National Security Agency

Fort George G. Meade, Maryland

SPECIFICATION NSA NO. 86-3
4 MARCH 1988

NATIONAL SECURITY AGENCY SPECIFICATION

TEPACHE
INTERFACE CONTROL DOCUMENT
(ICD)

FOR OFFICIAL USE ONLY

TABLE OF CONTENTS

FOR OFFICIAL USE ONLY

FOR OFFICIAL USE ONLY

LIST OF FIGURES

LIST OF TABLES

INTERFACE CONTROL DOCUMENT (ICD)

1. INTRODUCTION

1.1. SCOPE

This document defines the functional, electrical and mechanical interface of the TEPACHE embeddable communications security Key Generator Module. The ICD is intended to serve the needs of the host system end item equipment developer. This document contains the detailed technical information that is required to integrate this communications security (COMSEC) module into existing or future telecommunications and information processing systems. System-unique security engineering guidance for the proper use of a TEPACHE module in telecommunication equipment handling classified information will be provided by the National Security Agency (NSA) on an individual application basis.

1.2. TERMINOLOGY

Throughout this document, the embeddable TEPACHE COMSEC Key Generator Module will be referred to as the KGM, and the terminal equipment containing the KGM will be referred to as the host. All signals are referenced to the KGM, that is, inputs are signals from the host or the output circuit (I/O) to the KGM and output signals are from the KGM to the host or I/O. A comprehensive glossary of terms and acronyms is provided in Section 9.

1.3. COMMENTS

Comments and questions regarding this ICD or the application of the KGM to any specific program should be forwarded to:

DIRECTOR, NSA
ATTENTION Y24
9800 SAVAGE ROAD
FT. GEORGE G. MEADE, MD 20755-6000

2. APPLICABLE DOCUMENTS

CSESD-11 - Communications Security Equipment System Document for fill devices KYK-13, KYX-15, KOI-18 (Document is CONFIDENTIAL).

Handling and Control Requirements for CCI Equipment and Components During Manufacture and Assembly, dated 1 October 1985. (document is UNCLASSIFIED).

Fill connector drawing 0N241775

3. SECURITY CONSIDERATIONS

3.1. CONTROLLED CRYPTOGRAPHIC ITEMS

The KGM is a Controlled Cryptographic Item (CCI) which is unclassified. Because it embodies a classified cryptographic design and can be used to secure classified information when appropriately keyed, the KGM must be controlled in accordance with the requirements set forth in a document titled, "HANDLING AND CONTROL REQUIREMENTS FOR CCI EQUIPMENT AND COMPONENTS DURING MANUFACTURE AND ASSEMBLY", dated 1 October 1985. The CCI controls apply to the KGM until integration, and then to equipments or systems that contain the CCI KGM.

3.2. NSA ENDORSEMENT

The KGM is approved by the National Security Agency for processing classified information at all levels. This approval is not extended automatically to host systems or equipment that contain the KGM. All systems and components of systems that are required to process classified information must be evaluated on a system unique basis prior to endorsement. For further information, and a list of those vendors who are authorized to produce and sell the TEPACHE KGM with NSA endorsement, contact the NSA.

3.3. TEMPEST

TEMPEST design techniques have been used throughout the development of the KGM. However, the KGM carries no implied TEMPEST approval. Each integration of the KGM into host end item equipment must be evaluated in light of system specific TEMPEST requirements. Contact NSA for further guidance.

4. DESCRIPTION

4.1. FUNCTIONAL DESCRIPTION

The KGM is a member of the National Security Agency's family of standard embeddable COMSEC products. It is a general purpose half duplex cryptographic device capable of providing COMSEC protection of digital data at all classification levels. The KGM is designed to support a wide variety of system architectures by acting as an intelligent slave to the host terminal equipment. The microprocessor compatible interface and control structure is highly flexible, and the powerful command set will permit the host equipment developer to select the most efficient modes of operation to satisfy unique system requirements.

The purpose of the KGM is to encrypt and decrypt digital data for file protection, point to point communications security, and network communications security. Encrypting information permits it to be transmitted over normal communications channels or stored in unprotected media. Decrypting the enciphered signal "recovers" the information in its original form.

The KGM is intended for use in a wide variety of microprocessor-based commercial equipments such as host computers, terminals, workstations, and smart peripherals. It is placed in series between a host computer and a communications or disk controller. Both the host and I/O ports of the KGM are 16-bits wide and are designed to interface with both 8-and 16-bit hosts. Handshake lines from the host to the KGM and from the KGM to the I/O allow system control. Six cryptographic modes can be used for encryption and decryption of data. The KGM is asynchronous and can support encryption and decryption in some modes up to 7 Mbit/sec. All commands and data pass through the KGM allowing it to control all information for unprotected communications or data storage. The KGM can be configured to send the data, after being encrypted, to the I/O or to the host. Conversely, encrypted data coming from the I/O can be routed to the host for storage or directly to the KGM for decryption. The configuration of the KGM is set up via commands by the host. The KGM also controls and audits the amount of both command and data bypassed to the I/O from the host, and maintains control over the fill, storage, and manipulation of keys. Commands, plain text, cipher text, and RED key are all managed within the KGM.

4.2. KGM FEATURES

* Multilayer printed wiring board configuration
* 96 pin package
* PC board mountable within the host equipment

- Operates on standard 5V DC power from host
- 6 cryptographic operating modes
- Message Authentication Code (MAC) mode
- Asynchronous operation at speeds up to 7 Mbit/sec
- Compatible with 8-and 16-bit hosts
- Operates with input block sizes of 8, 64, and 128 bits, depending on mode
- LSTTL compatible
- Microprocessor style control interface
- Complies with the NSA standard interface
- Supports NSA standard 8-bit COMSEC Command Language (CCL)
- Has commanded self-test capability
- Provides controlled and auditable bypass
- Configurable as a System Manager or Node
- Compatible with the KOI-18 common fill device and the DATA TRANSFER DEVICE (DTD)
- Internal volatile active storage for 255 keys and key packets
- Capable of wrapping/unwrapping keys for storage in host
- Capable of combining keys, updating keys, and Over The Air Rekeying (OTAR)
- CRYPTO IGNITION KEY (CIK) use optional (host discretion)

4.3. ARCHITECTURAL DESCRIPTION

A block diagram of the KGM is shown in Figure 4-1. The KGM is designed to work in either 8-bit or 16-bit bus systems. All commands and responses from the KGM are 8 bits, independent of mode. For data, the 16 bit configuration is the default. To select the 8-bit data mode, the SET 8-BIT MODE command must be given. When the 8-bit mode is set, host lines h8-h15 and i8-i15 must be pulled up to 5v (See Section 7).

The KGM has 6 ports - host port, I/O port, fill port, CIK port, housekeeping port, and power port - which are described in detail below. Section 7 provides details on KGM pinouts and electrical characteristics. Throughout this document, an * in conjunction with a signal name indicates an active low signal.

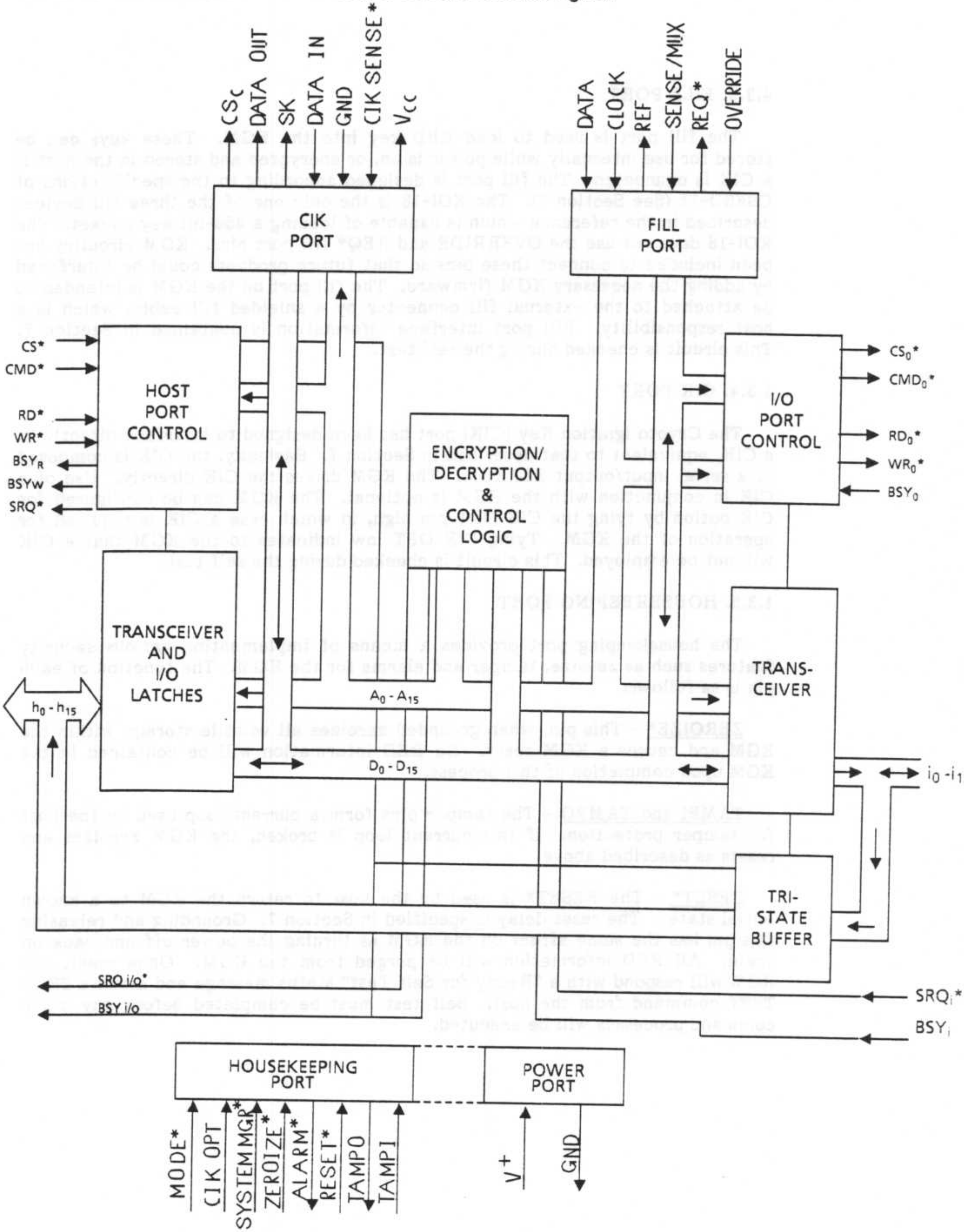
4.3.1. HOST PORT

The host port circuitry includes a 16-bit transceiver latch, and port control logic. Sixteen-bit data words or 8-bit command words pass from the host into the KGM. Data and status words may also pass from the KGM back to the host. All data, commands, and status which are transferred between the KGM and the host pass over the host port lines (h0-h15). This includes words read from the I/O port, words read from the host port latch, and commands and data written to the KGM by the host. The port control circuitry provides handshaking and timing signals for the host port and is described fully in Section 7. The proper operation of the host port is checked during the SELF TEST command process.

4.3.2. I/O PORT

The I/O circuitry includes port control logic, a 16-bit transceiver and a 16-bit tri-state buffer. Data and commands pass from the KGM to the I/O through the transceiver. The tri-state buffer permits 16 bits of data coming from the I/O to bypass through the KGM to the host. This bypass circuit was included in the KGM to prevent the KGM from slowing down the I/O, and so that the host builder will not be required to provide security analysis for the circuit. The service request (SRQi*) and the busy (BSYi) are used by the KGM and sent to the host to synchronize data flow from the I/O. The other port control signals are all used to control the output transceiver. The port control circuitry provides handshaking and timing signals for the I/O port. These signals are fully described in Section 7. The proper operation of the I/O port is checked during the self test.

Fig. 4-1: KGM Block Diagram



4.3.3. FILL PORT

The fill port is used to load RED key into the KGM. These keys can be stored for use internally while power is on, or encrypted and stored in the host if a CIK is connected. The fill port is designed according to the specifications of CSESD-11 (See Section 2). The KOI-18 is the only one of the three fill devices described in the reference which is capable of loading a 256-bit key packet. The KOI-18 does not use the OVERRIDE and REQ* fill port pins. KGM circuitry has been included to connect these pins so that future products could be interfaced by adding the necessary KGM firmware. The fill port on the KGM is intended to be attached to the external fill connector by a shielded fill cable, which is a host responsibility. Fill port interface information is contained in Section 7. This circuit is checked during the self test.

4.3.4. CIK PORT

The Crypto Ignition Key (CIK) port has been designed to interface directly to a CIK, equivalent to that described in Section 7. Basically, the CIK is composed of a serial input/output EEPROM. The KGM drives the CIK directly. Use of a CIK in conjunction with the KGM is optional. The KGM can be configured for CIK option by tying the CIK OPT pin high, in which case a CIK is required for operation of the KGM. Tying CIK OPT low indicates to the KGM that a CIK will not be employed. This circuit is checked during the self test.

4.3.5. HOUSEKEEPING PORT

The housekeeping port provides a means of implementing various security features such as zeroize, tamper and alarms for the KGM. The function of each pin is as follows:

ZEROIZE* - This pin, when grounded zeroizes all volatile storage within the KGM and causes a KGM reset. No RED information will be contained in the KGM upon completion of this process.

TAMPI and TAMPO - The tamper pins form a current loop used by the host for tamper protection. If this current loop is broken, the KGM zeroizes and resets as described above.

RESET* - The RESET* is used by the host to return the KGM to a known initial state . The reset delay is specified in Section 7. Grounding and releasing this pin has the same affect on the KGM as turning the power off and back on again. All RED information will be purged from the KGM. Once reset, the KGM will respond with a "Ready for Self Test" status message and await a SELF TEST command from the host. Self test must be completed before any other command processes will be executed.

ALARM* - When the KGM detects an internal failure, it will ground the ALARM* pin. The alarm output can be used by the host to activate an audible or visual alarm. When in an alarm condition, the KGM stops all internal processing, sets the ports to high impedance and will not output any data or accept any commands. An alarm can be cleared by performing a hardware reset (grounding the RESET* pin), zeroizing (grounding the ZEROIZE* pin), or cycling power off/on. Alarms are purposely created during self test and should be ignored by the host during that time.

SYSTEM MGR* - The System Manager pin is used to select either System Manager (ground) or Node (+5V) operating configurations.

CIK OPT - The CIK Option pin is used to configure the KGM to function with a CIK (CIK Option pin tied to +5V) or to function without a CIK (CIK Option pin tied to ground). This pin is sampled during KGM initialization and should not be changed during operation.

MODE* - The MODE pin is used to specify which modes can be selected by the host. If the pin is grounded Mode F and continued state operation for Modes A, B,and C are available for selection by the host if required. If the pin is tied to 5V Mode F and continued state operation of Modes A, B, and C are not available for selection by the host. This pin is intended to be jumpered to either ground or +5V.

4.3.6. Power Port

This port provides a means of accepting prime power input for the KGM. The host provides +5V DC power to the KGM (See Section 7). Internally, the KGM has low power detection circuitry. If power is interrupted, the low power detector will reset the KGM. The host reset and zeroize signals can also be used to reset the KGM when the host detects low power.

4.4. CRYPTOGRAPHIC OPERATING MODES

The KGM has 6 cryptographic modes of operation. They are referred to as modes A, B, C, D, E, and F for both encryption and decryption. The selection of the one mode of operation for data processing over another is based on system requirements. Each data processing mode offers unique synchronization and error extension features. A Message Authentication Code (MAC) function is also available. Both data and commands may be bypassed through the KGM from the host to the I/O. A Bypass Control Word within the KGM determines the allowable amount of bypass. Section 5 contains additional details.

4.5. KEY PROCESSING METHODS

Secure communication systems rely heavily on the secure, timely and relatively transparent distribution of keys. This process is referred to as the key management system. The KGM provides the means for communication system developers to custom design a key management architecture that "fits" the application. Internal active storage of multiple keys, local (within the host equipment) storage of encrypted keys, crypto ignition key, remote electronic rekeying, key combining, key updating and manual key distribution techniques are all available for use in the KGM. The specific details of key processing methods are contained in Section 5.

4.6. PRIME POWER

Prime power, +5vdc +/-5% from the host is required by the KGM for reliable operation. The KGM contains under voltage detectors for prime power input. Detection of an under voltage condition on the prime power input causes the KGM to perform a complete erasure of all internally stored keys. Over voltage protection (See Absolute Maximum Ratings) must be provided by the host equipment.

5. PRINCIPLES OF OPERATION

The KGM has been designed to provide a cryptographic data processing function for many different types of digital data telecommunication systems. The interface structure and operating modes have been developed to provide very efficient data and command processing rates, while promoting a simple and secure integration effort. The data processing modes have been selected to support a variety of cryptographic schemes. These methods are important in selecting the mode of operation that best suits the needs of the system in terms of overhead, noise environment and processing complexity.

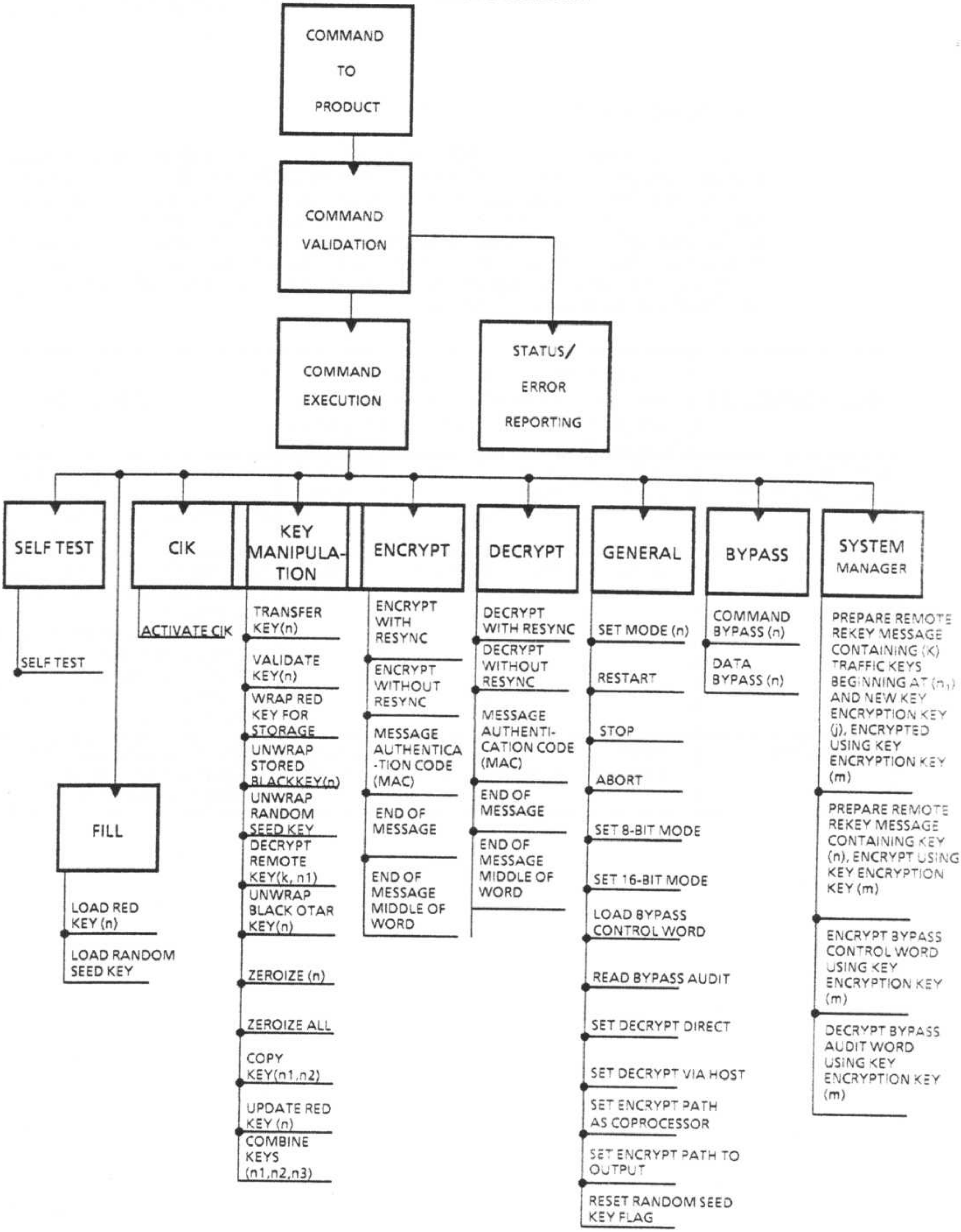
KGM operation is dependent on host commands. Figure 5-1 shows the major groups of commands and the available individual commands. Section 6 contains additional details on each command. The proposed design enhances security by enforcing the separation of processes within the KGM to insure that only a single process is active at any time. Security starts at the Executive where host commands must be validated before execution. Only one command is processed at a time. After leaving the Executive, security is dependent upon the individual procedure being executed in processing the command. This is addressed by comprehensive exception handling at a procedural process level and immediate return to the Executive when any error is detected. The KGM attempts no recovery from detected errors. It merely makes information about the error available to the host via the status message. In the Executive, the KGM will await another command from the host. If the host reattempts the same command and the error condition persists, the abort condition will again occur.

5.0 CONFIGURATION

The commands allow the KGM to be configured in different ways to make systems integration easier. The "SET 8-BIT MODE" and "SET 16-BIT MODE" commands allow the host and I/O ports to be configured as 8-bit ports or as 16-bit ports. The default mode is 16-bits. Also note that all commands and status words have only 8 significant bits and are the same for both 8- and 16-bit interfaces. The Encrypt and Decrypt paths can also be configured to aid in integration for different applications. There are three basic encrypt/decrypt path configurations as listed below:

CONFIGURATION	DECRYPT		ENCRYPT		COMMANDS
	INPUT	OUTPUT	INPUT	OUTPUT	
Normal (Decrypt Direct)	I/O Port	Host Port	Host Port	I/O Port	1) Default configuration on power up. 2) SET DECRYPT DIRECT command sets configuration.
Mixed	Host Port	Host Port	Host Port	I/O Port	1) A SET DECRYPT VIA HOST command sets this configuration. 2) If previously in coprocessor mode and a SET ENCRYPT PATH TO OUTPUT command is given this configuration will result.
Coprocessor	Host Port	Host Port	Host Port	Host Port	1) SET ENCRYPT PATH AS COPROCESSOR command sets this configuration.

