unity)

[1] IP – Internet Protocol
[2] Once the Telecommunications Industry Association (TIA) standard is finalized

The terminal products discussed in this Security Policy support FIPS-Approved secure voice and data communication using Advanced Encryption Standard (AES) algorithm encryption/decryption as specified in FIPS 197. The terminal products also ensure data integrity using a Cipher-based Message Authentication Code (CMAC) algorithm as specified in Special Publication 800-38B. The FIPS 140-2 cryptographic module providing the cryptographic services to the terminals is a single software component called the Harris AES Software Load Module. The HALM provides cryptographic services directly to a Digital Signal Processor (DSP) application on Harris terminals.

The Harris AES Software Load Module is validated at the FIPS 140-2 Section levels shown in Table 1.

Table 1 – Security Level Per FIPS 140-2 Section

Section	Section Title	Level
1	Cryptographic Module Specification	1
2	Cryptographic Module Ports and Interfaces	1
3	Roles, Services, and Authentication	1
4	Finite State Model	1
5	Physical Security	N/A
6	Operational Environment	1
7	Cryptographic Key Management	1
8	EMI/EMC[3]	1
9	Self-tests	1
10	Design Assurance	1
11	Mitigation of Other Attacks	N/A

2.2 Module Specification

The Harris AES Software Load Module is a Level 1 software module with a multi-chip standalone physical embodiment. The physical cryptographic boundary of the HALM is the outer chassis of the terminal in which it is stored and executed. The logical cryptographic boundary of the Harris AES Software Load Module is defined by a single executable (HALM_module_R04A01.ess; Software Version: R04A01) running on a Texas Instruments TMS320C55x DSP/BIOS[4] 5.33.03 software kernel within the Harris terminals. The kernel is a modifiable operational environment since the DSP is also processing instructions supporting the non-security aspects of the terminal. See Figure 3 for a depiction.

The module was tested and validated on the Harris XL-75P VHF radio. As allowed per FIPS Implementation Guidance section G.5, the vendor affirms the module's continued validation compliance when operating on any of the following platforms:

- 5300 (Mobile 800Mhz only)
- 5400 (Portable only)
- 5500 (Portable only)
- 7200
- 7300
- Unity

[3] EMI/EMC – Electromagnetic Interference / Electromagnetic Compatibility
[4] BIOS – Basic Input Output System

- XG-25P
- XG-25M
- XG-75 UHF-L
- XG-75 (800 MHz)

The cryptographic module maintains validation compliance when ported to any other radio platform employing the same operational environment. The CMVP makes no statement as to the correct operation of the module when ported to an operational environment which is not listed on the validation certificate.

The module is entirely encapsulated by the logical cryptographic boundary shown in Figure 3 below. The logical cryptographic boundary of the module is shown with a teal-colored dotted line.

