MOTOROLA

MC6859

Advance Information

DATA SECURITY DEVICE

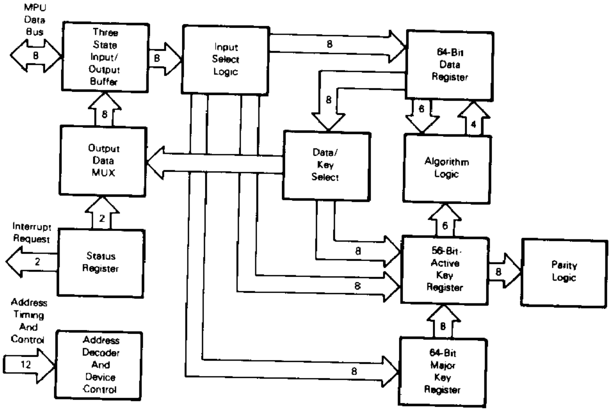
The MC6859 Data Security Device (DSD) is a monolithic MOS integrated circuit designed to be integrated into a wide range of equipment requiring protection of data by the employment of cryptographic measures.

The cryptographic algorithm utilized by the device is the Data Encryption Standard (DES) as adopted by the U.S. Department of Commerce, National Bureau of Standards (NBS), in publication FIPS PUB 46 (1-15-1977).

Through the use of flexible on-chip control and status circuitry and external control lines, the DSD provides direct capability of adapting the functional implementation of the DES algorithm for various specific system requirements for data protection.

- Direct Compatibility with the M6800 Microprocessor Family
- Data Encryption Standard Algorithm
- Two Separate Interrupt Output Lines for Program Controlled Interrupt Capability
- Up to 400 KBPS Throughput Rate of 64-Bit Block Cipher (Exclusive of Software Overhead)
- TTL Compatible
- Single +5 V Power Supply

DATA SECURITY DEVICE BLOCK DIAGRAM

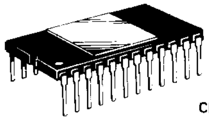


This document contains information on a new product. Specifications and information herein are subject to change without notice.

MOS

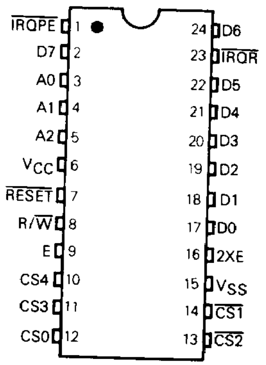
DEPLETION LOAD
(N-CHANNEL, SILICON-GATE)

DATA SECURITY DEVICE



L SUFFIX
CERAMIC PACKAGE
CASE 716

PIN ASSIGNMENT


$\overline{\text{IRQPE}}$ 1
D7 2
A0 3
A1 4
A2 5
V_{CC} 6
$\overline{\text{RESET}}$ 7
R/$\overline{\text{W}}$ 8
E 9
CS4 10
CS3 11
CS0 12
24 D6
23 $\overline{\text{IRQR}}$
22 D5
21 D4
20 D3
19 D2
18 D1
17 D0
16 2XE
15 V_{SS}
14 $\overline{\text{CS1}}$
13 $\overline{\text{CS2}}$

NOTICE

This product may not be exported without prior approval from the U.S. Department of State, Office of Munitions Control.

MAXIMUM RATINGS

Characteristics	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to +7.0	V
Input Voltage	V_{in}	−0.3 to +7.0	V
Operating Temperature Range MC6859	T_A	T_L to T_H 0 to 70	°C
Storage Temperature Range	T_{stg}	−55 to +150	°C

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Ceramic Package	θ_{JA}	60	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precuations be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. Reliability of operation is enhanced if unused inputs are tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{CC}).

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \bullet \theta_{JA}) \qquad (1)$$

Where:

$T_A \equiv$ Ambient Temperature, °C

$\theta_{JA} \equiv$ Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D \equiv P_{INT} + P_{PORT}$

$P_{INT} \equiv I_{CC} \times V_{CC}$, Watts – Chip Internal Power

$P_{PORT} \equiv$ Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$$P_D = K \div (T_J + 273°C) \qquad (2)$$

Solving equations 1 and 2 for K gives:

$$K = P_D \bullet (T_A + 273°C) + \theta_{JA} \bullet P_D^2 \qquad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

DC ELECTRICAL CHARACTERISTICS (V_{CC}=5.0 Vdc ±5%, V_{SS}=0, T_A=T_L to T_H, unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage		V_{IH}	$V_{SS}+2.0$	–	V_{CC}	V
Input Low Voltage		V_{IL}	$V_{SS}-0.3$	–	$V_{SS}+0.8$	V
Input Leakage Current (V_{in}=0 to 5.25 V)		I_{in}	–	1.0	2.5	µA
Three-State (Off State) Input Current (V_{in}=0 to 5.25 V)	D0-D7	I_{IZ}	–	2.0	10	µA
Output High Voltage (I_{Load}=−205 µA) (See Figure 2)	D0-D7	V_{OH}	$V_{SS}+2.4$	–	–	V
Output Low Voltage (I_{Load}=1.6 mA) (I_{Load}=3.2 mA) (See Figure 2)	D0-D7 $\overline{IRQPE}$, $\overline{IRQR}$	V_{OL}	– –	– –	$V_{SS}+0.4$ $V_{SS}+0.6$	V
Output Leakage Current (Off State) (V_{OH}=2.4 V)	$\overline{IRQPE}$, $\overline{IRQR}$	I_{OZ}	–	1.0	10	µA
Internal Power Dissipation (Measured at T_A=0°C)		P_{INT}	–	–	1000	mW
Input Capacitance (V_{in}=0, T_A=25°C, f=1.0 MHz)	D0-D7 All Others	C_{in}	– –	– –	12.5 7.5	pF
Output Capacitance (V_{in}=0, T_A=25°C, f=1.0 MHz)	$\overline{IRQPE}$, $\overline{IRQR}$	C_{out}	–	–	50	pF