Fig. 5-1: KGM Command Structure



FOR OFFICIAL USE ONLY

5.1. ENCRYPTION AND DECRYPTION

To set up an encryption process, the following steps are necessary. The host specifies the appropriate mode by sending the command SET MODE (n). Once the mode is set, after power-up it remains set until changed by another set mode command. The host then specifies which key is to be used by sending the command TRANSFER KEY (n) [This is not necessary if switching between decrypt and encrypt.]. The KGM checks the key in location n to assure that it is a traffic key and that the parities are correct. It then returns a status message which tells the host if previous states have been saved or not (See discussion below). Next, the host commands ENCRYPT WITH RESYNC or ENCRYPT WITHOUT RESYNC*. The host now begins sending data to be encrypted. The KGM uses the BSYw line to specify to the host when it is ready to accept more data. When encryption is complete, the KGM writes the cipher text to the I/O port (or to the host port if the KGM had been configured for the coprocessor mode using the command SET ENCRYPT PATH AS COPROCESSOR). The I/O uses its BSYi line to indicate when it is ready to accept more data; the KGM uses its BSYr line to indicate to the host when it is acceptable to read the host port, if configured for the coprocessor mode. When all data has been written to the KGM by the host, the host can issue an END OF MESSAGE or END OF MESSAGE MIDDLE OF WORD command. This will cause the KGM to process any words of data previously written to the KGM but not yet processed, even if it is an incomplete block. The KGM then awaits more data for encryption. A STOP command will cause the KGM to process the remaining words of data, save the current crypto states (MI's) for the encrypt/decrypt process, purge all temporary RED information (not including key storage) from the KGM, and return to the Executive. An ABORT command will terminate data processing, purge all temporary RED information (not including key storage) from the KGM, and return to the Executive. No remaining words of data will be processed and the current crypto states (MI's) will not be saved. A DECRYPT WITH RESYNC or DECRYPT WITHOUT RESYNC* command can also be used. This allows the KGM to switch rapidly between encrypt and decrypt using the same key. These commands have the same affect as the END OF MESSAGE command except when all data being encrypted has been sent out, it waits for data to be decrypted. When any other command is recieved, the KGM performs the equivelant of a STOP command and then executes the command.

To set up the decryption process, the following steps are necessary. The host specifies the mode, if not previously specified, by sending the command SET MODE (n). The host then specifies which key is to be use by sending the command TRANSFER KEY (n) [The TRANSFER KEY command is not necessary if switching from encrypt to decrypt using the same key since the key and the state information are already in place.]. Next, the host commands DECRYPT WITH RESYNC or DECRYPT WITHOUT RESYNC*. If the KGM had been configured using the command SET DECRYPT DIRECT, ciphertext will be routed directly from the I/O circuit into the KGM for decryption. The I/O circuit uses the BSYo line to indicate to the KGM when cipher text is available. After decryption, the KGM uses the BSYr line to indicate to the host when plain text is available to be read by the host at the host port. If the KGM has been

Continued state operation (ENCRYPT or DECRYPT WITHOUT RESYNC) for Modes A, B, and C is not allowed unless the MODE pin is grounded. Modes D and E allow continued state operation independent of the MODE* pin.

configured using the command SET DECRYPT VIA HOST, cipher text will be routed from the I/O circuit to the host for queuing or storage. The KGM uses the BSYi/o line to indicate when cipher text is available for the host to read. To decrypt cipher text stored or queued in the host, the KGM uses the BSYw line to indicate to the host when it is ready to accept data. When decryption is complete, the KGM uses its BSYr line to indicate to the host when it is acceptable to read the plain text at the host port. An END OF MESSAGE or END OF MESSAGE MIDDLE OF WORD command will cause the KGM to process the remaining words of data and wait for additional data to be decrypted. A STOP command will cause the KGM to process the remaining words of data, save the current crypto states (MI's) for the encrypt/decrypt process, purge all temporary RED information (not including key storage) from the KGM, and return to the Executive. An ABORT command will terminate data processing, purge all temporary RED information (not including key storage) from the KGM, and return to the Executive. No remaining words of data will be processed and the current crypto states (MI's) will not be saved. An ENCRYPT WITH RESYNC or ENCRYPT WITHOUT RESYNC* command can also be used in place of the END OF MESSAGE command to stop the decryption process when the KGM is being rapidly switched between encrypt and decrypt, as when emulating full-duplex communications. This command is very similar to END OF MESSAGE except that it shifts to encrypt and awaits data. Any other command causes the KGM to execute the equivalent of a STOP and then executes the command.

5.1.1. MODE A

This mode requires 64-bit data blocks for encryption or decryption. From a cold start, the command ENCRYPT WITH RESYNC or DECRYPT WITH RESYNC is used. When either of these commands is used, the first 64 bits transmitted or received are used to establish cryptographic synchronization. Subsequent 64-bit blocks can be processed without further need to establish synchronization. If any other command other than ABORT is used at the end of the message by both transmitter and receiver, the next message can be sent out to the same receiver without resynchronizing. If desired to do this, the next message should be processed using the command ENCRYPT WITHOUT RESYNC* or DECRYPT WITHOUT RESYNC*. This mode has error extension. Hence, a low noise communication channel is desired. This mode provides the maximum data throughput rate.

5.1.2. MODE B

This mode is similar in operation to Mode A; however, 8-bit data blocks are used for encryption or decryption instead of 64-bit blocks. This mode will only operate with 8-bit mode selected. From a cold start, the command ENCRYPT WITH RESYNC or DECRYPT WITH RESYNC is used. When either of these commands is used, the first 64-bits transmitted or received are used to establish cryptographic synchronization. Subsequent 8-bit blocks can be processed without further need to establish synchronization. If any command other than ABORT is used at the end of the message by both transmitter and receiver, the next message can be sent out to the same receiver without resynchronizing.*

Continued state operation (ENCRYPT or DECRYPT WITHOUT RESYNC) for Modes A, B, and C is not allowed unless the MODE pin is grounded. Modes D and E allow continued state operation independent of the MODE* pin.

If desired to do this the next message should be processed using the command ENCRYPT WITHOUT RESYNC* or DECRYPT WITHOUT RESYNC*. This mode has error extension. Hence, a low-noise communication channel is desired. This mode, although much slower than Mode A, is well suited for terminal applications.

5.1.3. MODE C

This mode requires 64-bit data blocks for encryption or decryption. When a key is used for the first time in a session, the command ENCRYPT WITH RESYNC or DECRYPT WITH RESYNC is used. When either of these commands is used, the first 64 bits transmitted or received are used to established cryptographic synchronization. Subsequent 64-bit blocks can be processed without further need to establish synchronization. If any command other than ABORT is used at the end of the message by both transmitter and receiver, the next message can be sent out to the same receiver without resynchronizing.* If desired to do this, the next message should be processed using the command ENCRYPT WITHOUT RESYNC* or DECRYPT WITHOUT RESYNC*. This mode has no error extension. Hence, noisy communications channels could utilize this mode.

5.1.4. MODE D

This mode requires 128-bit data frames for encryption or decryption. The data frames are represented by a pair of blocks each 64-bits in length. From a cold start, the command ENCRYPT WITH RESYNC or DECRYPT WITH RESYNC is used. When either of these commands is used, the first 64-bits transmitted or received are used to establish cryptographic synchronization. Once synchronization is attained, data frames are ready for processing. There is one peculiarity of this mode involving input and output block sizes. In the 16-bit encrypt or decrypt mode, after synchronization is achieved, the KGM expects the following sequence to occur:

1) Host writes 9 data words to the KGM.

2) The KGM encrypts/decrypts the first 8 data words and outputs to either host or I/O depending on configuration.

3) Host sends next 8 data words to KGM.

4) KGM encrypts/decrypts the last data word from the previous input block and the first seven words of the last input block and outputs them to host or I/O.

5) The process continues until the KGM receives a command (END OF MESSAGE, END OF MESSAGE MIDDLE OF WORD, or any command other than ABORT). Input blocks of less than 8 words must be followed by one of the above commands in order for the KGM to complete data processing.

Continued state operation (ENCRYPT or DECRYPT WITHOUT RESYNC) for Modes A, B, and C is not allowed unless the MODE pin is grounded. Modes D and E allow continued state operation independent of the MODE* pin.

When the 8-bit mode is used, the number of words are doubled (two 8-bit words for each 16-bit word described above). If any command other than ABORT is used at the end of the message by both the transmitter and receiver, the next message can be sent out to the same receiver without resynchronization. The next message should be processed using the command ENCRYPT WITHOUT RESYNC or DECRYPT WITHOUT RESYNC. This mode has error extension. Hence, a low-noise communication channel is desirable.

5.1.5. MODE E

This mode, although similar to Mode D, does not have the peculiarity of Mode D and requires only 8-bit data frames for encryption or decryption. It employs the same synchronization scheme as Mode D. This mode will only operate with the 8-bit mode selected. This mode has error extension. Hence, a low noise environment is desirable.

5.1.6. MODE F

MODE F is not allowed unless the MODE* pin is grounded. This mode requires 128-bit data frames for encryption or decryption. The data frames are represented by a pair of blocks each 64-bits in length. No resync vectors are needed with this mode. In this mode a 128-bit block encrypts to a unique 128-bit block. This mode has error extension. Hence, a low-noise communication channel is desirable.

5.2. CRYPTOGRAPHIC SYNCHRONIZATION

Cryptographic synchronization generation is handled entirely by the KGM in all cryptographic modes of operation. After a prescribed number of bits have been received, the KGM assumes synchronization has been established. It is the responsibility of the host to verify synchronization and initiate resync if necessary.

The KGM contains active storage for 253 traffic Keys and their individual Message Indicators (MI's). These MI's can be used to place the cryptographic logic into the same state as it was previously when communicating with a particular user of that key. Continued state operation (ENCRYPT or DECRYPT WITHOUT RESYNC) for Modes A, B, and C is not alowed unless the MODE* pin is grounded. If the same key is used to communicate with several users, the key can be placed in several storage locations within the KGM and separate MI's can be stored for each user. Since the KGM automatically saves the crypto state at the end of each message, the overhead necessary for resynchronization with each message can be minimized. These saved states (MI's) will be lost when the KGM is reset or power is cycled. They are also not part of the key packet which is stored in the host when a WRAP RED KEY FOR STORAGE command is used. They are also not transferred to a new key location when a copy key command is used. This means that after each power-up it is necessary to send at least the first message WITH RESYNC.

5.3. KEY PROCESSING

The KGM provides a considerable amount of flexibility in storing and processing key. All keys used within the KGM are a part of an associated key packet that contains the following information:

Key
Parity
Key Type
Update Count

During any key processing operation, several of these fields are checked to insure that the proper key is being used. Error conditions are reported to the host, and operations may be denied if an attempt is made to use a key improperly. Refer to detailed command processing descriptions in Section 6 for more information.

5.3.1. MANUAL KEY ENTRY

The manual key entry port (fill port) is unidirectional (input only). It is used to input RED key with associated key packet. When the KGM receives the fill command, LOAD RED KEY (n), it checks the fill sense line to determine if a fill device is attached, and then looks for a serial clock signal. It then reads serial data in sync with the serial clock. Internally, it checks the key parity, packet parity and checkword. It will load the key into the designated location if the checkword and parity check, and the key packet indicates that it is the proper key for the designated location. The KGM will notify the host via an interrupt (Service Request) and an appropriate status message that the commanded process has been completed or what error has occurred. As a result of this process, a RED key packet is present in location (n). The MI's for location (n) are both zero.

If multiple keys are to be loaded, this process must be invoked repeatedly by sending a new command before loading each key. Disconnecting the fill device will abort the fill process once the command has been given. If the process is aborted, the KGM zeroizes all temporary storage locations and returns to the Executive. Keys already loaded will remain in active RAM storage, and will be available for use.

Encrypted keys can also be delivered to the host via disk or typed in on the keyboard. They can be loaded into the KGM using the command UNWRAP STORED BLACK KEY (n). When the loading process is complete, the KGM will respond with a "Ready" status message.

The Random Seed key, used internally in the random number generation process, can be loaded over the fill port using the command LOAD RANDOM SEED KEY, or via the host port in encrypted form using the command UNWRAP RANDOM SEED KEY. When the loading process is complete, the KGM will respond with a "Ready" status message.

5.3.2. REMOTE REKEYING

An encrypted key can be delivered to the host over the communications link, and can be decrypted for use by the KGM. This is accomplished by the host issuing either the command DECRYPT REMOTE KEY (k,n1) or UNWRAP BLACK OTAR KEY (n), where k is the number of Traffic keys included in the message and n1 is the starting location for the sequential storage of the Traffic keys, and where (n) is the location where the one RED Traffic key is to be stored in the KGM. The command DECRYPT REMOTE KEY (k,n1) is used when both a new Unique and a new Traffic key are included in the rekey message. The command UNWRAP BLACK OTAR KEY (n) is used when only a new Traffic key is included in the rekey message. In this case the KGM will automatically update the Unique as part of the command process.

If a host/KGM system is configured as a System Manager, it can also initiate rekey messages as described in Section 5.5. As a Node, it can only decrypt remote key messages.

5.3.3. KEY UPDATING

The update process is invoked by the host by sending the command UPDATE RED KEY (n) to the KGM. The KGM executes the command as outlined in the Command Processes Section. The updated key, its other key packet information, and incremented update count are stored in RAM in the same location (n). The MI's in location (n) are zeroed out. A key can be upated no more than 255 times.

5.3.4. KEY STORAGE AND RECALL

For applications with large numbers of keys, it is impractical to load all the keys from the fill port every time the KGM is to be used. The Storage key can be used to encrypt these keys for storage in the host; and subsequently decrypt them for active use in the KGM. However, the Storage key can be used to encrypt keys only when a system employs a CIK. The KGM allows decryption of stored BLACK keys whether or not a CIK is employed.

The Storage key is initially loaded over the fill port using the LOAD RED KEY (2) command. To store a key packet, the host sends the command WRAP RED KEY FOR STORAGE (n) to the KGM. The KGM executes the command and outputs a 64-bit MI plus 256 bits of encrypted key packet to the host. A similar process would take place to decrypt encrypted key packets stored in the host for active use in the KGM. From the Executive, the KGM would accept the command UNWRAP STORED BLACK KEY (n). The KGM would execute the command, check the key packet parity and key parity, check the key type, and store the key packet in storage location (n). Up to 253 Traffic keys can be stored actively in the KGM. All keys are zeroized when the KGM is turned off or power is lost.

The KGM is configured for CIK option via proper connection of the CIK OPT pin. If configured to use a CIK, the CIK is manually inserted and the host issues a command to the KGM, ACTIVATE CIK. When the CIK sequence is completed, the KGM responds to the host with "READY". Stored keys can then be unwrapped. If the KGM encounters a problem while executing the ACTIVATE CIK command, it will respond with an error status message. Note that if the CIK is removed from the system, all keys will be zeroized and the KGM will be rendered inoperable until the CIK is replaced.

5.4. BYPASS

Bypass as used here is concerned only with sending words from the host to the I/O without encryption. Both command words and data words can be sent to the I/O. These are separately controlled within the KGM. A Bypass Control Word is originally loaded into the KGM in encrypted form over the host port. The KGM decrypts it and stores it within the KGM. The Bypass Control Word dictates to the KGM the allowable number of words of data and command bypass, and the minimum number of encrypted words which must be sent out between bypass commands. Once loaded, the Bypass Control Word is stored in the KGM in non-volatile storage until modified by loading a new word.

To send commands to the I/O, COMMAND BYPASS (n) is used (n is the specified number of words to be bypassed). Each time COMMAND BYPASS is received by the KGM, it checks the Bypass Control Word to assure that the command bypass requested by the host is allowable.

The host can also command data words to be bypassed to the I/O. To do this the host commands bypass (n) words of data via the DATA BYPASS (n) command. When this command is sent to the KGM, the Bypass Control Word will be checked to assure that the requested bypass is allowable.

Upon power up, reset, restart, hardware zeroize or tamper, a sufficient number of encrypted words must be processed prior to initiating a data or command bypass command. The number of encrypted words must be greater than or equal to that specified by the Bypass Control Word.

The KGM also contains data bypass and command bypass counters. These counters are incremented once for each word bypassed. The storage process used within the KGM periodically consolidates the bypass counters. This consolidation process requires up to 2 seconds to accomplish. The process occurs at each power-up, after a RESTART command and when a Bypass Audit Word is requested. The consolidation process also occurs depending on how much bypass is performed. If bypass is performed during every 32 second period, the consolidation process will occur every 8.5 hours. If no bypass occurs, the process never occurs. The KGM produces the 32 second time period to maintain a periodic check of the bypass counters. If the application system will be hampered by a 2 second wait when it may be performing an operation, the host should purposely cause the consolidation to occur every 8 hours of continuous operation by using one of the commands listed above (e.g. RESTART). To avoid potential errors in the bypass audit count the host should wait a minimum of 32 seconds after the last bypass command before power-down of the KGM. If this is not done, it is possible that the bypass count odometer may be a larger or smaller value than it should be. The counter values, along with the Bypass Control Word, and a flag which indicates if the KGM is configured as a System Manager or a Node, can be encrypted and sent out to the host by the command READ BYPASS AUDIT. These encrypted counter and Bypass Control Word values can be used for an audit of the system containing the KGM.

5.5. SYSTEM MANAGER/NODE

The KGM can be configured as a Node or as a System Manager. The System Manager has all the capabilities of the Node plus it can generate Remote Rekey messages, generate Bypass Control Words and read Bypass Audit Words. The configuration is set by connecting the SYSTEM MGR* pin to ground (System Manager) or 5V (Node). When configured as a System Manager, the KGM will accept loading of Key Encryption keys (Unique or Storage keys), Traffic keys, or Random Seed keys into storage RAM locations 3-255. Key Encryption keys can be the Unique keys or Storage keys of the Nodes in the System Manager's system. The key type designation in their packets differentiates them from Traffic keys

To prepare a Remote Rekey message, either the command PREPARE REMOTE REKEY MESSAGE CONTAINING (k) TRAFFIC KEYS BEGINNING AT (n1) AND NEW KEY ENCRYPTION KEY (j), ENCRYPTED USING KEY ENCRYPTION KEY (m) or PREPARE REMOTE REKEY MESSAGE CONTAINING KEY (n) ENCRYPTED USING KEY ENCRYPTION KEY (m) is used. If the former command is used, the KGM sends the host the MI for the rekey message; and the new Unique (Key Encryption key) key and parity, and the new Traffic keys and parity, encrypted using the current Unique (Key Encryption key) for the Node. If the latter command is used, the KGM sends the host the MI for the rekey message; and the new key and parity encrypted using the current Unique (Key Encryption key) for the Node. The System Manager host can then send the rekey message to the Node, embedded in normal traffic, encrypted using a Traffic key. The host does this by sending the encrypted rekey message to the KGM for encryption just as any plain text would be sent to the KGM.

The System Manager can also wrap Random Seed keys for distribution to Nodes. Since the System Manager has access to and can store Random Seed keys and Storage keys in active storage, the normal wrap and store process can be performed.

The System Manager can also prepare Bypass Control Words for Nodes in the system. These are prepared using the command ENCRYPT BYPASS CONTROL WORD USING KEY ENCRYPTION KEY (m) followed by a Bypass Control Word. When this command is complete, the KGM sends the host the MI and the encrypted Bypass Control Word. The System Manager host can then send this message to the Node, embedded in normal traffic, encrypted using a Traffic key. Note that the default mode for the KGM without a valid Bypass Control Word is no bypass for command or data.

Bypass Audit Words can be requested by the System Manager from Nodes in the System. Upon request by the System Manager, the node host would use the command READ BYPASS AUDIT, to command the KGM to produce a Bypass Audit Word. This word is encrypted in the Node's Unique; and contains "odometer" readings of the number of command and data words bypassed, the Bypass Control Word currently in use by the Node, and the state of the SYSTEM MGR* pin to tell if the Node is configured as a System Manager or a Node. The Node can send this encrypted message to the System Manager over normal traffic channels. The System Manager can decrypt the audit message using the DECRYPT BYPASS AUDIT WORD USING KEY ENCRYPTION KEY (m) command. The System Manager can then compare the Bypass Control Word currently in use at the Node with the one previously sent. The "odometer" readings for data and command bypass can be compared with those expected, based on current usage. The reading of the bypass audit is provided for use in a system where additional security and concern about unauthorized bypass is present. If the bypass audit information is critical, the KGM should not be shut off until at least 32 seconds have elapsed from the last bypass command.

All other capabilities of the System Manager are identical to the Node configuration.

5.6. ZEROIZATION

Zeroization can be accomplished using either a command or by direct hardware action. Commands can be used to selectively zeroize key storage locations in RAM (ZEROIZE (n)) or more completely zeroize the KGM by zeroizing all key storage locations throughout the KGM (ZEROIZE ALL). Zeroization commands are accomplished by overwriting the applicable locations with zeros.

Hardware zeroize is initiated by bringing the ZEROIZE* pin on the KGM to ground. This zeroizes all internal RED key storage. Bringing the RESET* pin to ground or breaking the tamper loop has the same affect as the hardware zeroize.

5.7. RESET/RESTART

Reset is accomplished either through firmware or hardware. As in the zeroize, each is different. The host command RESTART causes the KGM to reset to an initial condition but does not affect RED key storage in the KGM.

A hardware reset is initiated by grounding the KGM RESET* pin. This causes the same events to occur as would occur from power up, including zeroizing all internal RED key storage.

5.8. SELF TEST

Upon power up or upon recovery from a low power detection, tamper detection, hardware reset, or hardware zeroize, the KGM will respond with the status message "Ready for Self Test", and await a SELF TEST command from the host. Self test must be successfully completed before any other command processes will be executed. Self test does an extensive check of the internal operation of the KGM including a cryptographic alarm check. As part of the self test process, the host, I/O, CIK and fill ports are checked by writing data out and reading it back from those ports. The host must place the host and I/O busses in a high impedance state before commanding self test. The host port and the I/O port must both have pull-up resistors (See Section 7). The I/O busy lines BSYi and BSYo must remain low and SRQi* must be high. During the self test process the alarm pin ALARM* will be driven low indicating an alarm. This is normal during the self test process. At the conclusion of this check, the KGM issues a "READY" status message to the host and awaits a command.