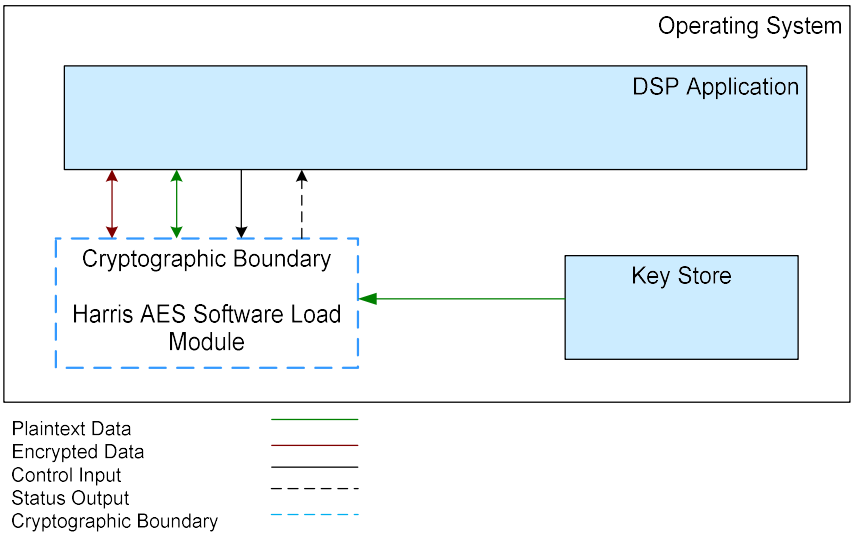

Figure 3 – Logical Cryptographic Boundary

As a software cryptographic module, the Harris AES Software Load Module has a physical cryptographic boundary in addition to its logical cryptographic boundary. The Harris terminal hardware that uses the HALM is designed around a Texas Instruments (TI) TMS320C55x device. Each terminal supports a Liquid Crystal Display (LCD), Light Emitting Diode (LED), keypad, speaker, microphone, Universal Device Connector (UDC), and a number of buttons, knobs and switches (as defined in Table 2 and Table 3). The enclosure of the terminal is considered to be the physical cryptographic boundary of the module as shown with a teal-colored dotted line in Figure 4 below.

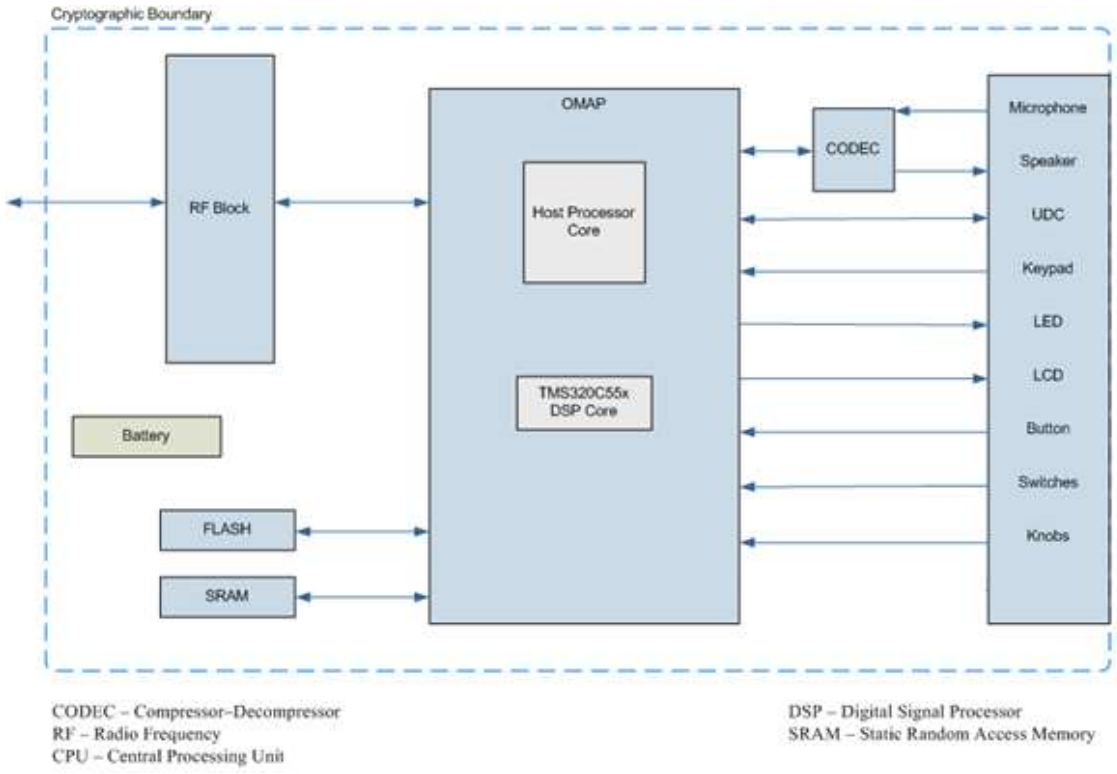

Figure 4 – Physical Cryptographic Boundary

2.3 Module Interfaces

The HALM implements distinct module interfaces in its software design. Physically, the module ports and interfaces are considered to be those of the Harris terminals on which the software executes. However, the software communicates through an Application Programming Interface (API), which allows a DSP application to access the executable. Both the APIs and the physical ports in interfaces can be categorized into the following logical interfaces defined by FIPS 140-2:

- Data Input Interface
- Data Output Interface
- Control Input Interface
- Status Output Interface

These logical interfaces (as defined by FIPS 140-2) map to the module's physical interfaces, as described in Table 2 and Table 3.