6. KGM COMMANDS

6.1. COMSEC COMMAND LANGUAGE (CCL)

The CCL has been developed to support NSA's standard KGM line of embeddable COMSEC modules. This language is used throughout the family where common functionality exists. This method has been selected to enhance the structured nature of the KGM line.

All commands are comprised of one operator followed by 0 to n operand words. Operand words are free format. The command operator format is as follows:

bit	7	6 5 4	3 2 1 0 LSB
FIELD	PARITY	CLASS	TYPE

All operators are adjusted for odd parity. Operands have no parity bit.

There are eight possible classes of commands. Each class is assigned as follows:

CLASS	DESCRIPTION
000	Reserved for Future Use
001	General Purpose Comands
010	Key Processing Commands
011	Data Processing Commands
100	Special Purpose Commands
101	Special Purpose Commands
110	Special Purpose Commands
111	Special Purpose Commands

6.2. HOST/KGM Command Format

Commands issued by the host to the KGM consist of an 8-bit CCL operator byte followed by 0 to n 16-bit operand words. Possible operands in the KGM commands set are: n, n1, n2, n3, k, j, m, and data words.

When the KGM is configured in the 16-bit mode (either by default or by issuing the SET 16-BIT MODE command) a host to KGM command is formatted as follows:

CCL Operator - 1st host to KGM transfer

(CMD* = 0) Data -	XXH	CCL Operator
Host Data Line -	h15 ... h8	h7 ... h0

Operand #1 (if required) - 2nd host to KGM transfer

(CMD* = 1) Data -	Operand #1 Most Significant Byte	Operand #1 Least Significant Byte
Host Data Line -	h15 ... h8	h7 ... h0

Operand #2 (if required) - 3rd host to KGM transfer

(CMD* = 1) Data -	Operand #2 Most Significant Byte	Operand #2 Least Significant Byte
Host Data Line -	h15 ... h8	h7 ... h0

Following Operands, if required are the same format.

X = Don't Care

When the KGM is configured in the 8-bit mode (as a result of the SET 8-BIT MODE command) a host to KGM command is formatted as follows:

CCL Operator - 1st host to KGM transfer

(CMD* = 0) Data -	XXH			CCL Operator			
Host Data Line -	h15	...	h8	h7	...	h0	

Operand #1 (if required) - 2nd host to KGM transfer

(CMD* = 1) Data -	XXH			Operand #1 Least Significant Byte		
Host Data Line -	h15	...	h8	h7	...	h0

3rd host to KGM transfer

(CMD* = 1) Data -	XXH			Operand #1 Most Significant Byte		
Host Data Line -	h15	...	h8	h7	...	h0

Operand #2 (if required) - 4th host to KGM transfer

(CMD* = 1) Data -	XXH			Operand #2 Least Significant Byte		
Host Data Line -	h15	...	h8	h7	...	h0

5th host to KGM transfer

(CMD* = 1) Data -	XXH			Operand #2 Most Significant Byte		
Host Data Line -	h15	...	h8	h7	...	h0

Following Operands, if required, are the same fomat.

X - Don't Care, these data lines are not used in the 8-bit mode.

An example would be the UNWRAP STORED BLACK KEY (n) command. The command format is 26H, n, 320-bits.

To decrypt the 320-bit stored key packet and retain as key in location AAH (n=AA) when configured in the 16-bit mode the command sequence would be:

1st host to KGM transfer

OPERATOR - UNWRAP STORED BLACK KEY.

(CMD* = 0) Data -	00H	26H
Host Data Line -	h15 ... h8	h7 ... h0

2nd host to KGM transfer

OPERAND - n

(CMD* = 1) Data -	00H	AAH
Host Data Line -	h15 ... h8	h7 ... h0

3rd host to KGM transfer

OPERAND - data bits

(CMD* = 1) Data -	Bit 15 ... Bit 8	Bit 7 ... Bit 0
Host Data Line -	h15 ... h8	h7 ... h0

4th host to KGM transfer

OPERAND - data bits

(CMD* = 1) Data -	Bit 31 ... Bit 24	Bit 23 ... Bit 16
Host Data Line -	h15 ... h8	h7 ... h0

5th through 21st host to KGM transfers

OPERANDS - data bits

22nd host to KGM transfer

OPERAND - data bits

(CMD* = 1) Data -	Bit 319	...	Bit 312	Bit 311	...	Bit 304
Host Data Line -	h15	...	h8	h7	...	h0

(20) 16-bit data words = 320 bits

To decrypt the 320-bit stored key packet and retain as key in location AAH when configured in the 8-bit mode the command sequence would be:

1st host to KGM transfer

OPERATOR - UNWRAP STORED BLACK KEY

(CMD* = 0) Data -		XXH			26H	
Host Data Line -	h15	...	h8	h7	...	h0

2nd host to KGM transfer

OPERAND - n (least significant byte)

(CMD* = 1) Data -		XXH			AAH	
Host Data Line -	h15	...	h8	h7	...	h0

3rd host to KGM transfer

OPERAND - n (most significant byte)

(CMD* = 1) Data -		XXH			00H	
Host Data Line -	h15	...	h8	h7	...	h0

4th host to KGM transfer

OPERAND - data bits

(CMD* = 1) Data -		XXH		Bit 7	...	Bit 0
Host Data Line -	h15	...	h8	h7	...	h0

5th host to KGM transfer

OPERAND - data bits

(CMD* = 1) Data -		XXH		Bit 15	...	Bit 8
Host Data Line -	h15	...	h8	h7	...	h0

6th through 42nd host to KGM transfers

OPERANDS - data bits

43rd host to KGM transfer

(CMD* = 1) Data -		XXH		Bit 319	...	Bit 312
Host Data Line -	h15	...	h8	h7	...	h0

X - Don't Care, these data lines are not used in the 8-bit mode.

(40) 8-bit data bytes = 320 bits

6.3. HOST/KGM COMMAND PROTOCOL

This section explains how the host writes any command to the KGM. This sequence is repeated at the beginning of each command to follow, this will be referred to as "Host/KGM Command Protocol":

1. **The host checks the status of BSYw. If not active the host can write a command to the KGM. Note: if BSYw = 1, the host can still write a command to the KGM, however, if data is in the KGM host port input buffer, it will be lost.**

2. **The host sets CS* low, places the command operator byte on the bus, lines h0 - h7, and;**

3. **Sets CMD* to zero.**

4. At this point the WR* signal is toggled and the command operator will be latched into the KGM on the rising edge of WR*.

5. On the rising edge of WR*, BSYw will go active. This indicates that the host port input buffer is full and/or command processing is occurring. BSYw will then go inactive telling the host processor that an operator may be written if required. This process is repeated until all of the operands have been latched into the command processor. When the last operand is latched into the KGM, as determined by the operator byte evaluation, the command process is verified and if valid, performed. Note that only the operator byte is sent to the KGM with CMD* = 0. All operands are sent to the KGM with CMD* = 1.

6. When the command process is complete, BSYw and BSYr will go inactive and the KGM will assert SRQ* indicating that a status message is in the host output buffer (for all commands except traffic commands). In the case of a traffic command, the KGM will make BSYw inactive as soon as the KGM internal microcontroller reads the host port input buffer and validates the command, and will wait for data from the host, or will sense BSYo if expecting data from the I/O. For traffic commands, the KGM will not respond with a status message unless an error has occurred.

7. If an error condition was encountered during command processing, BSYr will go low and SRQ* will go active, indicating that the host processor should read the status word from the host port output buffer. SRQ* will be reset upon completion of a host read cycle.

6.4. KGM Command Set

Table 6-1 contains a list of commands recognized by the KGM. Any attempt to send commands not included in the table will result in the status message "Unrecognizable Command".

Table 6-1: COMMAND LIST

Command	Hex Code
ABORT	16H
ACTIVATE CIK	1FH
COMBINE KEYS (n1, n2, n3)	ADH
COMMAND BYPASS (n)	D5H
COPY RED KEY (n1, n2)	2CH
DATA BYPASS (n)	D3H
DECRYPT BYPASS AUDIT WORD USING KEY ENCRYPTION KEY (m)	5BH
DECRYPT REMOTE KEY (k, n1)	D6H
DECRYPT WITH RESYNC	32H
DECRYPT WITHOUT RESYNC	34H
ENCRYPT BYPASS CONTROL WORD USING KEY ENCRYPTION KEY (m)	DAH
ENCRYPT WITH RESYNC	31H
ENCRYPT WITHOUT RESYNC	B3H
END OF MESSAGE	B5H
END OF MESSAGE MIDDLE OF WORD	75H
LOAD BYPASS CONTROL WORD	D0H
LOAD RANDOM SEED KEY	70H
LOAD RED KEY (n)	A2H
MESSAGE AUTHENTICATION CODE MODE (MAC)	51H
PREPARE REMOTE REKEY MESSAGE CONTAINING (k) TRAFFIC KEYS BEGINNING AT (n1) AND NEW KEY ENCRYPTION KEY (j), ENCRYPTED USING KEY ENCRYPTION KEY (m)	D9H

Command	Hex Code
PREPARE REMOTE REKEY MESSAGE CONTAINING KEY (n),ENCRYPTED USING KEY ENCRYPTION KEY (m)	F4H
READ BYPASS AUDIT	54H
RESET RANDOM SEED KEY FLAG	F2H
RESTART	13H
SELF TEST	91H
SET DECRYPT DIRECT	DCH
SET DECRYPT VIA HOST	5EH
SET ENCRYPT PATH AS COPROCESSOR	5DH
SET ENCRYPT PATH TO OUTPUT	DFH
SET MODE (n)	B0H
SET 8-BIT MODE	57H
SET 16-BIT MODE	58H
STOP	10H
TRANSFER KEY (n)	ABH
UNWRAP BLACK OTAR KEY (n)	25H
UNWRAP RANDOM SEED KEY	F1H
UNWRAP STORED BLACK KEY (n)	26H
UPDATE RED KEY (n)	29H
VALIDATE KEY (n)	2AH
WRAP RED KEY FOR STORAGE (n)	A8H
ZEROIZE ALL	20H
ZEROIZE (n)	A1H

The remainder of this section will cover each KGM command, specifying its format, prerequisites and delay time, and giving a brief description of its operation and subsequent status messages generated by the KGM as a result of successful or aborted command execution.

Delay time is defined as the time between receipt of the command and assertion of the SRQ* to the host after the command is processed. Some of the commands for high-speed operations do not issue a status word (NO SRQ*) when completed. For these commands, only the BSYw line goes low when command execution is complete. These exceptions are noted in the command listing (no "Ready" status word is used).

All status messages are single-byte (8-bits wide) and appear on h0-h7 in both the 8- and 16-bit modes of the KGM. The following is a comprehensive list of available status messages:

HEX CODE	STATUS MESSAGE
00	Ready (Successful Operation)
01	Data Loss
02	Unrecognizable Command
03	Invalid Command
04	No Fill Device Attached
07	Key Parity Error
08	Wrong Key Type
09	Key Not Found
0A	Invalid Key
0D	Storage Key Not Found
0F	CIK Failure
10	CIK Not Inserted
11	Update Limit Reached
53	Ready For Self Test
59	Random Seed Key Required
5A	Bypass Control/Audit Word Parity Error
5C	Ready, Encrypt Resync Required
5D	Ready, Decrypt Resync Required
5E	Bypass Exceeds Limit
5F	Ready, Encrypt & Decrypt Resync Required
FF	Process Error

The individual command descriptions follow.

Command: ABORT

Format: 16H

Description: The ABORT command causes the KGM to exit from the current process and enter the Executive. Any data words which have been read into the KGM, but not yet processed, will be lost. If KGM is in traffic mode, traffic counters (i.e. words encrypted since last bypass) will be made current up to the point of the Abort. The cryptoprocessor and Key-In-Use will be cleared, and the MI's for the key in use will be lost. As a result, the ENCRYPT WITH RESYNC and DECRYPT WITH RESYNC command will be required on next use of the key. The KGM will remain in the Executive.

Prerequisites: None

Status Code:

Condition	Status Message	Format
Successful operation	"Ready"	00

Command: ACTIVATE CIK

Format: 1FH

Description: CIK process is initiated. CIK device is updated and the KGM has the RED Storage key available for decryption of stored keys or encryption of keys for storage.

Prerequisites:
1. Storage key and RSK must have been loaded previously.
2. CIK device must be attached and CIK option selected

Status Code:

Condition	Status Message	Format
KGM cannot write to CIK	"CIK Failure"	0F
CIK option not selected	"Invalid Command"	03
Random Seed key flag not set	"Random Seed Key Required"	59
Parity error in Storage key	"Key Parity Error"	07
Storage key all zeros	"Storage Key Not Found"	0D
Successful operation	"Ready"	00

Command: COMBINE KEYS (n1, n2, n3)

Format: ADH, n1, n2, n3

Description: This command will combine two keys (modulo 2 addition) to derive a third key. (n1) and (n2) are the key locations of the two keys to be combined. (n3) is the key location where the resulting key is stored. The new key will be a Traffic key.

Prerequisites:

1. KGM must have a valid key at (n1) and (n2).
2. $3 \le (n1),(n2),(n3) \le 255$

Status Code:

Condition	Status Message	Format
No key at (n1) or (n2)	"Key Not Found"	09
Parity error on key from (n1) or (n2)	"Key Parity Error"	07
Keys (n1), (n2) are not the same type	"Wrong Key Type"	08
Command received before original command complete	"Data Loss"	01
Resulting key failed activity check	"Invalid Key"	0A
3>(n1),(n2),(n3)>255	"Invalid Command"	03
Successful operation	"Ready"	00

Command: COMMAND BYPASS (n)

Format: D5H, n

Description: Instructs KGM to send (n) words of command to the I/O port without encryption. KGM checks this bypass against that allowed by Bypass Control Word. After completing the bypass the KGM returns to the Executive.

Prerequisites: KGM must have Bypass Control Word loaded.

Status Code:

Condition	Status Message	Format
Bypass request not valid (exceeds internal bypass setting)	"Bypass Exceeds Limits"	5E
Command received before original command complete	"Data Loss"	01
Successful operation	"Ready"	00

Command: COPY RED KEY (n1, n2)

Format: 2CH, n1, n2

Description: Red key packet in location (n1) is copied to location (n2). The encrypt and decrypt MI's in (n1) remain unchanged but are not copied to (n2). The MI's in (n2) will both be zero as a result of this command. If the KGM is configured as a System Manager, (n1) and (n2) can be RAM locations 1 to 255, but locations 1 and 2 are restricted to Key Encryption keys. If the KGM is not configured as a System Manager, (n1) and (n2) are restricted to RAM locations 3 to 255.

Prerequisites:
1. Valid key must be loaded in (n1).
2. (n1) and (n2) can be RAM locations 1 to 255 if configured as a System Manager, 3 to 255 if otherwise.

Status Code:

Condition	Status Message	Format
3>n1>255 if configured as node	"Invalid Command"	03
1>n1>255 if configured as system manager	"Invalid Command"	03
Parity error on key from location (n1)	"Key Parity Error	07
Key from location n1 is not valid for location n2	"Wrong Key Type"	08
Command received before original command complete	"Data Loss"	01
Successful operation	"Ready"	00

Command:	DATA BYPASS (n)	
Format:	D3H, n	
Description:	Instructs KGM to allow (n) words of data to pass without encryption from host port to I/O port. KGM certifies bypass request against that allowed by the Bypass Control Word. After completing the bypass the KGM returns to the Executive.	
Prerequisites:	Bypass Control Word must be present.	

Status Code:	Condition	Status Message	Format
	Bypass request not valid (exceeds internal bypass setting)	"Bypass Exceeds Limit"	5E
	Command received before original command complete	"Data Loss"	01
	Successful operation	"Ready"	00

Command: DECRYPT BYPASS AUDIT WORD USING KEY ENCRYPTION KEY (m)

Format: 5BH, Encrypted Bypass Audit Word (320 bits), m

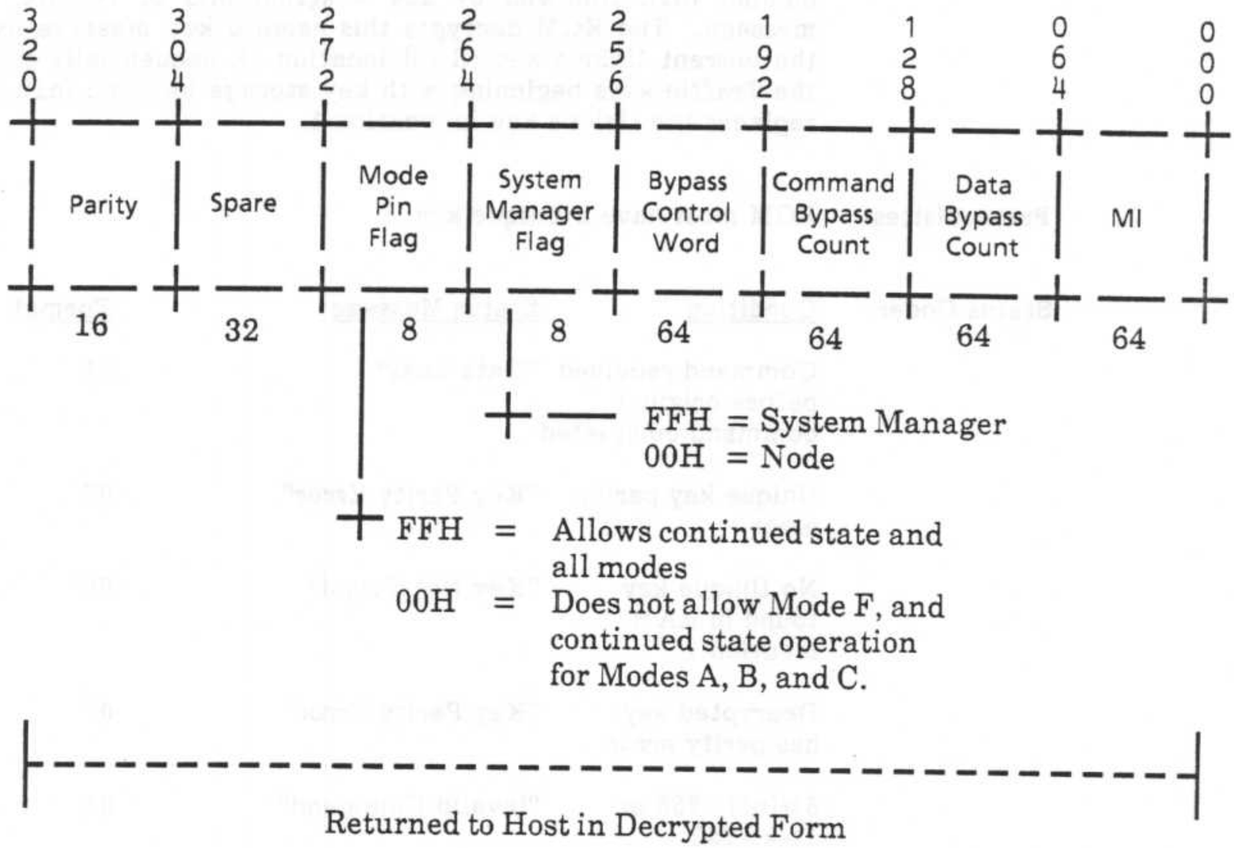
Description: The purpose of this command is to permit the System Manager to verify that the Bypass Control Word, stored at a Node in the system, is correct. In addition, the accumulated number of command and data bypass, and the state of the System Manager and Mode pins, are decrypted for use by the System Manager. As a result of this command, 256 bits of Bypass Audit Word are returned to the host.

Prerequisites:
1. Must be configured as a System Manager.
2. Valid Key Encryption key must be in location (m).

Status Code:

Conditions	Status Message	Format
KGM not configured as System Manager	"Invalid Command"	03
3>(m)>255	"Invalid Command"	03
Key (m) has parity error	"Key Parity Error"	07
No key in location (m) or key is not Key Encryption key	"Wrong Key Type"	08
Invalid key	"Invalid Key"	0A
Command received before original command complete	"Data Loss"	01
Bypass Audit Word parity error	"Bypass Control/ Audit Word Parity Error"	5A
Successful operation	"Ready"	00

ENCRYPTED BYPASS AUDIT WORD FORMAT



Command:	DECRYPT REMOTE KEY (k,n1)
Format:	D6H, k, n1, [224 + (k)(160)] Bits Remote Key
Description:	Instructs KGM to decrypt k remote Traffic keys and a remote Unique key; and retain the Traffic keys in sequential locations starting with (n1) and the Unique key in location 1. The command contains number of Traffic keys (k), location number (n1), followed by 224 + k(160) bits of remote key message. The KGM decrypts this remote key message using the current Unique key (RAM location 1), sequentially stores the Traffic keys beginning with key storage location (n1), and replaces the Unique key in location 1.
Prerequisites:	KGM must have a Unique key.

Status Code:

Condition	Status Message	Format
Command received before original command completed	"Data Loss"	01
Unique key parity error	"Key Parity Error"	07
No Unique key found in RAM location 1	"Key Not Found"	09
Decrypted key has parity error	"Key Parity Error"	07
3>(n1)>255 or 1>k>253	"Invalid Command"	03
n1+(k-1)>255	"Invalid Command"	03
Invalid key	"Invalid Key"	0A
Successful operation	"Ready"	00

Command:	DECRYPT WITH RESYNC
Format:	32H
Description:	Instructs the KGM to enter the traffic mode of operation wherein the KGM will read data from either the host port (following a SET DECRYPT VIA HOST command) or I/O port (following a SET DECRYPT DIRECT command); decrypt that data using the cryptographic processor specified by the last SET MODE (n) command; and send the decrypted data to the host port. This command is used to establish synchronization with the encrypter and causes the KGM to read a 64-bit MI from the data input port immediately prior to the first word of cipher text. If issued in the Executive, this comamnd must have been preceded by a TRANSFER KEY (n) command. If issued while in the traffic mode, the current key-in-use will remain in use. Once this command is issued, the KGM will remain in the traffic mode until some other command is issued.
Prerequisites:	Key-In-Use must be a valid Traffic key.

Status Code:	Condition	Status Message	Format
	No key number in Key-In-Use register	"Key Not Found"	09
	Key-In-Use is not a Traffic key	"Wrong Key Type"	08

Command: DECRYPT WITHOUT RESYNC

Format: 34H

Description: Instructs the KGM to enter the traffic mode of operation wherein the KGM will read data from either the host port (following a SET DECRYPT VIA HOST command) I/O port (following a SET DECRYPT DIRECT command); decrypt that data using the cryptographic process specified by the last SET MODE (n) command; and send the decrypted data to the host port. This command is used after synchronization with the encrypter has been established by a prior use of the DECRYPT WITH RESYNC command and causes the KGM to start processing traffic immediately without reading in a 64-bit MI. If issued in the Executive,this command must have been preceded by a TRANSFER KEY (n) command. If issued while in the traffic mode, the current key-in-use will remain in use. Once this command is issued, the KGM will remain in the traffic mode until some other command is issued. This command is not allowed for Modes A, B, or C if MODE* pin is +5V.

Prerequisites:

1. Key-In-Use must be a valid Traffic key.
2. Key must have been synchronized with the encrypter via a prior DECRYPT WITH RESYNC.
3. MODE* pin must be grounded if Mode A, B, or C is selected.

Status Code:

Condition	Status Message	Format
No key number in Key-In-Use register	"Key Not Found"	09
Key-In-Use is not a Traffic key	"Wrong Key Type"	08
The key in use has not yet been used with a DECRYPT WITH RESYNC command or MODE* pin not grounded and MODE A, B, or C is selected.	"Invalid Command"	03

Command: ENCRYPT BYPASS CONTROL WORD USING KEY ENCRYPTION KEY (m)

Format: DAH, Bypass Control Word (48 bits), m

Description: The purpose of this command is to prepare a Bypass Control Word, via encryption, to be sent to a Node on the system. This Node then uses the Bypass Control Word to control bypass of data and commands. As a result of this command, 128 bits of MI and encrypted Bypass Control Word are returned to the host.

Prerequisites:
1. Must be configured as a System Manager.
2. Valid Key Encryption key must be in location (m).

Status Code:

Conditions	Status Message	Format
KGM not configured as System Manager	"Invalid Command"	03
3>(m)>255	"Invalid Command"	03
Key (m) has parity error	"Key Parity Error"	07
No key in location (m) or key (m) is not a key Encryption key	"Wrong Key Type"	08
Invalid key	"Invalid Key"	0A
Command received before original command complete	"Data Loss"	01
Random Seed key flag not set	"Random Seed Key Required"	59
Successful operation	"Ready"	00

BYPASS CONTROL WORD FORMAT

```
                                     Supplied by Host
                  |-----------------------------------------------------|
0                 0                 0        0                 0      0
6                 4                 3        2                 0      0
4                 8                 2        4                 8      0
+-----------------+-----------------+--------+-----------------+------+
|                 |                 |        |                 |      |
|                 |                 |        |                 |      |
+-----------------+-----------------+--------+-----------------+------+
        |16                |16          |8            |16           |8
Parity  |                  |            |             |             |
for     |                  |            |             |             |
Bypass  |                  |            |             |             |
Control |                  |            |             |             |
Word    |                  |            |             |             |
--------+                  |            |             |             |
                           |            |             |             |
Minimum number of encrypted|            |             |             |
words which must be sent   |            |             |             |
out between COMMAND BYPASS |            |             |             |
(n) commands               |            |             |             |
---------------------------+            |             |             |
                                        |             |             |
        Maximum number of words of      |             |             |
        Command Bypass words allowed    |             |             |
        by a COMMAND BYPASS (n)         |             |             |
        command                         |             |             |
        --------------------------------+             |             |
                                                      |             |
                Minimum number of encrypted           |             |
                words which must be sent              |             |
                out between DATA BYPASS (n)           |             |
                commands                              |             |
                --------------------------------------+             |
                                                                    |
                             Maximum number of words of             |
                             data bypass words allowed              |
                             by a DATA BYPASS (n)                   |
                             command                                |
                             ---------------------------------------+
```

Command: ENCRYPT WITH RESYNC

Format: 31H

Description: Instructs the KGM to enter the traffic mode of operation wherein the KGM will read data from the host port; encrypt that data using the cryptographic process specified by the last SET MODE (n) command; and send the encrypted data to either the I/O port (following a SET ENCRYPT PATH TO OUTPUT command) or the host port (following a SET ENCRYPT PATH AS COPROCESSOR command). The command is used to establish synchronization with the decrypter and causes the KGM to send out a 64-bit MI immediately prior to the first word of cipher text. If issued in the Executive, this command must have been preceded by a TRANSFER KEY (n) command. If issued while in the traffic mode, the current key-in-use will remain in use. Once this command is issued, the KGM will remain in the traffic mode until some other command is issued.

Prerequisites: Key-In-Use must be a valid Traffic key.

Status Code:

Condition	Status Message	Format
No key number in Key-In-Use register	"Key Not Found"	09
Key-In-Use is not a Traffic key	"Wrong Key Type"	08

Command: ENCRYPT WITHOUT RESYNC

Format: B3H

Description: Instructs the KGM to enter the traffic mode of operation wherein the KGM will read data from the host port; encrypt that data using the cryptographic process specified by the last SET MODE (n) command; and send the encrypted data to either the I/O port (following a SET ENCRYPT PATH TO OUTPUT command) or the host port (following a SET ENCRYPT PATH AS COPROCESSOR command). This command is used after synchronization with the decrypter has been established by a prior use of the ENCRYPT WITH RESYNC command and causes the KGM to start processing traffic immediately without sending out the 64-bit MI. If issued while in the traffic mode, the current key-in-use will remain in use. Once this command is issued, the KGM will remain in the traffic mode until some other command is issued. This command is not allowed for Modes A, B, or C if MODE* pin is +5V.

Prerequisites:
1. Key-In-Use must be a valid Traffic key.
2. Key must have been synchronized with the decrypter via a prior ENCRYPT WITH RESYNC.
3. MODE* pin grounded (if Mode A, B, or C selected).

Status Code:

Condition	Status Message	Format
No key number in Key-In-Use register	"Key Not Found"	09
Key-In-Use is not a Traffic Key	"Wrong Key Type"	08
The key in use has not yet been used with an ENCRYPT WITH RESYNC command or MODE* pin not grounded and Modes A, B, or C selected.	"Invalid Command"	03

Command:	END OF MESSAGE
Format:	B5H
Description:	This command signals the KGM that there is not more data in the current message The KGM encrypts/decrypts any traffic data already read and waiting to be processed and continues to wait for the first data word of the next message. This command is meaningful only when the KGM is processing traffic data or a MAC, and the actual process performed in response to this command will vary depending on the partial-frame processing requirements of the encryption/decryption mode being executed.
Prerequisites:	KGM must be processing traffic or MAC.

Status Code:	Condition	Status Message	Format
	KGM is not in traffic mode or MAC	"Unrecognizable Command"	02
	Successful operation traffic mode	No Status Message	
	Successful operation MAC mode	"Ready"	00

Command:	END OF MESSAGE MIDDLE OF WORD
Format:	75H
Description:	Designates the end of the plain text or cipher text message, when the last word in the message has only bits 0-7 which are valid. This command is used to end a message in 16-bit mode, when an odd byte is present. It will use only bits 0-7 of the last 16-bit data word sent to the KGM (the host should always set bits 8-15 to zero in this word). As a result of this command the KGM will process the words that have already been read into the KGM (including bits 0-7 only of the last word) and will remain in the current configuration (either Encrypt or Decrypt) and await the next message.
Prerequisites:	KGM must be processing traffic.

Status Code:	Condition	Status Message	Format
	KGM is not in traffic mode	"Unrecognizable Command"	02

Command:	LOAD BYPASS CONTROL WORD
Format:	D0H, 128 bits
Description:	Instructs KGM to load Bypass Control Word from host port. KGM decrypts the Bypass Control Word using the Unique key in RAM location (1). The decrypted output is then stored in the KGM in non volatile memory.
Prerequisites:	Unique Key must be present in RAM location (1)

Status Code:

Condition	Status Message	Format
Command received before original command complete	"Data Loss"	01
Unique key not found	"Key Not Found"	09
Unique key has parity error	"Key Parity Error"	07
Parity error in Bypass Control Word	"Bypass Control/Audit Word Parity Error"	5A
Invalid key	"Invalid Key"	0A
Successful operation	"Ready"	00

Command: LOAD RANDOM SEED KEY

Format: 70H

Description: This command is issued to the KGM whenever a new Random Seed key (RSK) is to be loaded. The Random Seed key is used in the random number generation process within the KGM. The KGM accepts the key over the fill port.

Prerequisites: Fill device must be attached to the fill port.

Status Code:

Condition	Status Message	Format
Parity error or checkword test failed in key from fill port	"Key Parity Error"	07
No fill device attached	"No Fill Device Attached"	04
Wrong key type	"Wrong Key Type"	08
Less than 256 bits loaded In	"Process Error"	FF
Invalid key	"Invalid Key"	0A
Successful operation	"Ready"	00

Command: LOAD RED KEY (n)

Format: A2H, n

Description: Instructs KGM to load RED key packet from fill port to location (n). As a result of this process, a RED key packet is present in RAM location (n). The MI's for location (n) are both zero.

Prerequisites: Fill device must be attached to fill port.

Status Code:

Condition	Status Message	Format
No fill device attached	"No Fill Device Attached"	04
Key parity error or checkword test failed in loaded RED key	"Key Parity Error"	07
Wrong key type for location (n) or key has wrong algorithm type	"Wrong Key Type"	08
1>n>255	"Invalid Command"	03
Invalid key	"Invalid Key"	0A
Random Seed key flag not set	"Random Seed Key Required"	59
Less than 256 bits loaded In	"Process Error"	FF
Command received before original command complete	"Data Loss"	01
KGM cannot write to CIK	"CIK Failure"	0F
Successful operation	"Ready"	00

Command: MESSAGE AUTHENTICATION CODE MODE (MAC)

Format: 51H

Description: Instructs the KGM to prepare a Message Authentication Code of 64 bits by cryptographic processing of data from the host port.

Prerequisites:
1. KGM must be in Executive, Encrypt or Decrypt.
2. Key-In-Use must be a valid Traffic key.

Status Code:

Condition	Status Message	Format
No key number in Key-In-Use register	"Key Not Found"	09
Key-In-Use not Traffic Key	"Wrong Key Type"	08
Successful operation	"Ready"	00

Command:	PREPARE REMOTE REKEY MESSAGE CONTAINING k TRAFFIC KEYS BEGINNING AT (n1) AND NEW KEY ENCRYPTION KEY (j), ENCRYPTED USING KEY ENCRYPTION KEY (m)
Format:	D9H, n1, m, j, k
Description:	This command is used to prepare a remote rekey message containing multiple Traffic keys and a new Key Encryption key all encrypted in the present Key Encryption key. The receiving Node will decrypt the message using its Unique key and store the new Key Encryption key as its new Unique and store the new Traffic keys in locations beginning at (n1) going to (n1 + k-1). A total of 224 + k(160) bits will be sent.
Prerequisites:	1.KGM must be configured as a System Manager (SYSTEM MGR* pin =0). 2. Valid Traffic keys must be present in locations (n1) to (n1 + k-1). 3. Valid Key Encryption key must be in location (m). 4. Valid Key Encryption key must be in location (j).

Status Code:

Condition	Status Message	Format
KGM not configured as System Manager	"Invalid Command"	03
Illegal key location (256-k<n1)	"Invalid Command"	03
Key parity error for key from (m), (j) or any key from (n1) thru (n1 + k-1)	"Key Parity Error"	07
Invalid Key	"Invalid Key"	0A
No Key in location (n1) thru (n1 + k-1) or not a Traffic key	"Wrong Key Type"	08
Key in locations (m) or (j), or not a Key Encryption key	"Wrong Key Type"	08

Random Seed key flag not set	"Random Seed Key Required"	59
Command received before original command complete	"Data Loss"	01
3>m>255 or 1>j>255 or 1>k>253 or 3>n1>255	"Invalid Command"	03
Successful operation	"Ready"	00

Command:	PREPARE REMOTE REKEY MESSAGE CONTAINING KEY (n) ENCRYPTED USING KEY ENCRYPTION KEY (m)
Format:	F4H, n, m
Description:	This command is used to prepare a remote rekey message containing a key packet encrypted in the present Key Encryption key. The receiving Node will decrypt the message using its Unique key, store the new key packet, and update its Unique key. A total of 320 bits will be output by the KGM as a result of this command.
Prerequisites:	1. KGM must be configured as a System Manager (SYSTEM MGR* = 0) 2. Valid key must be present in location (n). 3. Valid Key Encryption key must be in location (m).

Status Code:

Condition	Status Message	Format
KGM not configured as System Manager	"Invalid Command"	03
Command received before original command complete	"Data Loss"	01
3>(m)>255 or 3>(n)>255	"Invalid Command"	03
Key parity error for key in (n) or (m).	"Key Parity Error"	07
No key in designated location (m) or not Key Encryption key	"Wrong Key Type"	08
Random Seed key flag not set	"Random See Key Required"	59
Invalid key	"Invalid Key"	0A
Successful operation	"Ready"	00

Command:	READ BYPASS AUDIT
Format:	54H
Description:	Instructs the KGM to encrypt the accumulated bypass count, the Bypass Control Word, and the System Manager/Node and Mode pin settings. Following receipt of the "Ready" status message by the host, the KGM will send this 320-bit encrypted audit message to the host.
Prerequisites:	KGM Must Have Unique Key.

Status Code:

Condition	Status Message	Format
No Unique key	"Key Not Found"	09
Unique key has parity error	"Key Parity Error"	07
Command received before original command complete	"Data Loss"	01
Invalid key	"Invalid Key"	0A
Successful operation	"Ready"	00

Command:	RESET RANDOM SEED KEY FLAG
Format:	F2H
Description:	The Random Seed key flag is set whenever a valid Random Seed key has been loaded into the KGM. The RESET RANDOM SEED KEY FLAG command is used to reset the flag to indicate that a new Random Seed key needs to be loaded.
Prerequisites:	None

Status Code:

Condition	Status Message	Format
New Random Seed key needs to be loaded	"Random Seed Key Required"	59

Command:	RESTART		
Format:	13H		
Description:	The RESTART command returns the KGM to an initial state with all microcontroller registers and cryptoprocessor registers zeroized. This command does not affect the contents of key RAM locations, unlike a hardware reset (bringing the RESET* pin low and high again).		
Prerequisites:	None		
Status Code:	<u>Condition</u>	<u>Status Message</u>	<u>Format</u>
	Successful operation	"Ready"	00

Command: SELF TEST

Format: 91H

Description: Checks functionality of KGM hardware and all alarms. Failure during self-test puts the KGM into an alarm condition or results in sending a status message to the host.
CAUTION: The ALARM*, h0-h15, CSo*, CMDo*, i0-i15, WRo*, REQ, SK, DATA OUT, and CSc are exercised during normal operation of this command.

Prerequisites: None

Status Code:

Condition	Status Message	Format
Random Seed key flag not set	"Random Seed Key Required"	59
Successful operation	"Ready"	00

Command: SET DECRYPT DIRECT

Format: DCH

Description: This command will affect the configuration of the KGM in both decryption and encryption traffic modes. Once this command has been executed, the KGM will read the data to be decrypted from the I/O port and will write the decrypted data to the host port. During encryption, the KGM will read data to be encrypted from the host port and will write the encrypted data to the I/O port. This is the normal (defualt on power-up) configuration.

Prerequisites: None

Status Code:

Condition	Status Message	Format
Successful operation	"Ready"	00

Command:	SET DECRYPT VIA HOST
Format:	5EH
Description:	This command will affect the configuration of the KGM in both decryption and encryption traffic modes. Once this command has been executed, the KGM will read the data to be decrypted from the host port and will write the decrypted data back to the host port. During encryption, the KGM will read data to be encrypted from the host port and will write the encrypted data to the I/O port.
Prerequisites:	None

Status Code:	Condition	Status Message	Format
	Successful operation	"Ready"	00

Command:	SET ENCRYPT PATH AS COPROCESSOR
Format:	5DH
Description:	This command will affect the configuration of the KGM in both encryption and decryption traffic modes. Once this command has been executed, the KGM will read data to be encrypted from the host port and will write the encrypted data back to the host port. During decryption, the KGM will read the data to be decrypted from the host port and will write the decrypted data back to the host port.
Prerequisites:	None

Status Code:

Condition	Status Message	Format
Successful operation	"Ready"	00

Command: SET ENCRYPT PATH TO OUTPUT

Format: DFH

Description: This command will affect only the encrypt configuration of the KGM. Once this command has been executed, the KGM will read data to be encrypted from the host port and will write the encyrpted data to the I/O port.

Prerequisites: None

Status Code:

Condition	Status Message	Format
Successful operation	"Ready"	00

Command: SET MODE (n)

Format: B0H, n (Where n is 0A, 0B, 0C, 0D, 0E, or 0FH)

Description: This command sets the particular cryptographic mode to be used for encryption or decryption. The modes are A through F. Once set, the mode will remain unchanged until the KGM receives another SET MODE (n) command, a RESTART command, a hardware reset or until power is interrupted.

Prerequisites: MODE* pin must be grounded if Mode F is to be selected.

Status Code:

Condition	Status Message	Format
n≠A,B,C,D,E or F or MODE* pin is not grounded and n=F	"Invalid Command"	03
Command received before original command complete	"Data Loss"	01
Successful operation	"Ready"	00

Command:	SET 8-BIT MODE		
Format:	57H		
Description:	This command sets the KGM configuration for 8-bit mode. In this mode the KGM transfers bytes utilizing bits 0 to 7 of the 16-bit internal interface to both the host and the I/O.		
Prerequisites:	None		
Status Code:	Condition	Status Message	Format
	Successful operation	"Ready"	00

Command:	SET 16-BIT MODE
Format:	58H
Description:	This command sets the KGM configuration for 16-bit mode. In this mode the KGM transfers full words utilizing a 16-bit interface to both the host and the I/O.
Prerequisites:	None

Status Code:

Condition	Status Message	Format
Successful operation	"Ready"	00

Command:	STOP
Format:	10H
Description:	If the STOP command is received when performing Encrypt or Decrypt, it will cause the KGM to process the remaining words that have already been read into the KGM, save the crypto states (MI's) and exit from Encrypt or Decrypt. If the STOP command is received while in the middle of a command process (other than Encrypt or Decrypt) when the KGM is expecting data, it will cause the process to stop. In both cases, the microcontroller will be zeroized and the KGM will be in the Executive.
Prerequisites:	None

Status Code:

Condition	Status Message	Format
STOP command received when expecting data (not during Encrypt or Decrypt)	"Data Loss"	01
Successful operation	"Ready"	00

Command: TRANSFER KEY (n)

Format: ABH, n

Description: Instructs KGM to internally transfer a RED Traffic key and associated MI's from RAM location (n) to the cryptoprocessor.
Note: This command may be used to prepare for an encrypt or decrypt.

Prerequisites:
1. KGM must have a valid key in location (n).
2. Key must be a Traffic key

Status Code:

Condition	Status Message	Format
No key in location (n) or not a Traffic key	"Wrong Key Type"	08
Command received before original command complete	"Data Loss"	01
3>n>255	"Invalid Command"	03
Key parity error	"Key Parity Error"	07
Invalid key	"Invalid Key"	0A
Random Seed key flag not set	"Random Seed Key Required"	59
Successful operation	"Ready" or	00
	"Ready, Encrypt Resync Required" or	5C
	"Ready, Decrypt Resync Required" or	5D
	"Ready, Encrypt and Decrypt Resync Required"	5F

Command: UNWRAP BLACK OTAR KEY (n)

Format: 25H, n, 320 Bits Remote Key

Description: Instructs KGM to decrypt a remote key and retain as key in location (n). The command contains location number (n), followed by 320 bits of remote key message (64-bit MI and a 256-bit encrypted key packet). The KGM decrypts this remote key message using the current Unique key (RAM location 1), stores the result in key storage location (n), and updates the Unique key.

Prerequisites: KGM must have a Unique key.

Status Code:

Condition	Status Message	Format
Command received before original command complete	"Data Loss"	01
3>n>255	"Invalid Command"	03
Unique key parity error	"Key Parity Error"	07
Decrypted key parity error	"Key Parity Error"	07
Update limit reached on Unique key	"Update Limit Reached"	11
Invalid key	"Invalid Key"	0A
Unique key not present	"Key Not Found"	09
Decrypted key wrong type for location (n) or key has wrong algorithm type	"Wrong Key Type"	08
Successful operation	"Ready"	00

Command: UNWRAP RANDOM SEED KEY

Format: F1H, (320 bits)

Description: Instructs the KGM to decrypt the encrypted key packet for the Random Seed key and store the result in the KGM for use in the random number generation process. The command is followed by 320 bits of MI and encrypted key packet. The KGM decrypts using the Storage key.

Prerequisites: KGM must have a Storage key.

Status Code:

Condition	Status Message	Format
Command received before original command completed	"Data Loss"	01
Storage key not found	"Storage Key Not Found"	0D
Storage key parity error	"Key Parity Error"	07
Decrypted key parity error	"Key Parity Error"	07
Key not a Random Seed key or key has wrong algorithm type	"Wrong Key Type"	08
Invalid key	"Invalid Key"	0A
Successful operation	"Ready"	00

Command: UNWRAP STORED BLACK KEY (n)

Format: 26H, n, (320 bits)

Description: Instructs the KGM to decrypt stored key packet and retain as key in location (n). The command contains the location number (n), followed by 320 bits of MI and stored key packet. The KGM decrypts using the Storage key and stores the resulting key packet in storage location (n).

Prerequisites: KGM must have Storage key.

Status Code:

Condition	Status Message	Format
Command received before original command completed	"Data Loss"	01
Storage key not found	"Storage Key Not Found"	0D
Storage key parity error	"Key Parity Error"	07
Decrypted key parity error	"Key Parity Error"	07
1>n>255 or n=2	"Invalid Command"	03
Designated key location (n) not allowed for key type or key has wrong algorithm type	"Wrong Key Type"	08
Invalid key	"Invalid Key"	0A
Successful operation	"Ready"	00

Command:	UPDATE RED KEY (n)
Format:	29H, n
Description:	Instructs KGM to perform a cryptographic update process on RED key at location (n).
Prerequisites:	1. KGM must have a valid key at location (n). 2. n≠2

Status Code:

Condition	Status Message	Format
1>n>255 or n=2	"Invalid Command"	03
No key in location (n)	"Key Not Found"	09
Key parity error	"Key Parity Error"	07
Update count reached maximum value	"Update Limit Reached"	11
Command received before original command complete	"Data Loss"	01
Invalid key	"Invalid Key"	0A
Successful operation	"Ready"	00

Command: VALIDATE KEY (n)

Format: 2AH, n

Description: Instructs KGM to perform a parity check on the RED key stored in RAM location (n).