Table 2 – FIPS 140-2 Logical Interfaces (Portable Terminals)

FIPS 140-2 Logical Interface	Terminal Physical Port/Interface	Harris AES Software Load Module Interface
Data Input Interface	• Antenna • Microphone • UDC	Arguments for an API call to be used or processed by the module

FIPS 140-2 Logical Interface	Terminal Physical Port/Interface	Harris AES Software Load Module Interface
Data Output Interface	• Speaker • Antenna • LCD • LED • UDC	Arguments for an API call that specify where the result of the API call is stored
Control Input Interface	• Keypad • Knobs: Voice Group Selection Knob, Power On-Off/Volume Knob • Buttons: Emergency Button, PTT[5] Button, Option 1 Button, Option 2 Button • A/B Switch • UDC	API call and accompanying arguments used to control the operation of the module
Status Output Interface	• Speaker • Antenna • UDC • LCD • LED	Return values for API calls

Table 3 – FIPS 140-2 Logical Interfaces (Mobile Terminals)

FIPS 140-2 Logical Interface	Terminal Physical Port/Interface	Harris AES Software Load Module Interface
Data Input Interface	• Antenna Port • GPS Port • Serial Port (DB9) • CAN[6] Ports (qty 2) • I/O Port (44pin D-sub)	Arguments for an API call to be used or processed by the module
Data Output Interface	• Antenna Port • Serial Port (DB9) • CAN Ports (qty 2) • I/O Port (44pin D-sub)	Arguments for an API call that specify where the result of the API call is stored
Control Input Interface	• Antenna Port • Serial Port (DB9) • CAN Ports (qty 2) • I/O Port (44pin D-sub)	API call and accompanying arguments used to control the operation of the module

[5] PTT – Push–to–talk
[6] CAN – Controller Area Network

FIPS 140-2 Logical Interface	Terminal Physical Port/Interface	Harris AES Software Load Module Interface
Status Output Interface	• Antenna Port • GPS Port • Serial Port (DB9) • CAN Ports (qty 2) • I/O Port (44pin D-sub)	Return values for API calls

2.4 Roles, Services, and Authentication

There are two roles in the module (as required by FIPS 140-2) that operators may assume: a Crypto-Officer role and a User role. The terminal operator implicitly assumes one of these roles when selecting each command documented in this section.

2.4.1 Crypto-Officer Role

The Crypto-Officer (CO) role is responsible for initializing the module, self-test execution, and status monitoring. Descriptions of the services available to the CO are provided in Table 4 below. Please note that the keys and CSPs listed in the table indicate the type of access required:

- R – Read access: The Critical Security Parameter (CSP) may be read.
- W – Write access: The CSP may be established, generated, modified, or zeroized.
- X – Execute access: The CSP may be used within an Approved security function.

Table 4 – Mapping of Crypto-Officer Role's Services to Inputs, Outputs, CSPs, and Type of Access

Service	Description	Input	Output	CSP and Type of Access
HALM_INITIALIZE	Performs self-tests on demand	API call	Status output	None
HALM_UNWRAP_KEY	Unwraps a key	API call, key, data	Status output, key	AES-256 key – X
HALM_MAC_GENERATION	Generates a Message Authentication Code (MAC)	API call	Status output	AES-256 key – X

2.4.2 User Role

The User role has the ability to perform the module's cipher operation, and data or voice conversion services. Descriptions of the services available to the role are provided in Table 5 below. Type of access is defined in section 2.4.1 of this document.

Table 5 – Mapping of User Role's Services to Inputs, Outputs, CSPs, and Type of Access

Service	Description	Input	Output	CSP and Type of Access
HALM_GEN_KEYSTREAM	Generates keystream data	API call	Status output	AES-256 key – X
HALM_GEN_PRIVATE_MI	Generates a Message Indicator (MI) from the Initialization Vector (IV) value specified in the data input buffer	API call	Status output	AES-256 key – X
HALM_P25_XOR	Performs logical exclusive or operation	API call, Plaintext or Ciphertext	Status output, Plaintext or Ciphertext	None
HALM_LOAD_KEY	Load key into the module	API call, key	Status output	AES-256 key – R AES-128 key – R
HALM_SEND_STATUS	The status of the last functions called from the HALM_API is returned	API call	Status output	None
HALM_AES_OFB	AES OFB Encrypt	API call, key	Status output, encrypted data	AES-256 key – X
HALM_AES_ECB	AES ECB Encrypt	API call, key	Status output, encrypted data	AES-256 key – X AES-128 key – X
HALM_AES_ECB_DECRYPT	AES ECB Decrypt	API call, key	Status output, decrypted data	AES-256 key – X AES-128 key – X
HALM_AES_CBC	AES CBC Encrypt	API call, key	Status output, encrypted data	AES-256 key – X
HALM_AES_CMAC	AES CMAC	API call, key	Status output, MAC	AES-256 key – X