Prerequisites: Key must be loaded into location (n).

Status Code:

Condition	Status Message	Format
1>n>255	"Invalid Command"	03
Parity error	"Key Parity Error"	07
Command received before original command complete	"Data Loss"	01
Invalid Key	"Invalid Key"	0A
Key has wrong algorithm type	"Wrong Key Type"	08
Successful operation	"Ready"	00

Command: WRAP RED KEY FOR STORAGE (n)

Format: A8H, n

Description: Instructs KGM to encrypt key packet stored at location (n) in the KGM and deliver the encrypted key packet, along with 64 bits of MI to the host. The Storage key (RAM location 2) is used for this encryption. When complete, the RED key remains in RAM at location (n) and a BLACK key record has been sent to the host for storage.

Prerequisites:
1. KGM must have Storage key.
2. KGM must have valid key at location (n).
3. $n \neq 2$
4. CIK device must be attached and CIK Option selected.

Status Code:

Condition	Status Message	Format
1>n>255 or n=2	"Invalid Command"	03
No Storage key in location (2) found	"Storage Key Not Found"	0D
Storage key parity error	"Key Parity Error"	07
Key not found or parity error in key location (n)	"Key Parity Error"	07
Invalid key	"Invalid Key"	0A
CIK option not selected	"Invalid Command"	03
Random Seed key flag not set	"Random Seed Key Required"	59
Command received before original command complete	"Data Loss"	01
Successful operation	"Ready"	00

Command: ZEROIZE ALL

Format: 2OH

Description: Zeroizes microcontroller, cryptoprocessor and all RED keys in storage RAM.

Prerequisites: None

Status Code:

Condition	Status Message	Format
Successful operation	"Ready"	00

Command: ZEROIZE (n)

Format: A1H, n

Description: This command zeroizes RAM location (n).

Prerequisites: None

Status Code:

Condition	Status Message	Format
0>n>255	"Invalid Command"	03
Command received before original command complete	"Data Loss"	01
Successful operation	"Ready"	00

6.5. KGM COMMAND EXECUTION TIMING

The Host/KGM interface is an asynchronous interface which buffers one Host to KGM transfer. The Host may write commands or operands to the KGM anytime that the BSYw control line is low. The Host is not limited to a maximum time between transfers (the KGM waits for any operands required by a command). See Section 7.3.a. for details on the Host Port interface timing.

For each command, Table 6-2 shows the times at which the KGM will require operands so that the KGM will not have to wait for that operand. An example explains this table. The format of the Decrypt Remote Key is the command followed by the first operand k, the second operand n1, and [224 + (k)160] data bits ([14 + (k)10] 16 bit data words).

From the first column of the table, 500 microprocessor instruction cycles after the Host issues the Decrypt Remote Key command the KGM will read the Host Port looking for the first operand, k. The Host may have written k immediatly after BSYw went low following the write of the command or it may delay the write until after the 500 instruction cycles. If the Host has not written by the end of the 500 cycles, the KGM will stop processing and wait for the Host write.

From the next column, 180 cycles after the host writes k the KGM will read the Host Port for the second operand, n1. Likewise, 4430 cycles later the KGM is looking for the first data word. Then, every 170 cycles the KGM will read another data word until 14 + k(10) data words have been read. The exception to this is after the 24th data word and after every 20th data word thereafter the delay will be 26,500 cycles instead of 170 cycles.

After the last data word is read there is a 30,600 cycle delay before the KGM writes the "ready" status message to the Host indicating that the command has been successfully completed and the KGM has returned to the Executive State. For commands that return data to the Host, the number in the "STATUS or data out" column represents the delay from the last Host to KGM write until the first write of data from the KGM to the Host.

The speed that the KGM will encrypt and decrypt data in its different cryptographic modes is shown in Table 6-3. Throughput is obtained by dividing the number of bits in a full block of data, by the time required to encrypt or decrypt that block. The result is given in megabits per second (Mbps). Table 6-3 is used in conjunction with Tables 6-4 and 6-5 to estimate the time required to execute an entire encyprt or decrypt process. Tables 6-4 and 6-5 show the number of instruction cycles required to enter and exit each of the cryptographic modes including the cycles required for synchronization overhead (reading or writing MI's and saving state). The numbers shown to exit a mode by means of a STOP command assume that the command was issued when the KGM is reading the last word before a full block boundry. Table 6-5 includes the time required to encrypt or decrypt this partial block, write out the resulting data, save crypto state, and return to the Executive State.

Table 6-2: Command Timing

COMMAND	OPERANDS 1st	2nd	3rd	4th	data	data	nth	STATUS or data out
	(number of microporcessor instruction cycles)							
Activate CIK								20,300,000
Combine Keys	510	11,000	12,800					16,400
Command Bypass	510				250	290		340
Copy Red Key	510	160						14,600
Data Bypass	510				250	290		340
Dec Bypass Audit Word	520				170	170		11,700
Decrypt Remote Key	500	180			4,430	170		30,600
data (20x) -)								
data (20x+1)						26,500		
Enc Bypass Control Word	520				170	170		17,200
Load Bypass Control Word					500	180		41,000
Load RSK					250	fill device		203,000
Load Red Key								
n not = 2	500				250	fill device		22,200
n = 2	500				250	fill device		20,200,000
MAC					700	80	(stop cmd)	870
data (4x) -) data (4x+1)						640		
Prepare Rekey (multiple)	510	7,160	5,710	3,880				200
Prepare Rekey (single)	490	7,150						22,500
Read Bypass Audit								15,900,000
Reset RSK								4,370
Restart								16,000,000
Self Test								20,000,000
Set Decrypt Direct								660
Set Decrypt Via Host								660
Set Enc - Coprocessor								660
Set Enc - to Output								660
Set Mode	500							240
Set 8-Bit Mode								660
Set 16-Bit Mode								660
Transfer Key								
con. state key	500							8,470
new key	500							19,100

Table 6-2: Command Timing (continued)

COMMAND	OPERANDS (number of microporcessor instruction cycles): 1st	2nd	3rd	4th	data........data	nth	or data out	STATUS
Unwrap Black OTAR Key	510				190	170		50,500
Unwrap RSK					480	170	170	350,000
Unwrap Stored Black Key	5,950				210	170		18,200
Update Red Key	500							24,900
Validate Key	500							18,100
Wrap Red Key For Storage	530							30,900
Zeroize All								751,000
Zeroize	500							3,070

ASSUMPTIONS:

- All commands issued from the Executive state.
- Operating on a 16-bit bus.
- Host provides operands and data as fast as TEPACHE can handle them.

Table 6-3: Cryptographic Mode Throughput Rates

MODE	DATA BUS WIDTH (bits)	BLOCK SIZE (bits)	THROUGHPUT (Mbps)
A (normal)	16	64	7.60
A	16	64	1.05
A	8	64	0.69
B	16	8	0.16
B	8	8	0.16
C	16	64	0.94
C	8	64	0.63
D (normal)	16	128	2.87
D	16	128	1.38
D	8	128	0.82
E	16	8	0.071
E	8	8	0.071
F	16	128	1.18
F	8	128	0.75

Tabel 6-4: Encrypt/Decrypt Timing (Enter)

COMMAND	CYCLES TO ENTER MODES (number of microprocessor instruction cycles)
Encrypt With Resync	
Mode A (Normal)	500
Mode A	1430
Mode B	1210
Mode C	1410
Mode D (Normal)	740
Mode D	1300
Mode E	1100
Mode F	780
Encrypt Without Resync	
Mode A	850
Mode B	850
Mode C	850
Mode D (Normal)	430
Mode D	850
Mode E	670
Decrypt With Resync	
Mode A (Normal)	970
Mode A	1520
Mode B	1510
Mode C	1570
Mode D (Normal)	990
Mode D	1600
Mode E	1390
Mode F	800
Decrypt Without Resync	
Mode A	580
Mode B	870
Mode C	860
Mode D (Normal)	440
Mode D	860
Mode E	690

ASSUMPTION:

- All commands issued from the Executive state.
- Operating on a 16-bit bus.
- Host and I/O read and write data as fast as TEPACHE can handle them.
- For Encrypt Without Resync and Decrypt Without Resync the key in use was saved on a block boundry.

Tabel 6-5: Encrypt/Decrypt Timing (Exit)

COMMAND	CYCLES TO EXIT MODES (number of microprocessor instruction cycles)
Abort	330
Stop	
from Executive	460
from Encryption	
Mode A (Normal)	4440
Mode A	6430
Mode B	6000
Mode C	6430
Mode D (Normal)	4770
Mode D	6900
Mode E	5880
Mode F	6980
from Decryption	
Mode A (Normal)	4460
Mode A	6460
Mode B	5890
Mode C	6300
Mode D (Normal)	4670
Mode D	6900
Mode E	5780
Mode F	5850

ASSUMPTIONS:

- Operating on a 16-bit bus.
- Host and I/O read and write data as fast as TEPACHE can handle them.
- For the Stop command, the command interrupt is issued during the read of the last word before a block boundry. (Except Mode F, interrupt is on the block boundry.)

7. INTERFACE SPECIFICATIONS

7.1. DETAILED INTERFACE DESCRIPTION

As briefly described in Section 4.3, the interface is composed of 8 major signal groupings - host port, host port control, I/O port, I/O port control, CIK port, fill port, housekeeping port and power port - utilizing 96 pins (including spares). The KGM pinout is shown in Figure 7-1, and Table 7-1 defines the function of each pin. The function of the port controls is illustrated in Truth Table 7-2 and 7-3 for the host and I/O ports.

7.2. ELECTRICAL SIGNAL CHARACTERISTICS

a. Electrical Power Requirements

1. Prime operation input voltage will have a value of 5.0 ±5% VDC supplied by the host.

2. Input current = 1 Ampere max.

3. Power Dissipation = 5 Watts max.

4. Transients

The host is responsible for providing power to the KGM. Positive transients must be limited to +7V and negative transients to -0.3V DC. If the regulated power within the KGM drops below 4.55 ± .05V during an input power transient, the KGM will perform a hardware reset. (Note that a hardware reset zeroizes all RED keys within the KGM).

b. Electrical Characteristics of Input/Output

1. Signal interfaces for all inputs and outputs are LSTTL compatible except as described in Figure 7-2.

2. Absolute maximum ratings beyond which the KGM may be damaged:

Supply Voltage, Vcc -0.3 to 7.0V DC

Figure 7-1
TEPACHE Pinout

Port	Pin	A	B	C	Port
FILL PORT	1	DATA	Vdd	DATA OUT	CIK PORT
	2	CLOCK	GND	CIK Vcc	
	3	REF	SPARE	CIK SENSE*	
	4	SENSE/MUX	TAMPI	SPARE	
	5	REQ*	ALARM*	CIK GND	
	6	OVERRIDE	CIK OPT	DATA IN	
	7	SPARE	TAMPO	SK	
HOST PORT	8	BSYw	ZEROIZE*	CSC	
	9	BSYr	RESET*	BSYo	I/O PORT
	10	SRQi/o*	SYST. MGR.*	SRQi*	
	11	BSYi/o	MODE*	BSYi	
	12	CMD*		CMDo*	
	13	CS*		CSo*	
	14	SRQ*		SPARE	
	15	WR*		WRo*	
	16	RD*		RDo*	
	17	ho		io	
	18	h1		i1	
	19	h2		i2	
	20	h3		i3	
	21	h4		i4	
	22	h5		i5	
	23	h6		i6	
	24	h7		i7	
	25	h8		i8	
	26	h9		i9	
	27	h10		i10	
	28	h11		i11	
	29	h12		i12	
	30	h13	SPARE	i13	
	31	h14	Vdd	i14	
	32	h15	GND	i15	

POWER PORT
&
HOUSEKEEPING PORT

Table 7-1: Pin Description

Pin	Name	Type	Description
A1	DATA	I	FILL DATA This pin carries data synchronized by the FILL CLOCK.
A2	CLOCK	I	FILL CLOCK This pin carries the clock signal for clocking data into the KGM over the fill port.
A3	REF	O	FILL LOGIC LEVEL REFERENCE This pin is used as a logic reference for the fill device. It is not intended to be used as a power supply to power circuits. It is nominally +5V DC.
A4	SENSE/MUX	I	FILL SENSE/MUX When SENSE/MUX = +5V, it signifies to the KGM that a fill device is attached.
A5	REQ*	O	FILL REQUEST The KGM pulls this signal low to initiate a fill process if required by the fill device.
A6	OVERRIDE	I	OVERRIDE FILL (Not used)
A7,B3	SPARE		Spare pins
A8	BSYw	O	BUSY WRITE When BSYw=1 the KGM is notifying the host that the KGM input buffer is full or that the KGM is performing a task and not ready for new data or a command. See Table 7-2. The KGM will accept writes when BSYw=1, but the previous unread contents of the input buffer may be lost.
A9	BSYr	O	BUSY READ When BSYr=0, the KGM is notifying the host that new data is in the host port output buffer and that the host can do a read of the KGM. See Table 7-2.
A10	SRQi/o*	O	I/O SERVICE REQUEST SRQi/o* is a direct feed-through from the SRQi* signal from the I/O port.

Pin	Name	Type	Description
A11	BSYi/o	O	I/O BUSY When BSYi/o=1, the KGM is notifying the host that the I/O is not ready to be read by the host or that the KGM is addressing the I/O. The KGM sets BSYi/o=1 when encrypting data or decrypting data received directly from the I/O port. See Table 7-2.
A12	CMD*	I	This signal is used by the host to specify whether command or data is being written to the KGM or read from the KGM or from the I/O. See Table 7-2.
A13	CS*	I	CHIP SELECT Signal from host. Alerts KGM that the host will perform a Read or Write. See Table 7-2.
A14	SRQ*	O	SERVICE REQUEST The KGM will bring SRQ*=0 (latched output) when a status word is available in the host port output buffer. When the host performs a read of the KGM, the SRQ* latch is released (SRQ*=1)
A15	WR*	I	WRITE STROBE When WR* is strobed low, the host writes a word into the KGM. See Table 7-2.
A16	RD*	I	READ STROBE When RD* is strobed low, the host reads a word from the KGM or the I/O. See Table 7-2.
A17-A32	h0-h15	I/O	HOST BUS h0-h15 are bidirectional data lines used to communicate between the host and KGM.
B1,B31	Vdd	I	POWER SUPPLY + These pins contains the input power for the KGM, regulated 5V DC.
B2, B32	GND	O	POWER SUPPLY GROUND These pins are also the reference for all signals.
B4	TAMPI	I	TAMPER IN This line senses current flow in a tamper loop using TAMPER OUT as a source. If TAMPER IN voltage is low (<.7V), then tamper will be detected. Tamper detection will cause an automatic zeroize and subsequent KGM reset.

Pin	Name	Type	Description
B5	ALARM*	O	ALARM Under normal conditions, ALARM*=1. When an alarm is present in the KGM, ALARM*=0.
B6	CIK OPT	I	CIK OPTION When this pin is high, the KGM is configured to utilize a CIK. This pin is sampled during KGM initialization and should not be changed while the KGM is in operation.
B7	TAMPO	O	TAMPER OUT The tamper out line is +5V, current limited to 10mA. This line is used as a source for TAMPER IN.
B8	ZEROIZE*	I	ZEROIZE When this pin is grounded, all RED keys within the KGM are erased. It will also result in a KGM reset.
B9	RESET*	I	RESET When RESET*=0 it causes the KGM to zeroize; and when RESET*=1, to begin its initialization.
B10	SYSTEM MGR*	I	SYSTEM MANAGER When this pin is connected to ground, the KGM is configured as a System Manager. When it is connected to 5V, the KGM is configured as a Node.
B11	MODE*	I	MODE When this pin is connected to 5V, continued state operation of Modes A, B, and C is not allowed and Mode F cannot be used.
B12-B30	SPARE		Spare pins
C1	DATA OUT	O	Data output to CIK.
C2	CIK VCC	O	CIK Power +5V power to CIK.
C3	CIK SENSE*	I	CIK Sense Sense line for CIK, 0V when CIK attached.
C4, C14	SPARE		Spare pins
C5	CIK GND	I	CIK Ground Ground for CIK.
C6	DATA IN	I	Data input from CIK.
C7	SK	O	CIK CLOCK Clock for CIK.

Pin	Name	Type	Description
C8	CSc	O	CIK CHIP SELECT Chip Select for CIK. The KGM uses this signal CSc=1 to notify the CIK that it is being addressed.
C9	BSYo	I	I/O BUSY When BSYo=1, the I/O is notifying the KGM that it is not ready to be read. See Table 7-3.
C10	SRQi*	I	I/O INTERRUPT This signal is fed through to the SRQi/o* pin on the host port and also used by the KGM for control.
C11	BSYi	I	INPUT BUSY When BSYi=1, the I/O is notifying the KGM that it is not ready to be written to. See Table 7-3.
C12	CMDo*	O	This signal is used by the KGM to specify whether command or data is being written to the I/O or a read is being performed by the KGM or host. See Table 7-3.
C13	CSo*	O	I/O CHIP SELECT The KGM uses this signal CSo*=0 to notify the I/O that it is being addressed. See Table 7-3.
C15	WRo*	O	I/O WRITE STROBE When WRo* is strobed low by the KGM, it writes a word into the I/O. See Table 7-3.
C16	RDo*	O	I/O READ STROBE When RDo* is strobed low by the KGM, it reads a word from the I/O or sends it to the host. See Table 7-3.
C17-C32	i15-i0	I/O	I/O LINES i0-i15 are bidirectional data lines used to communicate between the KGM and I/O.

Table 7-2: Host Port Truth Table

CS*	CMD*	RD*	WR*	BSYr	BSYw	BSYi/o	ACTION
1	X	X	X	X	X	X	No action
0	0	U	1	X	X	0	Host reads word from I/O
0	1	U	1	0	X	X	Host reads word from KGM
0	0	1	U	X	0	X	Host writes command word to KGM
0	1	1	U	X	0	X	Host writes data word to KGM
0	X	1	U	X	1	X	Host overwrites contents of KGM input buffer

U = Low Going Strobe

X = Don't Care

Table 7-3: I/O Port Truth Table

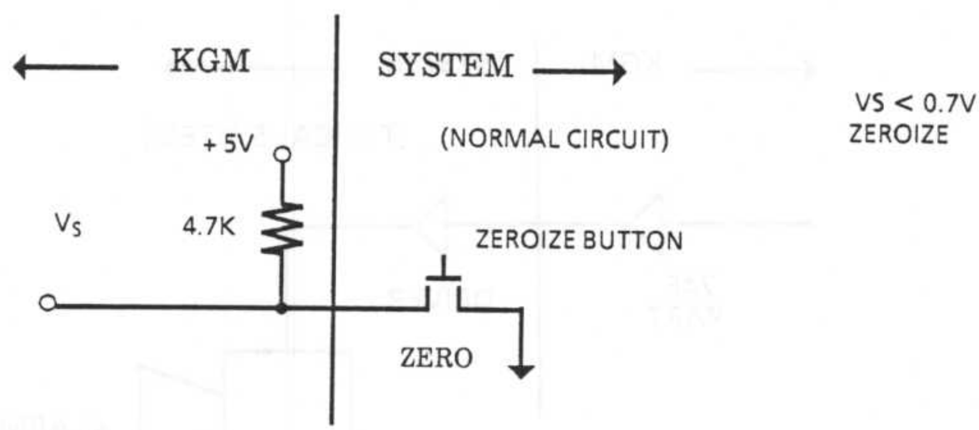
CSo*	CMDo*	RDo*	WRo*	BSYo	BSYi	ACTION
1	X	X	X	X	X	No action
0	1	U	1	0	X	Data is read from I/O by KGM
0	0	U	1	0	X	Data is read from I/O by host
0	1	1	U	X	0	KGM writes data to I/O
0	0	1	U	X	0	KGM writes command to I/O

U = Low Going Strobe

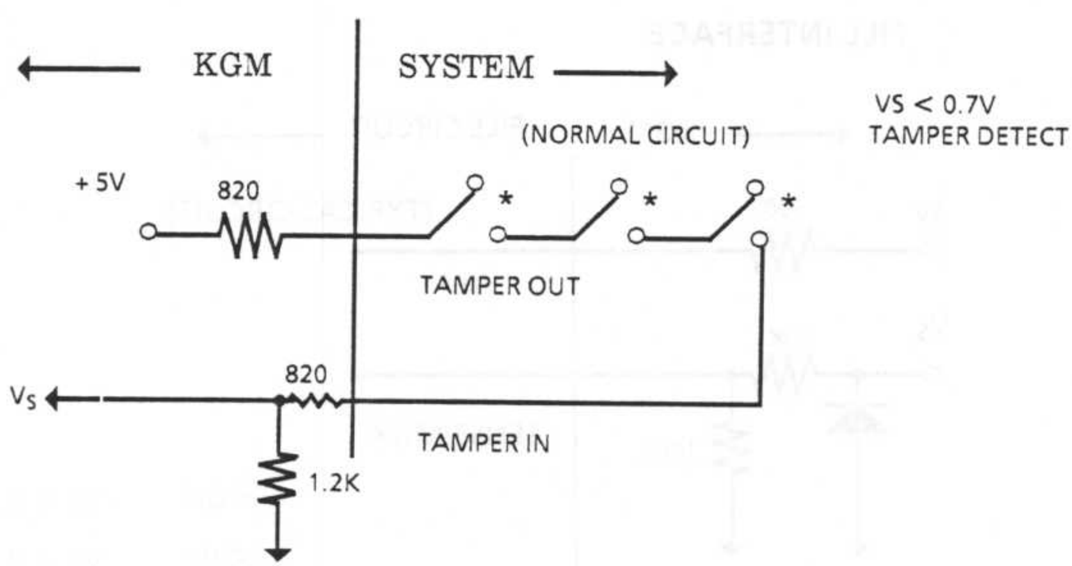
X = Don't Care

Fig. 7-2: Interface Circuits

ZEROIZE



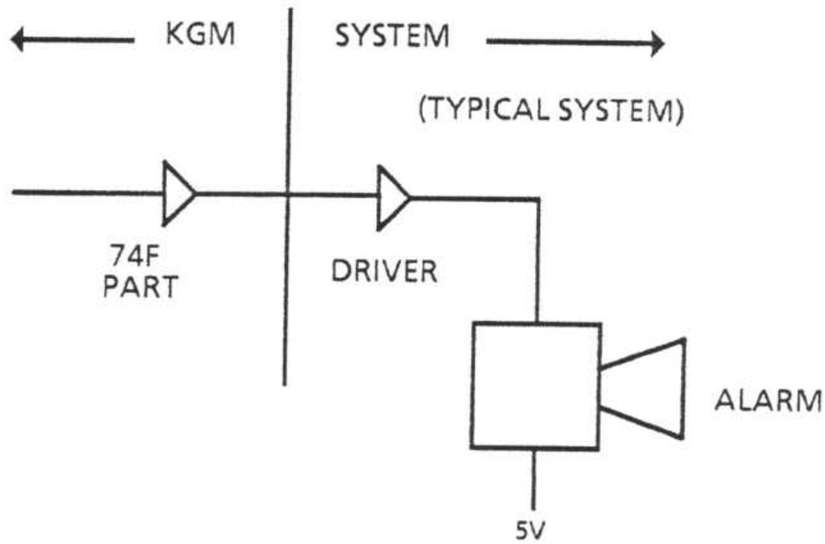
TAMPER OUT/TAMPER IN



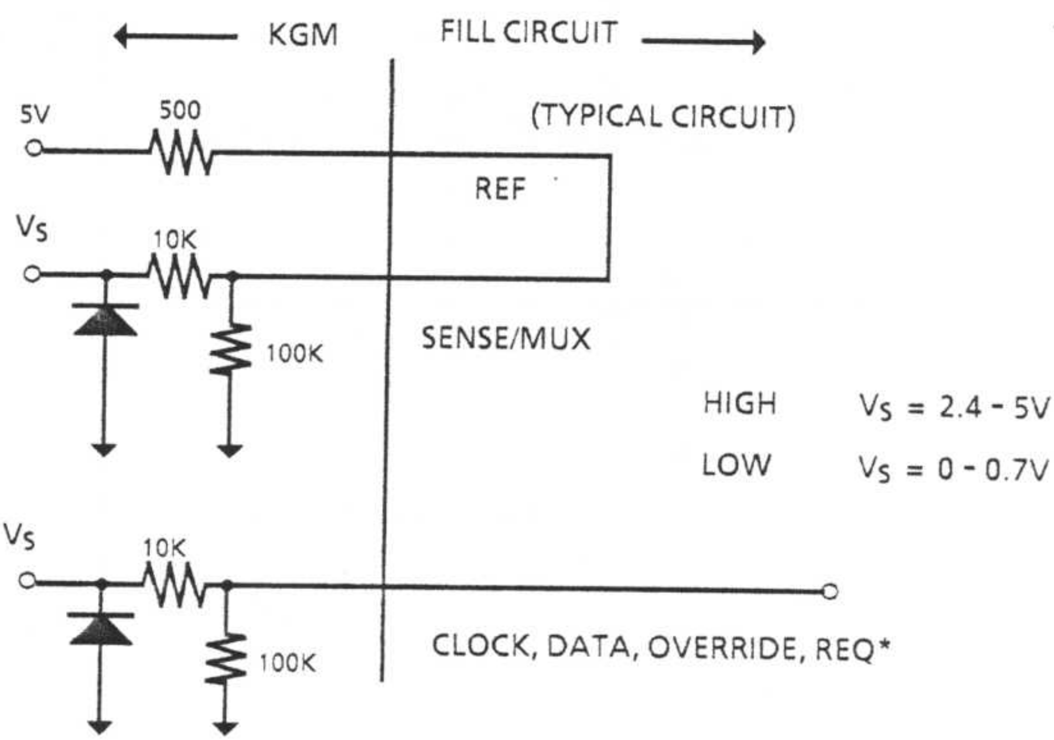
* SERIES N.C. SWITCHES

Figure 7-2

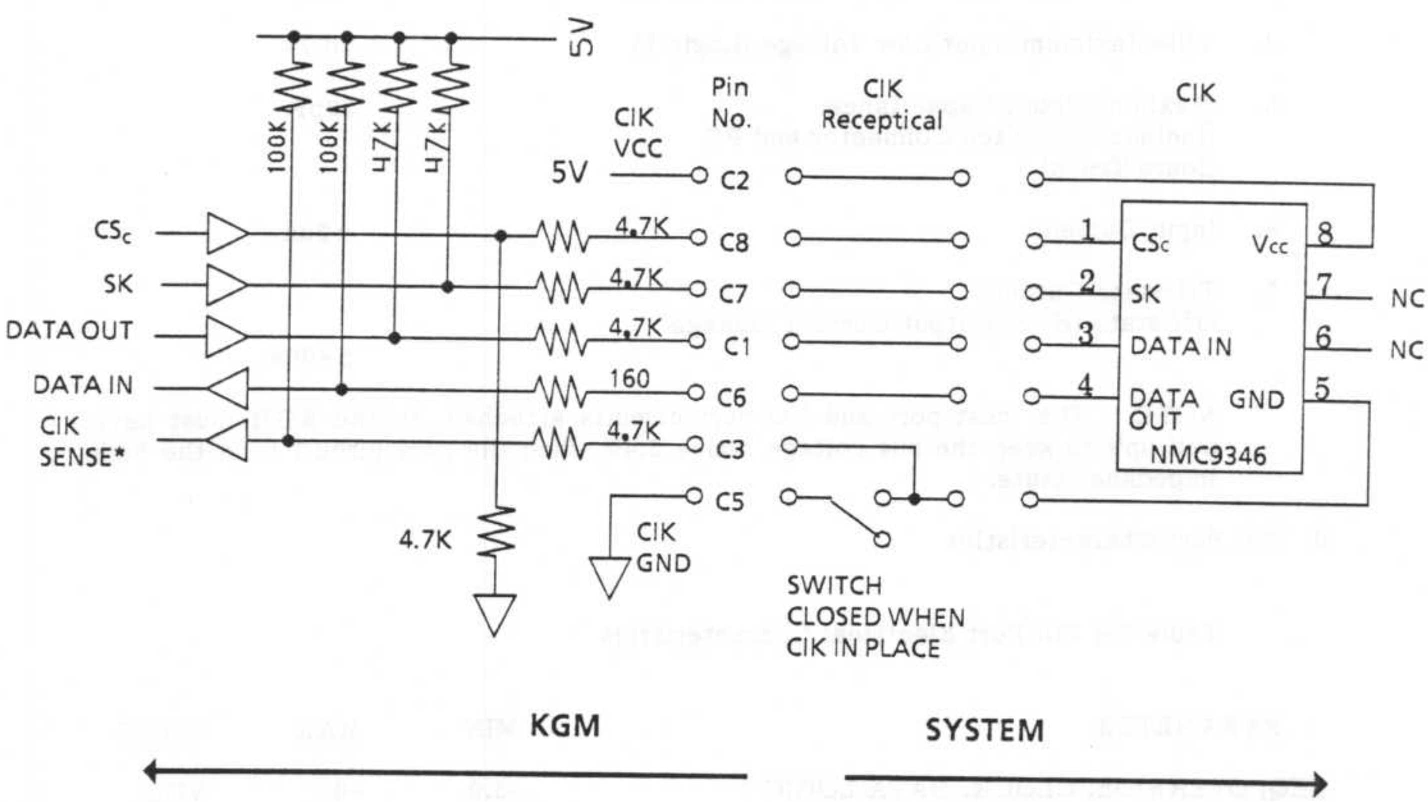
ALARM



FILL INTERFACE



Recommended Crypto Ignition Key (CIK) Circuit



c. Host & I/O Port D.C. Characteristics

1.	Voh--Minimum High Output Voltage (Logic 1) Iout=6ma; Vcc=+4.75V	+2.4V
2.	Vol--Maximum Low Output Voltage (Logic 0) Iout=6ma Vcc=+4.75V	+0.5V
3.	Vih--Minimum Input High Voltage (logic 1)	+2.0V
4.	Vil--Maximum Input Low Voltage (Logic 0)	+0.7V
5.	Maximum Input Capacitance [Includes Interface Connector and PC Board Track]	30pF
6.	Input Current	± 2ua
7.	Tri-state Outputs Off state (Hi Z) Output Current leakage	± 40ua

NOTE: The host port and I/O port circuits attached to the KGM must have pull-up's to keep the bus voltage above 2.4V when the port circuit is in the high impedance state.

d. Fill Port Characteristics

Table 7-4 Fill Port Electrical Characteristics

PARAMETER	MIN	MAX	UNITS
REQ, OVERRIDE, CLOCK, DATA LOGIC 1	-3.0	+0.7	VDC
REQ, OVERRIDE, CLOCK, DATA LOGIC 0	2.4	5.0	VDC

Reference CSESD-11

7.3. KGM TIMING RELATIONSHIPS

All voltage levels and loads will be as specified in Section 7.2. A temperature range of 0-70° C is assumed.

a. Host Port

The host may read data from the KGM host port output buffer or from the I/O circuit via the KGM. To perform a read of the KGM host port output buffer, the host must first check the BSYr line; if low, a valid KGM host port output buffer read can be performed. This read is performed by setting CMD*=1, and bringing CS* low which tells the KGM it is being addressed. The host then asserts the Read Strobe, RD*. KGM output will then appear on the KGM data lines ho-h15. The host should sample the data with the rising edge of the Read Strobe RD*. The output data will remain on h0-h15 for a short time after RD* go high. (See Figure 7-3). When the host reads from the I/O circuit via the KGM, the process proceeds as shown in Figure 7-7. To perform a read of the I/O circuit, the host must first check the BSYi/o line; if low, a valid read can be performed. This read is performed by setting CMD*=0 and bringing CS* low which tells the KGM it is being addressed. The host then asserts the Read Strobe, RD*. The KGM will then place data from the I/O circuit on the host data bus. The host should sample the data with the rising edge of the Read Strobe, RD*. The KGM will set BSYi/o=1 while encrypting data which is sent to the I/O port or decrypting data received from the I/O port. This is done to avoid contention between the KGM and host for the I/O circuit. (See Section 7.3.b)

Data may also be written by the host into the KGM host port. The host must first check the BSYw line; if low, a valid write can be performed (see Figure 7-4). Command or Data is selected by the host using CMD* select line. The host then brings Chip Select CS* low which tells the KGM it is being addressed. The host then places data on the KGM data lines h0-h15. The host asserts the Write Strobe and the KGM accepts the data from h0-h15 on the rising edge of the Write Strobe. The write cycle will be terminated by WR* going high. The KGM BSYw line will go high and stay high until the KGM reads its host port input buffer, and completes any operation initiated by the write command. The host can write to the KGM even if BSYw=1. In this case the host may overwrite the contents of the KGM host port input buffer.

In the encrypt and decrypt modes (described in Section 5.1) large amounts of data are being transferred. Data transfer during encryption and decryption is accomplished as follows:

Fig. 7-3: Host Reads Data from KGM Host Port Latch

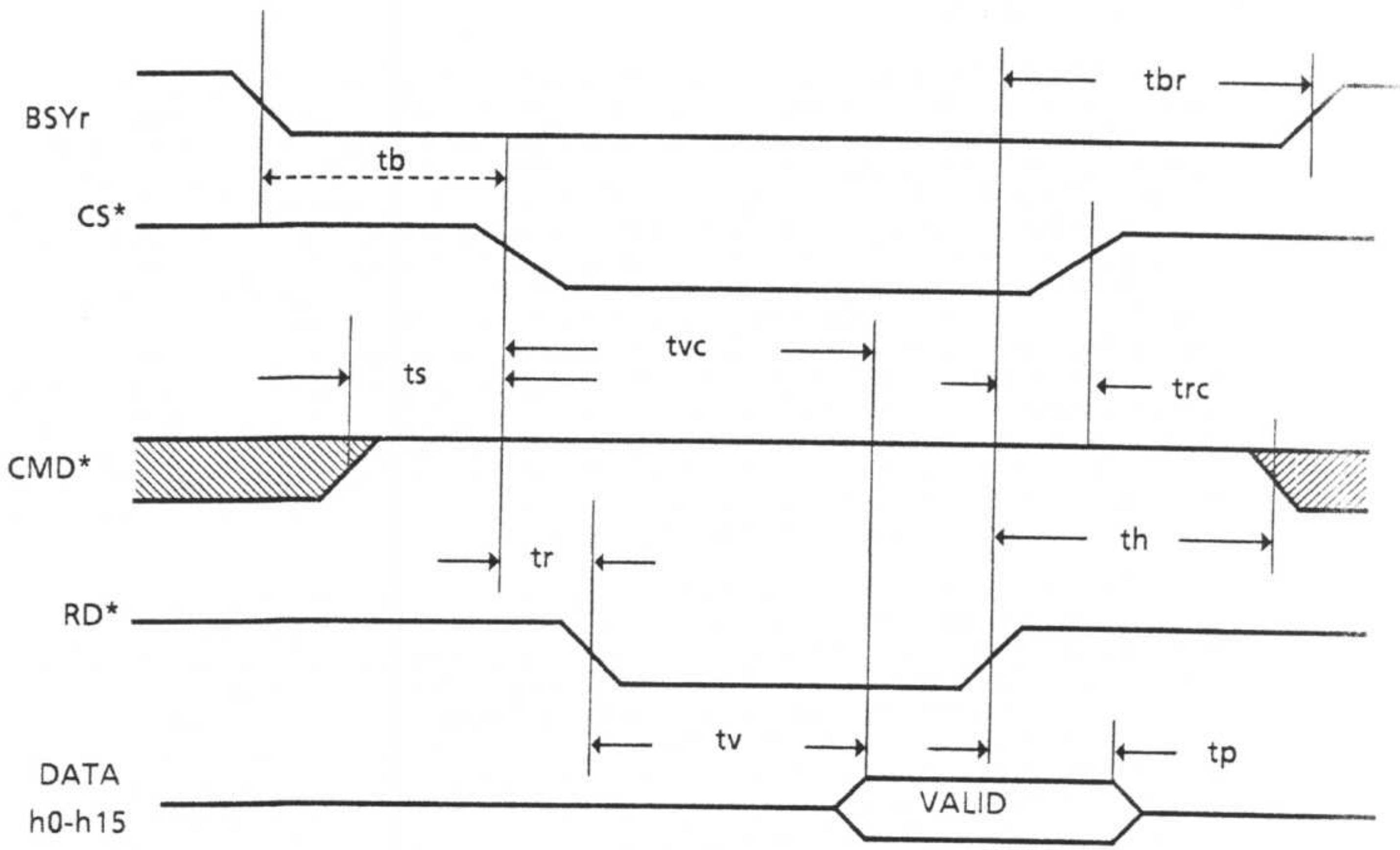


Fig. 7-4: Host Writes Data or Command to KGM

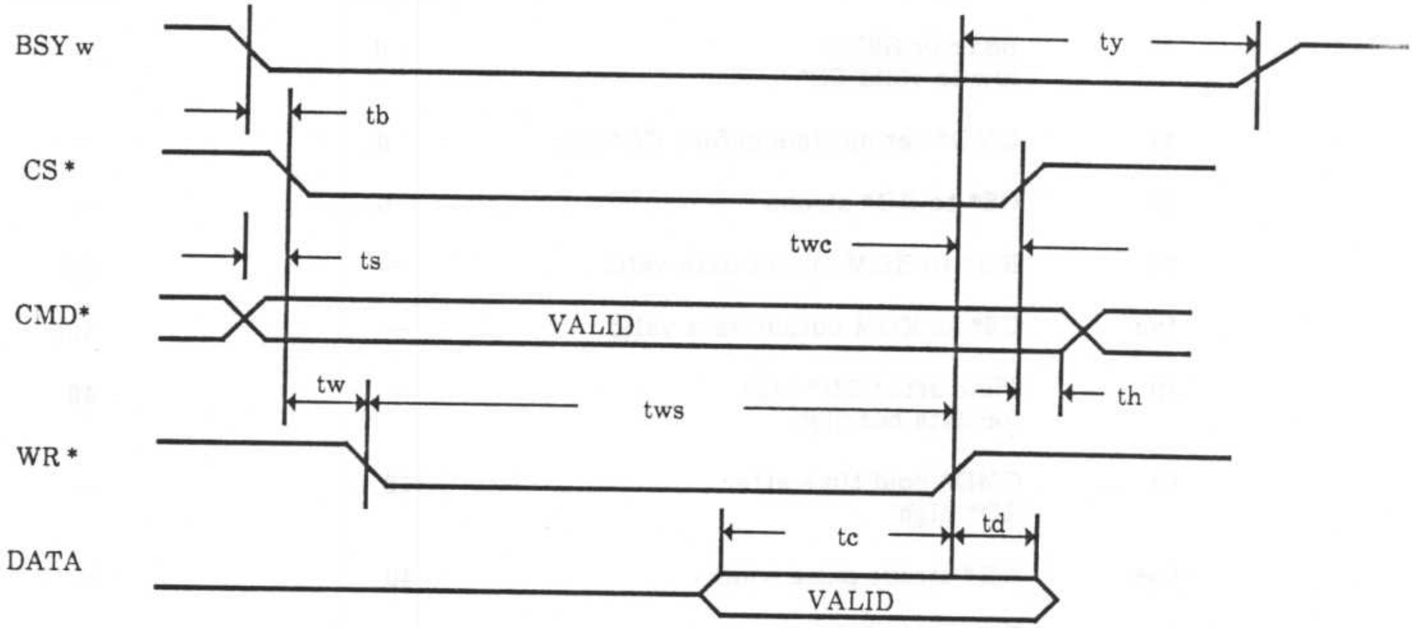
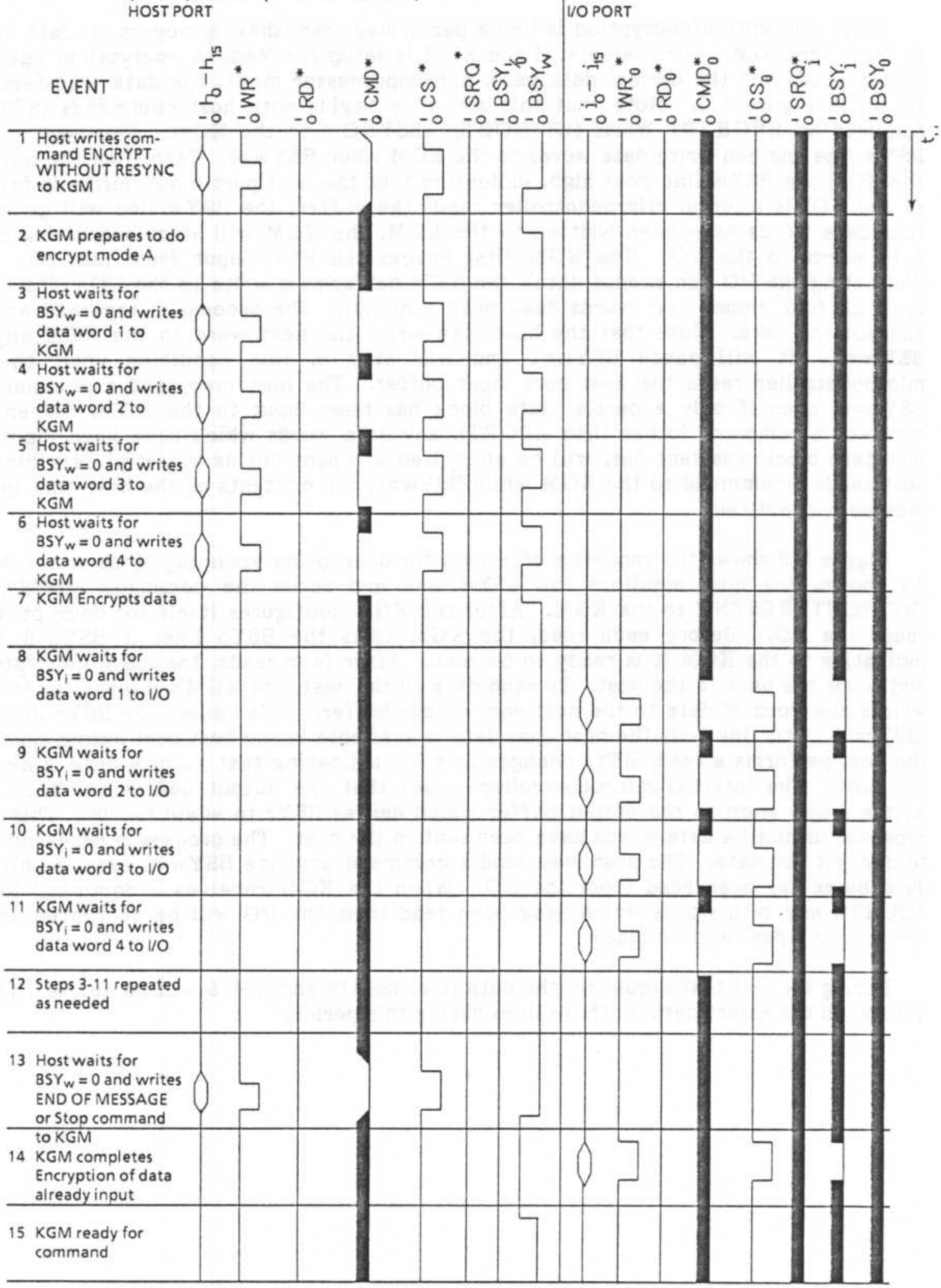


Table 7-5: Host Port Timing Characteristics

Symbol	Parameter	Min(nS)	Max(nS)
tb	BSYr or BSYw low to valid CS*	0	--
ts	CMD* set up time before CS* low	0	--
tr	CS* to RD* strobe	0	--
tv	RD* to KGM output data valid	--	62
tvc	CS* to KGM output data valid	--	106
tp	Time after RD* high for data bus high	--	40
th	CMD* hold time after RD* high	10	--
tws	WR* strobe pulse width	40	--
tw	CS* to WR* strobe	0	--
tc	Input data setup time before WR* high	15	--
td	Input data hold time after WR* high	8	--
ty	WR* high to BSYw high	--	127
trc	RD* high strobe to CS* high	10	--
twc	WR* high strobe to CS* high	10	--
tbr	RD* high to BSYr high	--	127

Fig. 7-5: Encrypt Mode A Events
(16 Bit, Non-Coprocessor Modes)