2.4.3 Non-Approved Services

As allowed per *FIPS 140-2 Implementation Guidance* 1.19, the module alternates on a service-by-service basis between Approved and non-Approved modes of operation. The module executes in the Approved mode of operation by default. The module switches to its non-Approved mode when a service with a non-Approved/non-compliant security function is executed. The module switches back to the Approved mode when a service using an Approved security functions is executed.

Table 6 below lists the services that constitute the non-Approved mode of operation.

Table 6 – Non-Approved Services

Service	Operator		Security Function(s)
	CO	User	
HALM_WRAP_KEY		✓	AES key wrap (non-compliant)

Critical security parameters are not shared between modes.

2.5 Physical Security

The physical security requirements do not apply since the HALM is a software module, which does not implement any physical security mechanisms.

2.6 Operational Environment

The software module was tested and found to be compliant with FIPS 140-2 requirements on the TMS320C55x DSP/BIOS 5.33.03 software kernel configured as described in section 3.1.1 below.

FIPS 140-2 mandates that a software cryptographic module at security level 1 shall be restricted to a single operator mode. By design, the operational environment of the module, the DSP/BIOS software kernel, is always in single operator mode. Hence, no additional steps are required to fulfill the requirement.

All cryptographic keys and CSPs are under the control of the OS[7], which protects the CSPs against unauthorized disclosure, modification, and substitution. The module only allows access to CSPs through its well-defined APIs. The module performs a Software Integrity Test using the AES Cipher-based MAC (CMAC) algorithm.

2.7 Cryptographic Key Management

The module implements the FIPS-Approved algorithms listed in Table 7.

Table 7 – FIPS-Approved Algorithm Implementations

Cert #	Algorithm	Standard	Modes / Methods	Key Lengths / Curves / Moduli	Use
1482	AES[8]	FIPS PUB[9] 197 NIST SP 800-38A	CBC[10], ECB[11], OFB[12]	256	Encryption/decryption
		NIST SP 800-38B	CMAC[13]	256	Generation/verification
2320	AES	FIPS PUB197 NIST SP 800-38A	ECB	128	Encryption/decryption

[7] OS – Operating System
[8] AES – Advanced Encryption Standard
[9] PUB – Publication
[10] CBC – Cipher Block Chaining
[11] ECB – Electronic Code Book
[12] OFB – Output Feedback
[13] CMAC – Cipher-Based Message Authentication Code

The module implements the non-Approved but allowed algorithms shown in Table 8 below.

Table 8 – Allowed Algorithms

Algorithm	Caveat	Use
AES	Key establishment methodology provides 256 bits of encryption strength	Key unwrap[14]

The module implements the non-Approved algorithms listed below (the use of these algorithms is limited to the module's non-Approved mode of operation).

- AES key wrap (non-compliant)

[14] Per FIPS 140-2 IG D.9, any Approved mode of AES is allowed as a key unwrapping technique.

The module supports the critical security parameters listed in Table 9.

Table 9 – List of Cryptographic Keys and CSPs

Key/CSP	Key/CSP Type	Generation / Input	Output	Storage	Zeroization	Use
AES-256 key	256-bit AES key	Generated Within the Physical Boundary; Input Electronically in Plaintext	Never exits the module	Plaintext in volatile memory and flash memory	Power cycle zeroizes volatile memory, key zeroization procedure zeroizes flash memory.	Used as input into ECB/CBC/OFB cipher operations and key unwrapping operations.
AES-128 key	128-bit AES key	Generated Within the Physical Boundary; Input Electronically in Plaintext	Never exits the module	Plaintext in volatile memory and flash memory	Power cycle zeroizes volatile memory, key zeroization procedure zeroizes flash memory.	Used as input into ECB cipher operations

2.8 Self-Tests

When the operating environment is configured as described in section 3.1.1 of this document, self-tests are performed on power-up. The module environment must be so configured to be in compliance with this policy.

The Harris AES Software Load Module performs the self-tests listed in Table 10 at module startup.