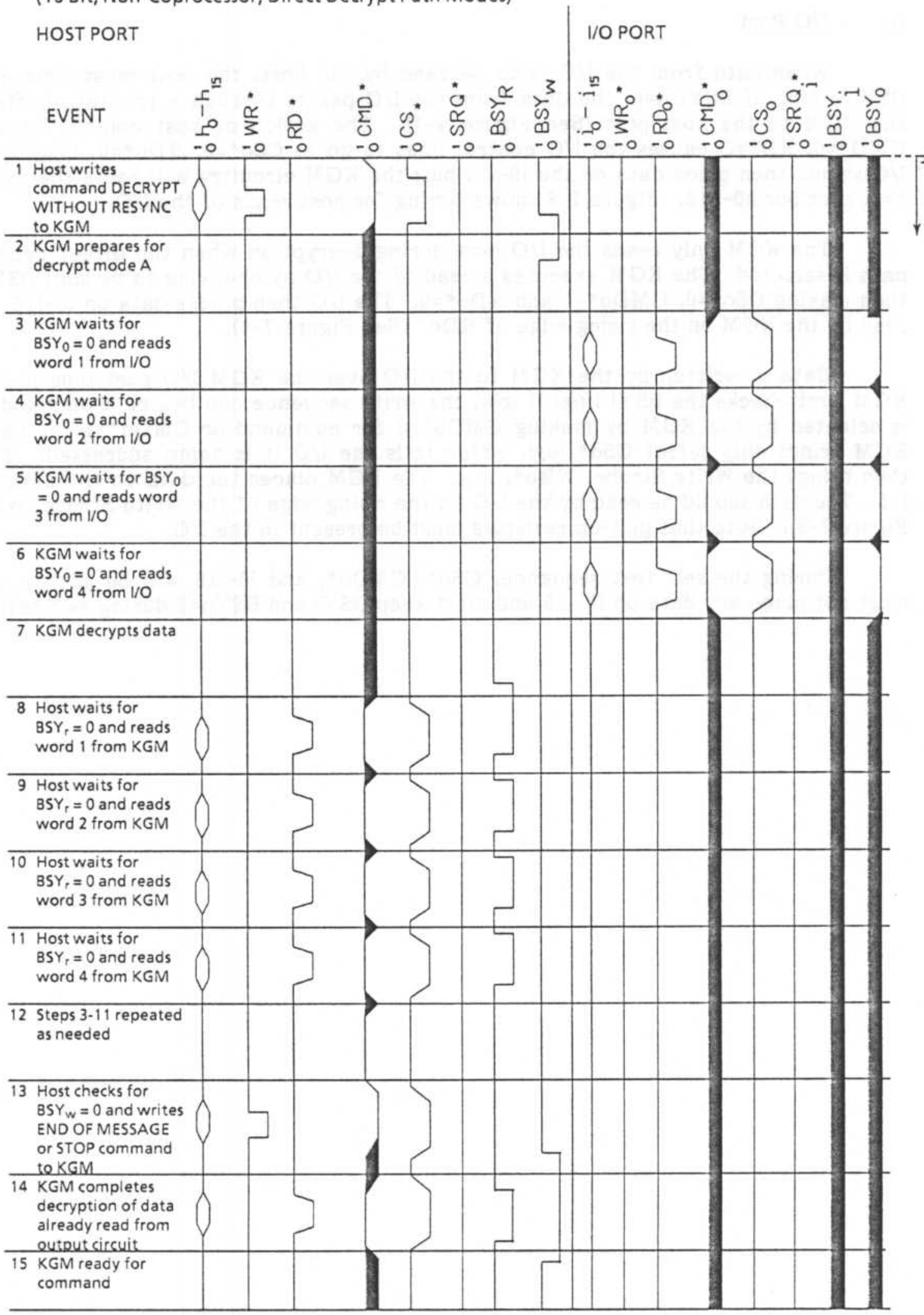
* For ENCRYPT WITH RESYNC, the host writes 4 words of MI to the KGM before beginning step 3.

When encryption/decryption is being performed, handshaking occurs on data transfers to or from the KGM. For example, if the KGM is setup for Mode A encryption, operating with 16-bit words and the normal data path, non-coprocessor mode, the data transfers will be as shown in Figure 7-5. Note that this sequence begins with host commands (SET MODE A followed by ENCRYPT WITH (WITHOUT) RESYNC) to the KGM. The host monitors the BSYw line and can write data words to the KGM when BSYw=0. Each time the host writes to the KGM the BSYw line goes high, indicating that the host port input buffer is full. As soon as the KGM's internal microcontroller reads the buffer, the BSYw line will go low. After four data words have been written to the KGM, the KGM will start outputting cipher text data words to the I/O. The KGM first checks the BSYi input from the I/O. If BSYi=0 (indicating the I/O can accept data), the KGM performs a write to the I/O. This is repeated until all four cipher text words have been sent out. The process is repeated as needed to encrypt the data. Note that the host may write the next word to the KGM anytime that BSYw=0. It will cause BSYw=1 and will stay in this condition until the internal microcontroller reads the host port input buffer. The host may send a command anytime BSYw=0, even if only a partial data block has been input to the KGM. When the KGM receives a command (other than ABORT), any data words which have been input since the last data block was sent out, will be encrypted and sent out as a partial data block. If the host sends a command to the KGM while BSYw=1, the contents of the host port input buffer may be overwritten.

Figure 7-6 shows the sequence of events for decryption assuming Mode A has been setup. As shown, the host monitors the BSYw line and sends the command DECRYPT WITH (WITHOUT) RESYNC to the KGM. After the KGM configures itself for decrypt Mode A, it reads the I/O. Before each read, the KGM tests the BSYo line; if BSYo=0 the I/O is indicating to the KGM it is ready to be read. After four reads, the KGM decrypts the data and sends the data to the host. To send data to the host, the KGM's internal microcontroller writes one word of data to the host port output buffer. This causes the BSYr line to go low (BSYr=0). This line tells the host that data is available in the host port output buffer. After the host performs a read, BSYr changes to a "1" indicating that no new data is available for the host. The internal microcontroller senses that the output buffer has been read, and writes a new word to the output buffer which causes BSYr to again be "0". This process is repeated until all 4 data words have been sent to the host. The process is repeated as needed to decrypt the data. The host may send a command anytime BSYw=0 even if only a partial data block has been read from the I/O. When the KGM receives a command (other than ABORT), any data words which have been read from the I/O will be decrypted and sent to the host as a partial data block.

During the self test sequence, the data lines ho-h15 and io-i15 will be active. The host or I/O should not assert data on these lines during this period.

Fig. 7-6: Decrypt Mode A Events
(16 Bit, Non-Coprocessor, Direct Decrypt Path Modes)



* For DECRYPT WITH RESYNC, the KGM reads 4 words of MI from I/O before beginning step 3.

b. I/O Port

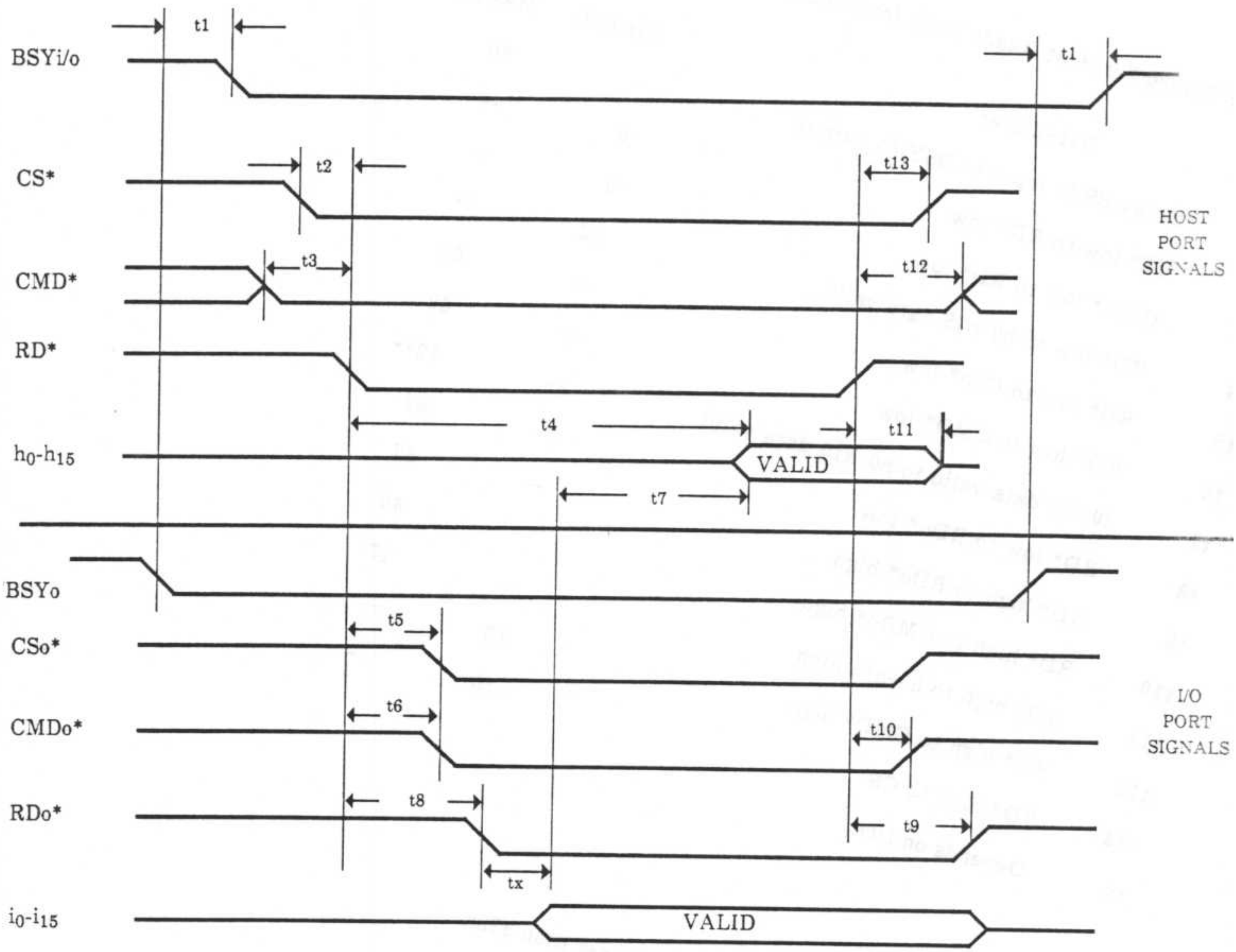
When data from the I/O is to be read by the host, the host must first check the BSYi/o pin. If BSYi/o=0, the data from the I/O passes through a tri-state buffer within the KGM to the host port (See Figure 4-1). The action of host control lines CS*=O, CMD*=0, RD*=O causes the I/O control lines to go to CSo*=0, RDo*=0, CMDo*=0. The I/O should then place data on the i0-i15 bus; the KGM circuitry will route the data to the host port bus h0-h15. Figure 7-8 shows timing for host reads of the I/O.

The KGM only reads the I/O port during decryption when the direct decrypt data path is selected. The KGM executes a read of the I/O by checking to be sure BSYo=0 and then making CSo*=0, CMDo*=1 and RDo*=0. The I/O then places data on io-i15, which is read by the KGM on the rising edge of RDo* (See Figure 7-8).

Data is written by the KGM to the I/O over the KGM I/O port pins i0-i15. The KGM first checks the BSYi line; if low, the write sequence can begin. Command or Data is selected by the KGM by making CMDo*=0 for command or CMDo*=1 for data. The KGM brings chip select CSo* low, which tells the I/O it is being addressed. The KGM then brings the Write Strobe, WRo*, low. The KGM places the data on the data lines i0-i15. The data should be read by the I/O on the rising edge of the Write Strobe, WRo* (See Figure 7-9). Note that pull-up resistors must be present in the I/O.

During the self test sequence, CSo*, CMDo*, and i0-i15 will be active. The I/O must not place any data on i0-i15 and must keep BSYi and BSYo=0 during self test.

Fig. 7-7: Host Reads Data from I/O



6: Timing for Host Reads Data from I/O Port

ı	Parameter	Min (nS)	Max (nS)
	Delay BSYo input to BSYi/o output	-	60
	CS* low to RD* low	0	--
3	CMD* low to RD* low	0	--
t4	RD* low to h0-h15 data valid	82	1*
t5	RD* low to CSo* low	--	60
t6	RD* low to CMD* low	--	60
t7	i0-i15 data valid to h0-h15 data valid	--	12**
t8	RD* low to RDo* low	--	67
t9	RD* high to RDo* high	--	67
t10	RD* high to CMDo* high	--	60
t11	RD* high to h0-h15 high	--	17
t12	RD* high to CMD* high	10	--
t13	RD* high to CS	10	--
tx	Depends on I/O		

NOTES

* Depends on tx (See Figure 7-7) but not less than 82ns

** 12ns max or tx conditions ,which ever is later

Fig. 7-8: KGM Reads Data from I/O

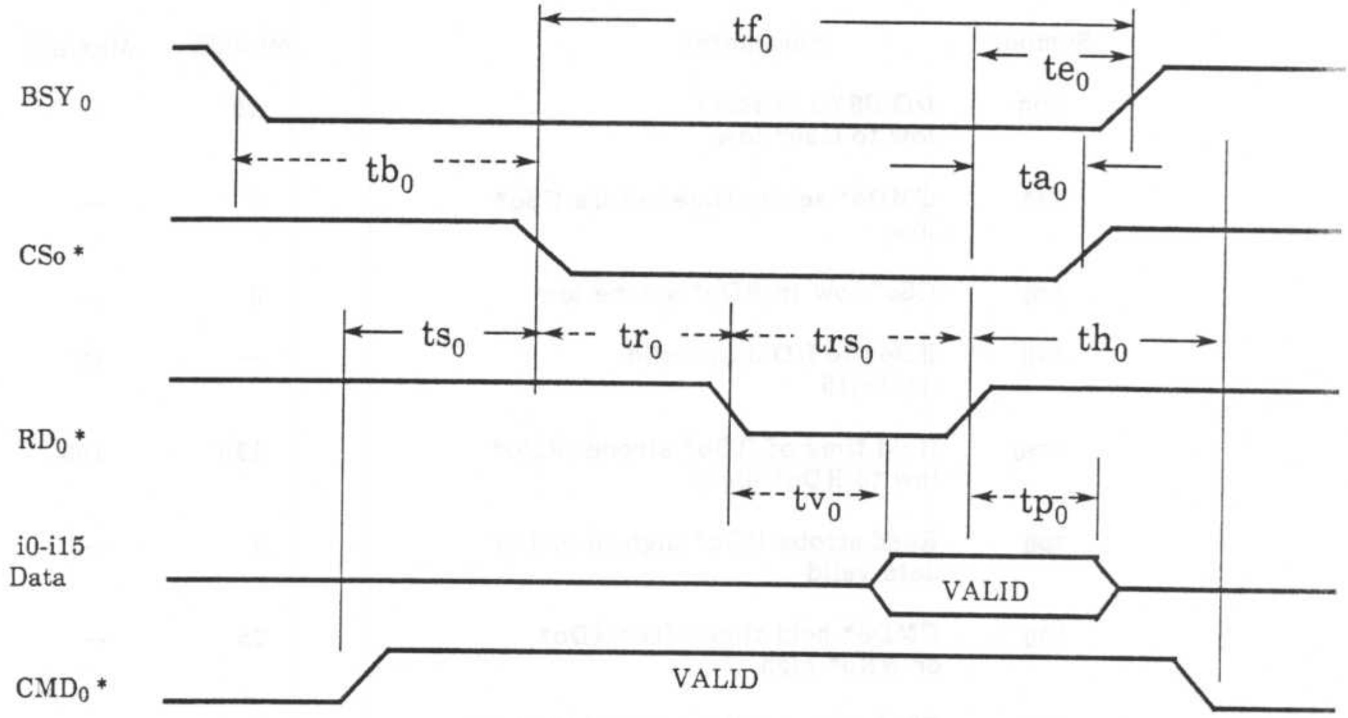


Fig. 7-9: KGM Writes Command or Data to I/O

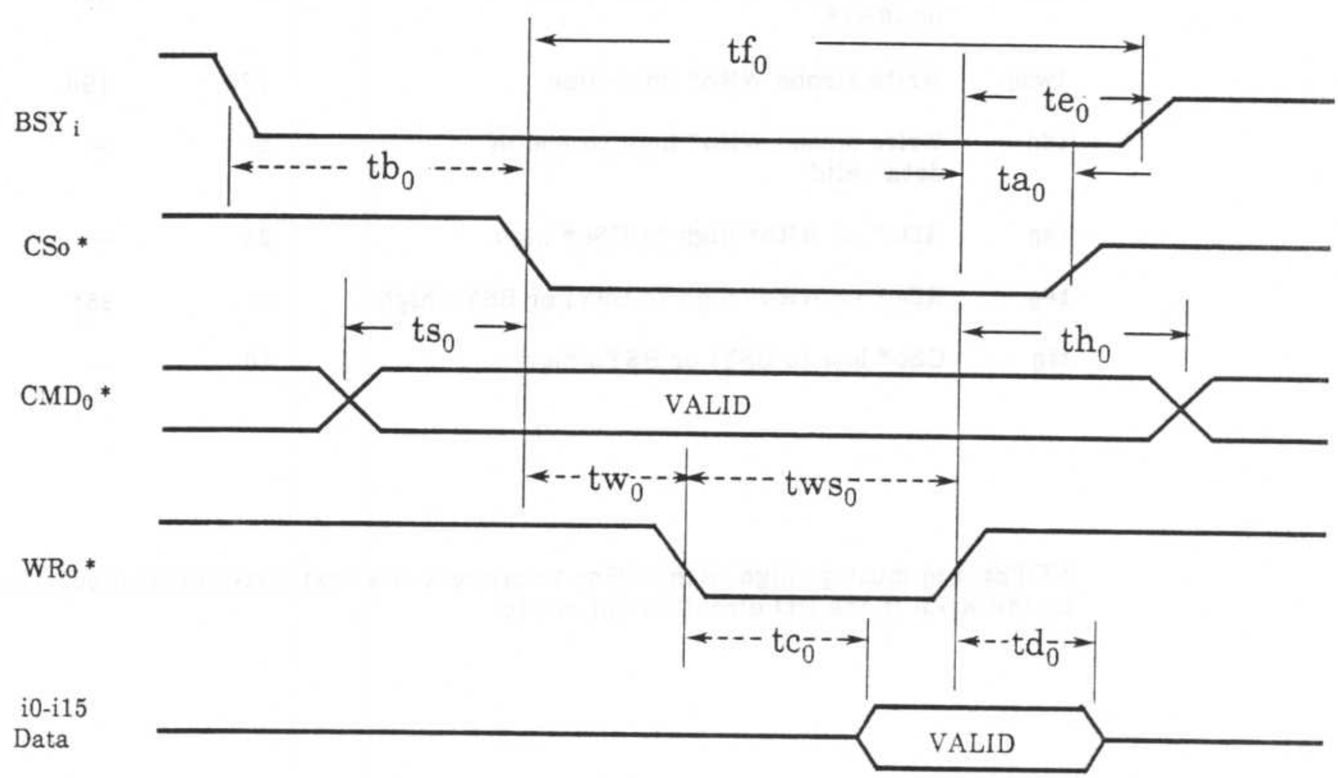


Table 7-7: I/O Port Timing Characteristics

Symbol	Parameter	Min(nS)	Max(nS)
tb_0	I/O BSYo or BSYi low to CSo* low	10	--
ts_0	CMDo* set up time before CSo* low	0	--
tr_0	CSo* low to RDo* strobe low	0	--
tv_0	RDo* to I/O data valid on i0-i15	--	72
trs_0	Hold time of RDo* strobe (RDo* low to RDo* high)	130	190
tp_0	Read strobe RDo* high to end of data valid	8	--
th_0	CMDo* hold time after RDo* or WRo* high	25	--
tw_0	CSo* low to write strobe WRo* low	0	--
tc_0	WRo* low to valid data from KGM on i0-i15	--	38
tws_0	Write strobe WRo* hold time	120	190
td_0	Write strobe WRo* high to end of data valid	9	--
ta_0	RDo* or WRo* high to CSo* high	25	--
te_0	RDo* or WRo* high to BSYi or BSYo high	--	95*
tf_0	CSo* low to BSYi or BSYo high	10	--

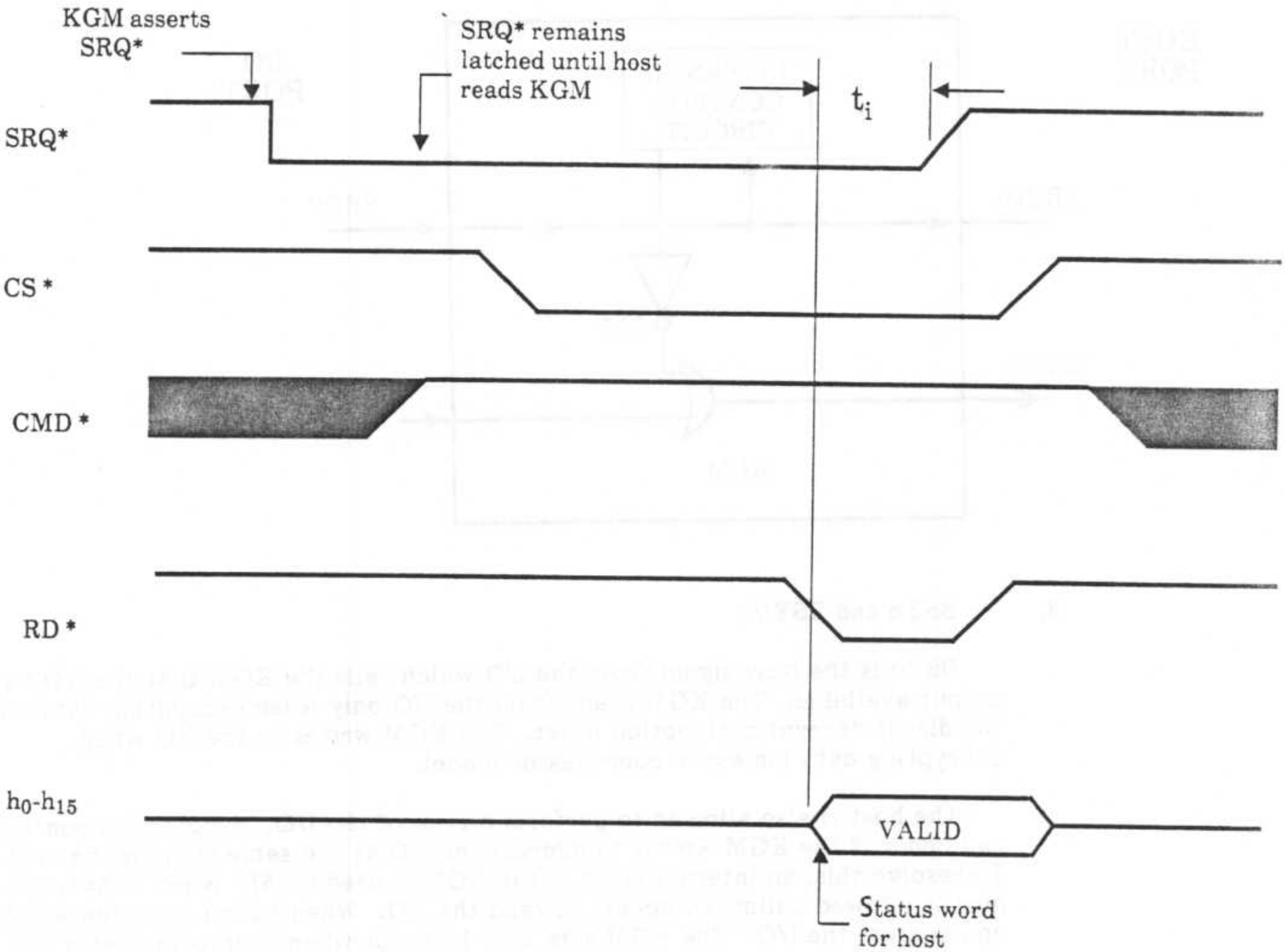
NOTE: te_0 must go high within 95ns to prevent the next possible read or write by the KGM if the I/O circuit is not ready.

c. Service Requests and Busy Signals

1. Service Requests to the host (SRQ*).

The KGM sends service requests (SRQ*) to the host notifying the host that a status word is in the KGM output buffer. SRQ*, when asserted, is latched and remains asserted until the host reads the KGM (CMD*=1). Reading the KGM resets the latch and removes SRQ*. Figure 7-10 shows the timing for SRQ*.

Fig. 7-10: Host Interrupt Timing

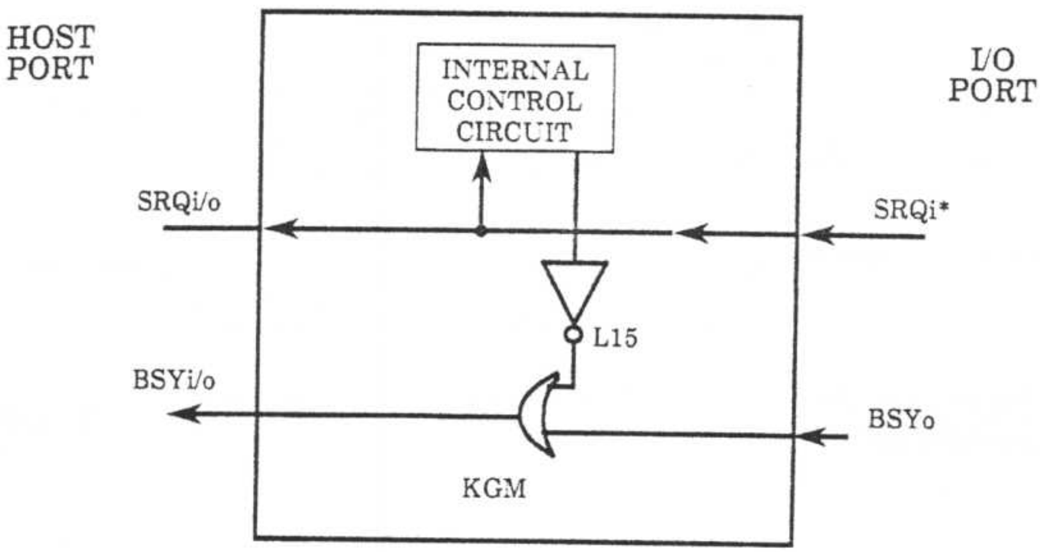


Symbol	Parameter	Min(nS)	Max(nS)
ti	RD* low (CS*=0, CMD*=1) to SRQ*=1	--	67

2. Service Requests from I/O Circuit (SRQi/o*) to host.

Interrupts generated by the I/O (SRQi*) are passed through the KGM and appear at the host port as SRQi/o*. Internally within the KGM, SRQi* causes the KGM to pause and allows the host to read the I/O. KGM activities will resume as soon as SRQi*=1. It is envisioned that when SRQi*=0 occurs, the host will want to read the I/O; and when the host reading of the I/O is completed, the I/O will make SRQi*=1 and allow the KGM to operate. See also Figure 7-11 and discussion at BSYi and BSYi/o below.

Figure 7-11 SRQi*, SRQi/o*, BSYo, BSYi/o Circuits



3. BSYo and BSYi/o

BSYo is the busy signal from the I/O which tells the KGM that the I/O has output available. The KGM reads from the I/O only when decrypting data and the direct decrypt path option is set. The KGM writes to the I/O when encrypting data (unless in coprocessor mode).

The host is also allowed to perform a read of the I/O. A potential conflict can occur if the KGM wishes to address the I/O at the same time as the host. To resolve this, an internal latch in the KGM is used (L15). When L15=1, the host is allowed unlimited access to read the I/O. When L15=0, only the KGM can address the I/O. The KGM sets L15=1 except when decrypting (with the direct decrypt path set) or encrypting (when not in coprocessor mode).

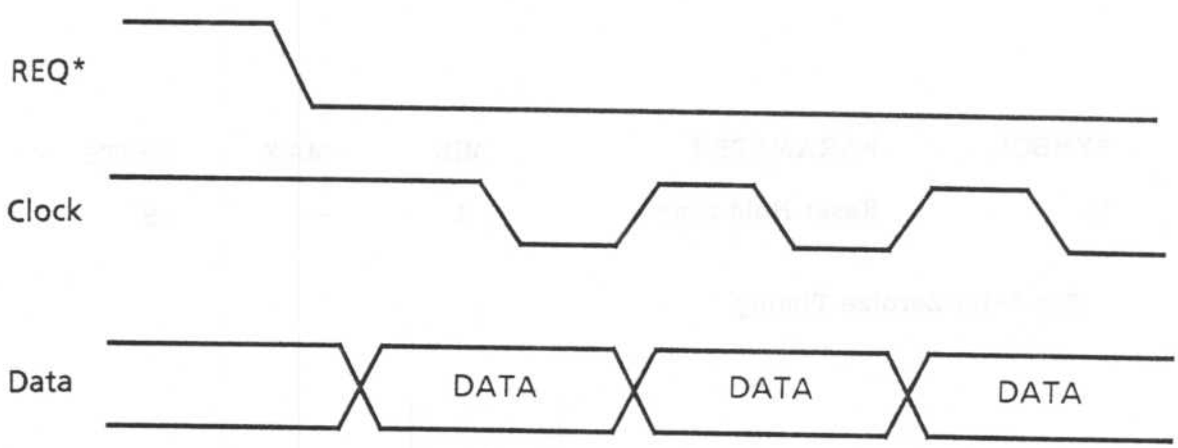
L15 is combined with BSYo to form BSYi/o as shown in Figure 7-11. As can been seen from the Figure, BSYi/o=1 when either BSYo=1 or L15=1. Thus when the KGM is in a mode which addresses the I/O port (L15=1), BSYi/o will always equal 1.

When SRQi*=0 as discussed above, and SRQi/o*=0, the KGM software will cause the current operation to pause and will set L15=0 to allow a host read of the I/O port. This will make BSYi/o = BSYo and the host can read the I/O.

d. Fill Port

The fill port is configured to be compatible with the existing KOI-18 fill device and the DTD. Note that a logic 1 is low and a logic 0 is high.

Fig. 7-12: Fill Port Timing

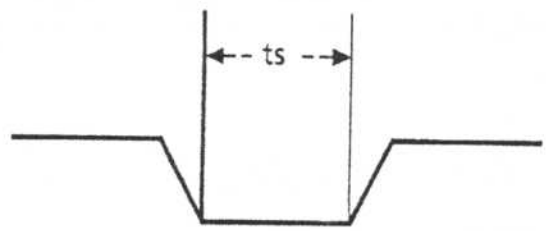


SYMBOL	PARAMETER	MIN	NOM	MAX	UNITS
f	Clock Frequency	--	3.6	10	KHz

e. Housekeeping Port

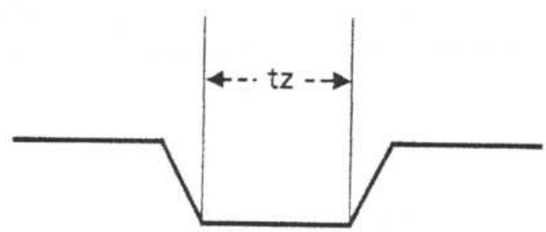
During self test, the alarm line ALARM* toggles.

Fig. 7-13: Reset Timing



SYMBOL	PARAMETER	MIN	MAX	UNITS
ts	Reset Hold time	1	--	uS

Fig. 7-14: Zeroize Timing



SYMBOL	PARAMETER	MIN	MAX	UNITS
tz	Zeroize Hold Time	1	--	uS

f. Cryptographic Ignition Key (CIK) Port

Fig. 7-15: CIK Port Timing

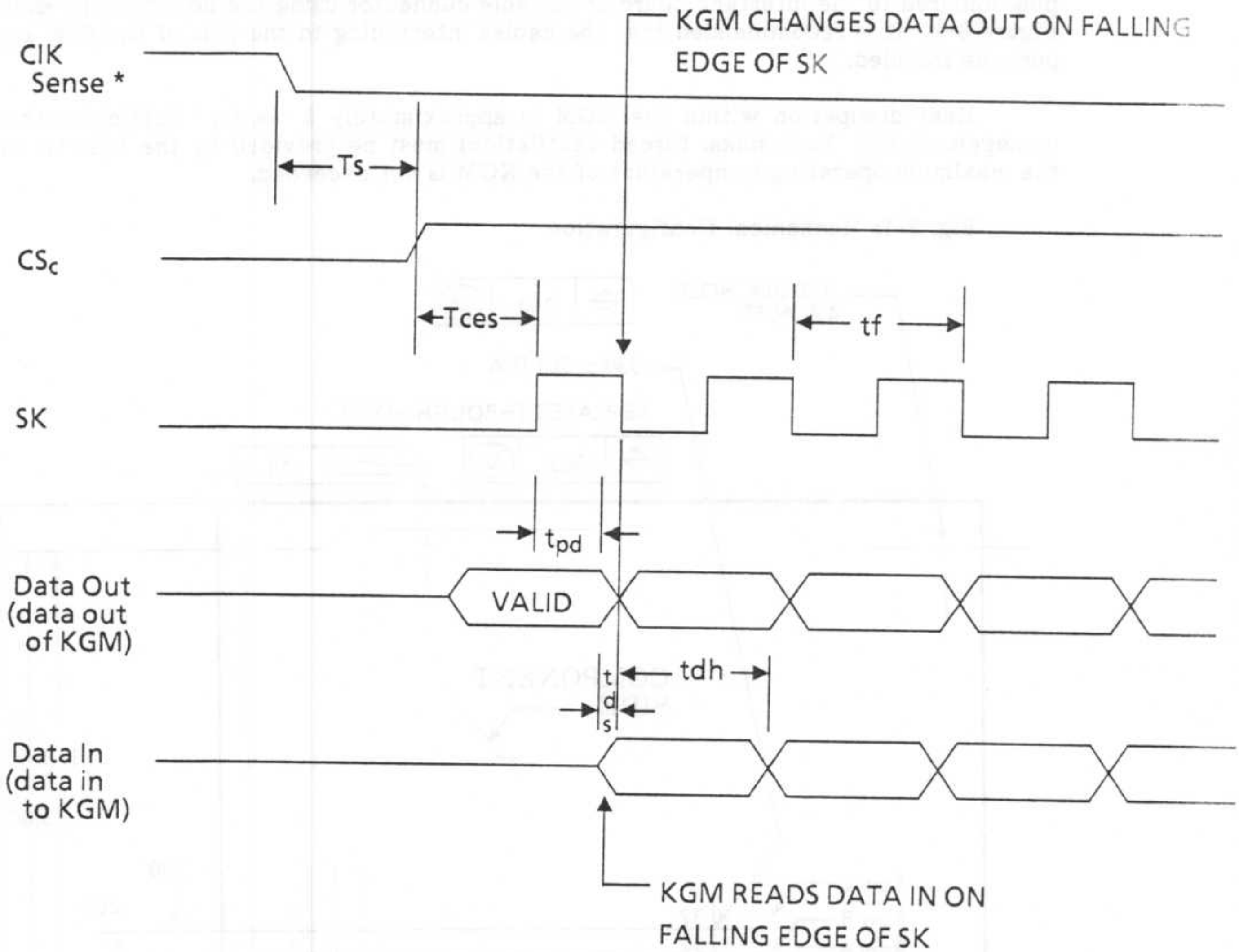


Table 7-8: CIK Port Timing Characteristics

SYMBOL	PARAMETER	MIN	NOM	MAX	UNITS
ts	Set up time before CSc=1	0	-	-	uS
tces	Set up time CSc=1 to SK high	1	-	-	uS
tds	DI Set Up	0.2	-	-	uS
tdh	DI Hold	0.1	-	-	uS
tpd	SK -> high to Data Valid	3	-	-	uS
tf	SK period	-	20	-	uS

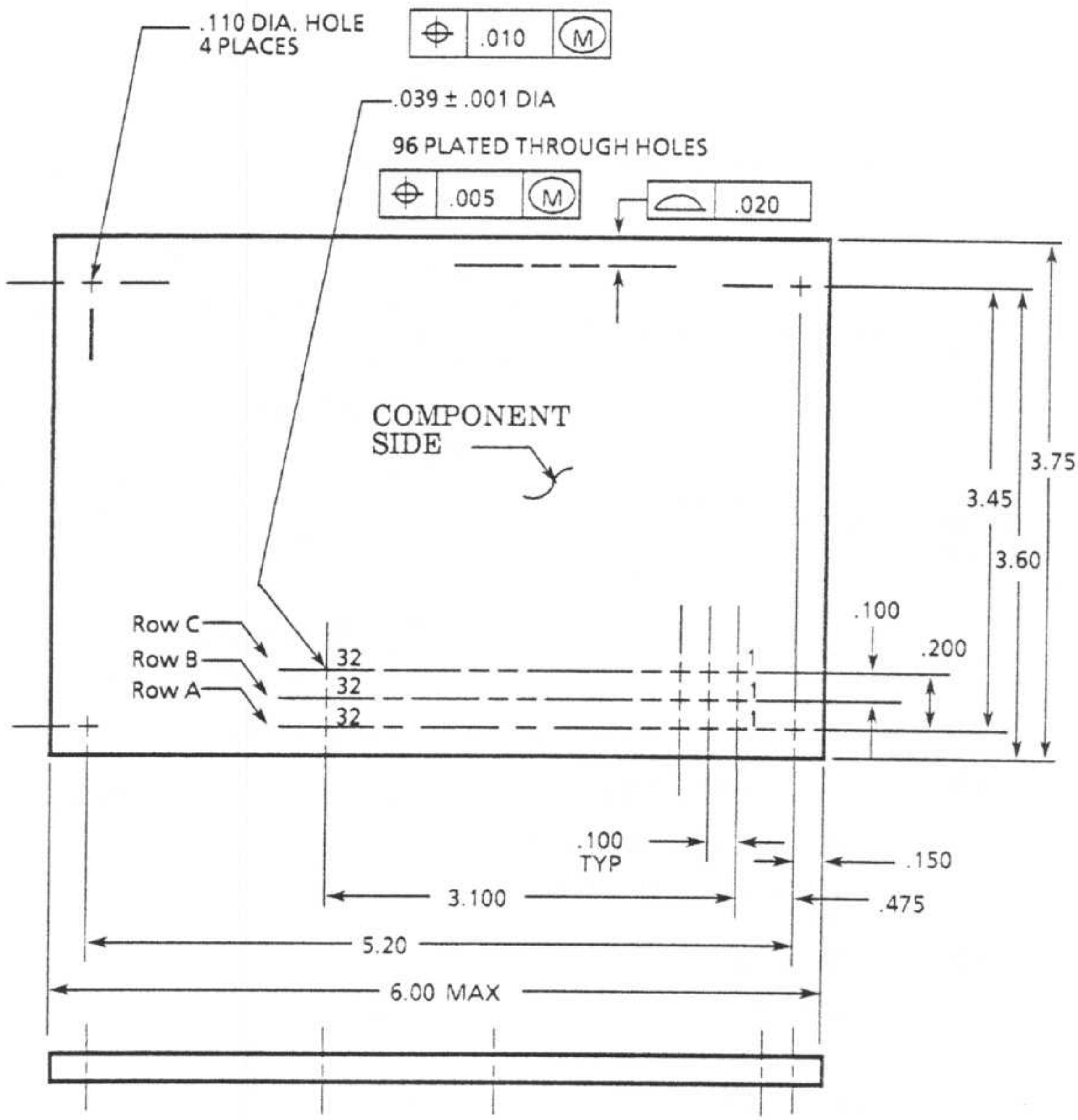
For further data see NMC 9346 data sheet

8. MECHANICAL INTERFACE SPECIFICATIONS

The KGM is a multilayer printed wiring board configuration. It is designed to mount on an interface board resident within the host equipment and provided by the host. Figure 8-1 shows the mechanical configuration. All connections to the KGM are through pins soldered to the interface board or suitable connector using the hole pattern shown in Figure 8-1. It is recommended that the cables interfacing to the pins of the CIK and fill ports be shielded.

Heat dissipation within the KGM is approximately 5 watts. Sufficient thermal management (ie. heat sinks, forced ventilation) must be provided by the host to ensure the maximum operating temperature of the KGM is not exceeded.

Fig. 8-1: Mechanical Configuration



9. GLOSSARY

BLACK designation - A designation applied to all telecommunications circuits, components, equipments, and systems which handle only encrypted or unclassified signals; and to telecommunications areas in which no classified signals occur.

Bypass Audit Word - A vehicle to obtain the count, stored internally on EEPROM, of the number of words bypassed (both command and data), along with the bypass control word and a flag identifying whether the KGM is configured as a Node or System Manager.

Bypass Control Word - A 64 bit word stored internally on EEPROM which is used by the KGM to verify and control the amount of command and data bypass permitted by the host.

byte - An 8 bit data unit.

Controlled Cryptographic Item (CCI) - A marking applied to those unclassified end items and assemblies which perform critical COMSEC functions, and require access controls and physical security protection to assure their continued integrity.

Crypto-Ignition key (CIK) - A device used to alter a cryptovariable to provide physical security for the cryptovariable.

data block - The amount of data which is encrypted/decrypted as a unit or frame in a particular mode.

decryption - The process that converts enciphered text to plain text by means of a cipher system.

encryption - The process that converts plain text into enciphered text by means of a cipher system.

Executive -	A state from which commands are accepted, validated and executed.
fill -	The set of bits representing the key packet.
fill device -	A family of devices developed to read in, transfer, and store keying variables. Example is the KOI-18/TSEC General Purpose Tape Reader.
key -	A sequence of random binary bits used to initially set up and periodically change the operations performed in crypto-equipment for purposes of encrypting or decrypting.
KEK (Key Encryption Key) -	A key that is used only for encrypting keys (i.e., Storage key, Unique key)
key generator -	A device or algorithm which employs a series of mathematical rules to deterministically produce a pseudo-random sequence of bits.
key packet -	A 256 bit data word containing a key along with its associated accounting information.
key type -	Information contained in a field within the key packet which indicates to the KGM the intended use of the key (i.e. Traffic key or Key Encryption key).
key updating -	A periodic key modification performed automatically or manually.
message indicator (MI) -	A sequence of bits needed along with the key to resume encryption/decryption in certain modes of operation.
Node -	The normal configuration of the KGM for encrypting/decrypting traffic. All KGM functions are enabled with the exception of those set aside for the System Manager.
OTAR (Over-The-Air Rekey) -	The encrypted transmission of cryptographic key from a remote source. Sometimes referred to as "rekey".
parity -	A group of bits used to verify that the cryptovariable has been properly transferred into a cryptographic device.

RED -	A designation applied to telecommunications circuits, components, equipments, and systems which handle classified plain text or other information which requires protection during electrical transmission, and to areas in which such information exists.
RSK (Random Seed Key) -	Key used during the randomization process.
Storage Key -	A key used to encrypt/decrypt other keys for safe storage in the host computer.
System Manager -	A particular configuration of the KGM which enables the generation of remote rekey messages and Bypass Control Words, along with the ability to read Bypass Audit Words. The System Manager maintains all the capabilities of a Node.
TEMPEST -	A short name referring to investigation and studies of compromising emanations. It is often used synonymously for the term "compromising emanations", e.g., TEMPEST tests, TEMPEST inspections.
Traffic Key -	A key used to encrypt/decrypt text for communications purposes.
Unique Key -	A key used to encrypt/decrypt remote Traffic keys (not used for data communications). This key is known only to the crytodevice and the network key manager.
Unwrap -	The process of decrypting a key for storage purposes.
Word -	A memory unit equal to the microcontrollers word size (16 bits).
Wrap -	The process of encrypting a key for storage purposes.
Zeroize -	To remove or eliminate the cryptovariable or other information from a cryptodevice.

RED - A designation applied to telecommunications circuits, components, equipments, and systems which handle classified plain text or other information which requires protection during electrical transmission, and to areas in which such information exists.

RSK (Random Seed Key) - Key used during the randomization process.

Storage Key - A key used to encrypt/decrypt other keys for safe storage in the host computer.

System Manager - A particular configuration of the KGM which enables the generation of remote rekey messages and Bypass Control Words, along with the ability to read Bypass Audit Words. The System Manager maintains all the capabilities of a Node.

TEMPEST - A short name referring to investigation and studies of compromising emanations. It is often used synonymously for the term "compromising emanations", e.g., TEMPEST tests, TEMPEST inspections.

Traffic Key - A key used to encrypt/decrypt text for communications purposes.

Unique Key - A key used to encrypt/decrypt remote Traffic keys (not used for data communications). This key is known only to the crytodevice and the network key manager.

Unwrap - The process of decrypting a key for storage purposes.

Word - A memory unit equal to the microcontrollers word size (16 bits).

Wrap - The process of encrypting a key for storage purposes.

Zeroize - To remove or eliminate the cryptovariable or other information from a cryptodevice.