Table 10 – Power-Up Self-Tests

Start-Up Test	Description
AES Known Answer Test (KAT)	The AES KAT takes a known key and encrypts a known plaintext value. The encrypted value is compared to the expected ciphertext value. If the values differ, the test is failed. The AES KAT then reverses this process by taking the ciphertext value and key; performing decryption; and comparing the result to the known plaintext value. If the values differ, the test is failed. If they are the same, the test is passed.
Software Integrity Test	The module checks the integrity of the binary (using a CMAC checksum value) at the start-up. If the MAC verifies correctly (i.e., the newly computed MAC is the same as the stored MAC value), the test passes. Otherwise, it fails.

The module is not required to perform any conditional self-tests as it does not perform the generation of random number or asymmetric key pairs.

The module enters the locked error state if it fails either start-up test. An operator may either restart the terminal, by power cycling the unit or return the terminal to a service depot. The module does not implement any Conditional Self-Test.

2.9 Mitigation of Other Attacks

This section is not applicable. The module does not claim to mitigate any additional attacks in an approved FIPS mode of operation.

3 Secure Operation

The Harris AES Software Load Module meets Level 1 requirements for FIPS 140-2. The sections below describe how to place and keep the module in a FIPS-Approved mode of operation.

3.1 Crypto-Officer Guidance

Harris terminals are delivered to end-users preloaded with a package containing its operational environment, DSP application firmware, and Harris AES Software Load Module. The CO does not have to perform any action in order to install or configure the module in the terminals.

In order to operate in FIPS mode, Harris developers have configured the terminals' TMS320C55x DSP/BIOS (version 5.33.03) per sections 3.1.1 and 3.1.2 below to invoke the self-tests on power-up of the DSP.

3.1.1 Environment Configuration

The DSP/BIOS provides a build-time configuration to allow a user-defined initialization function to be called before the operating system is fully initialized. To configure this function, two fields of the DSP/ BIOS Global Settings are set as follows:

1. The "Call User Init Function" field is set to "True"; and,
2. The "User Init Function" is set to the full name (including the leading underscore) of a suitable initialization function.

The DSP/BIOS is configured to execute a user initialization function called fips_init(). Figure 5 illustrates the required environment configuration. Section 3.1.2 below provides a description of the function.

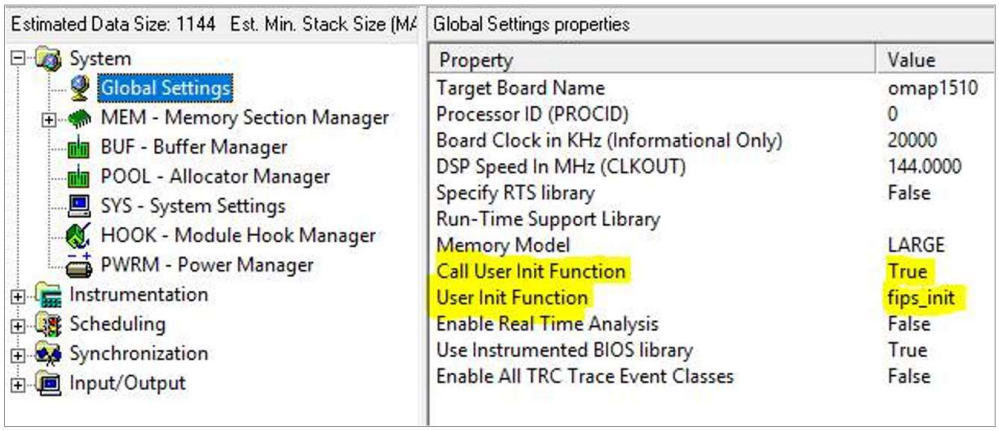

Figure 5 – TMS320C55x DSP/BIOS Global Settings

Additional information on the user initialization function can be found in in section 2.6 (GBL Module) of the *TMS320C55x DSP/BIOS 5.x Application Programming Interface (API) Reference Guide* (TI DOC SPRU404Q).

3.1.2 User Initialization Function

The fips_init() function executes the HALM initialization functions and FIPS-required power-up self-tests before invocation of the application, comprising three steps:

1. fips_init() provides pointers to memory managements functions (malloc() and free() to the HALM module;
2. fips_init() invokes the HALM "startup" function to initialize certain internal fields and to allocate required memory using the provided malloc() function; and,
3. fips_init() invokes the HALM initialize routine to perform the required integrity check and algorithm tests.

The following code fragment illustrates the fips_init() function needed to ensure operation in FIPS mode.

```
//
Ptr fips_heap_top = (Ptr)0x0080; // Reuse reserved heap memory segment
Ptr FIPS_alloc(Int segid, Uns size, Uns align)
{
  Ptr ptr = fips_heap_top;
  fips_heap_top = (Uns *)((Uns)fips_heap_top + size*sizeof(Uns));
  return ptr;
}

// Since free is only performed when module is torn down there is nothing to do here.
Bool FIPS_free(Int segid, Ptr addr, Uns size)
{
  return TRUE;
}

// This function assumes that the FIPS module is loaded into DSP memory and is ready to
be initialized and // run self test
Uns fips_init(void)
{
  // The fips_init function may include additional functionality (e.g., defensively
verifying that the FIPS load module has been loaded by the application processor prior to
reset of the DSP) as required

  {
     // Initialize the FIPS module external jump table for malloc/free
     FIPS_LM_header->global_jump[0] = (LMPF)&FIPS_alloc;
     FIPS_LM_header->global_jump[1] = (LMPF)&FIPS_free;
     FIPS_LM_header->global_jump[2] = (LMPF)&noExtFipsFunction;
     FIPS_LM_header->global_jump[3] = (LMPF)&noExtFipsFunction;
     FIPS_LM_header->global_jump[4] = (LMPF)&noExtFipsFunction;
     FIPS_LM_header->global_jump[5] = (LMPF)&noExtFipsFunction;

     // Call the FIPS module startup routine
     FIPS_LM_header->startup(0);

     // Call the FIPS module self-test and set the status
     msiCmdStruct.msiMsgId = HALM_INITIALIZE;
     halmStatusWord = HALM_API(&msiCmdStruct,(Uns *)0,(Uns *)0);
  }
}
```

3.1.3 Management

The Crypto-Officer should monitor the module's status regularly. If any irregular activity is noticed or the module is consistently reporting errors, then Harris customer support should be contacted.

3.1.4 Zeroization

The module does not store any keys or CSPs within its logical boundary. All ephemeral keys that are used by the module are zeroized upon reboot or session termination. Outside of the module, external flash memory stores operational keys. These keys are loaded at the point of origin, and may be zeroized by the operator of the radio using the "Key Zero" procedure documented in the terminal's Operator's Manual. After the external flash memory keys are zeroized, the radio must be returned to the point of origin for repair.

3.2 User Guidance

Users can only access the module's cryptographic functionalities that are available to them. Although the User does not have any ability to modify the configuration of the module, they should report to the Crypto-Officer if any irregular activity is noticed.

4 Acronyms

This section describes the acronyms.

Table 11 – Acronyms

Acronym	Definition
ADC	Analog to Digital Converter
AES	Advanced Encryption Standard
API	Application Programming Interface
BIOS	Basic Input/Output System
CBC	Cipher Block Chaining
CCCS	Canadian Centre for Cyber Security
CMAC	Cipher-based Message Authentication Code
CMVP	Cryptographic Module Validation Program
CO	Crypto-Officer
CODEC	Compressor-Decompressor
CPLD	Complex Programmable Logic Device
CRC	Cyclical Redundancy Check
CSP	Critical Security Parameter
DAC	Digital to Analog Converter
DSP	Digital Signal Processor
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FIPS	Federal Information Processing Standard
HALM	Harris AES Software Load Module
IP	Internet Protocol
IV	Initialization Vector
KAT	Known Answer Test
LCD	Liquid Crystal Display
LED	Light Emitting Diode
MAC	Message Authentication Code
MI	Message Indicator
NIST	National Institute of Standards and Technology
OFB	Output Feedback
OMAP	Open Multimedia Application Platform
OS	Operating System

Acronym	Definition
PTT	Push-to-talk
RAM	Random Access Memory
RPM2	Radio Personality Manager 2
RX	Receive
SRAM	Static Random Access Memory
TI	Texas Instruments
TX	Transmit
UDC	Universal Device Connector

Prepared by:
Corsec Security, Inc.

13921 Park Center Road, Suite 460
Herndon, VA 20171
United States of America

Phone: +1 (703) 267-6050
Email: info@corsec.com
http://www.corsec.com