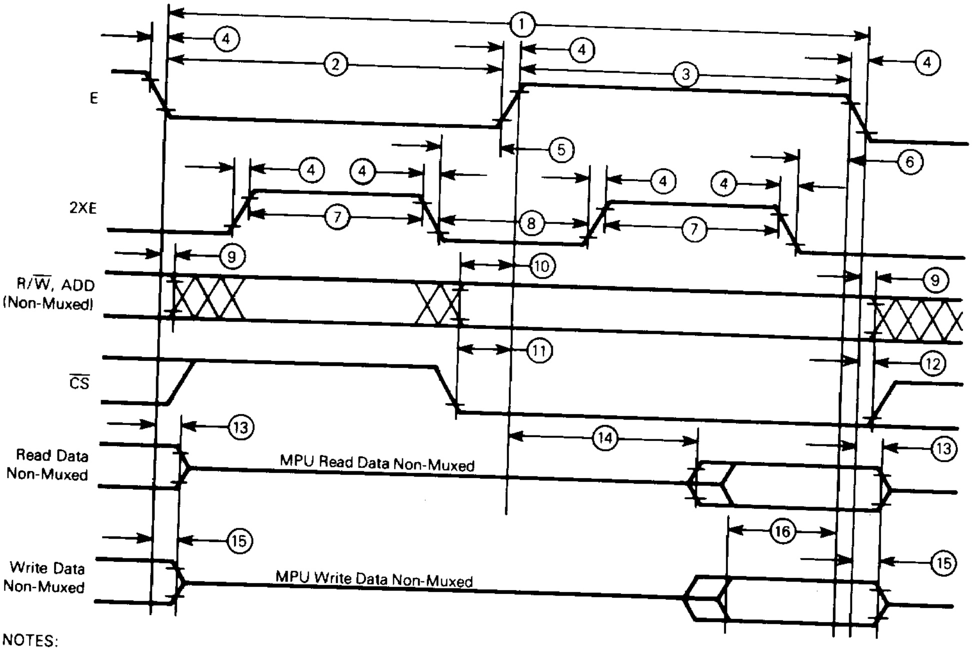
BUS TIMING CHARACTERISTICS

Ident. Number	Characteristic	Symbol	Min	Max	Unit
1	Cycle Time	t_{cyc}	1.0	10	μs
2	Pulse Width, E Low	PW_{EL}	430	–	ns
3	Pulse Width, E High	PW_{EH}	450	–	ns
4	Clock Rise and Fall Time	t_r, t_f	–	25	ns
5	2XE to E High Delay Time	t_{DH}	0	–	ns
6	2XE to E Low Delay Time	t_{DL}	0	–	ns
7	Pulse Width 2XE Low	PW_{2L}	210	–	ns
8	Pulse Width 2XE High	PW_{2H}	220	–	ns
9	Address Hold Time	t_{AH}	10	–	ns
10	Address Setup Time Before E	t_{AS}	80	–	ns
11	Chip Select Setup Time Before E	t_{CS}	80	–	ns
12	Chip Select Hold Time	t_{CH}	10	–	ns
13	Read Data Hold Time	t_{DHR}	20	50*	ns
14	Output Data Delay Time	t_{DHW}	–	290	ns
15	Write Data Hold Time	t_{DDR}	10	–	ns
16	Input Data Setup Time**	t_{DSW}	165	–	ns
17	Interrupt Release Time	t_{IR}	1200	–	ns

*The data bus output buffers are no longer sourcing or sinking current by t_{DHR} maximum (high impedance).

**Data is latched into the internal registers on the falling edge of 2XE and while enable is high. Therefore, for system considerations, $t_{DSW} = t_{DSW1} + t_D + 2X\ t_f$. Minimize t_D to ensure operation at 1 MHz. t_{DSW1} is the data setup time for the "AK6" mask set.

FIGURE 1 — BUS TIMING



NOTES:

1. Voltage levels shown are $V_L \leq 0.4$ V, $V_H \geq 2.4$ V, unless otherwise specified.
2. Measurement points shown are 0.8 V and 2.0 V, unless otherwise specified.

FIGURE 2 — BUS TIMING TEST LOADS

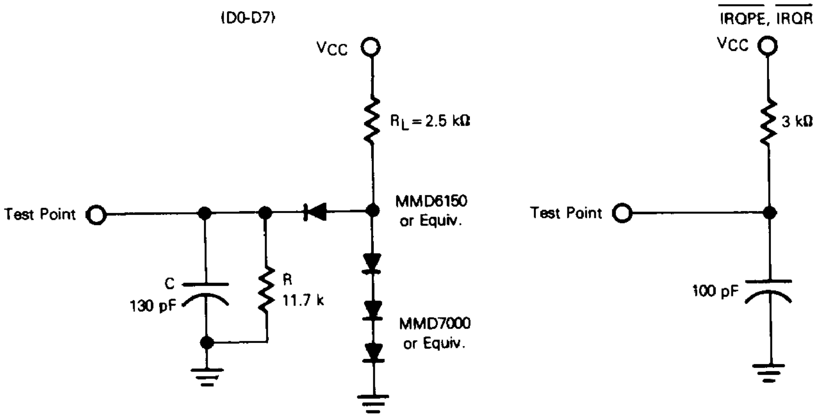
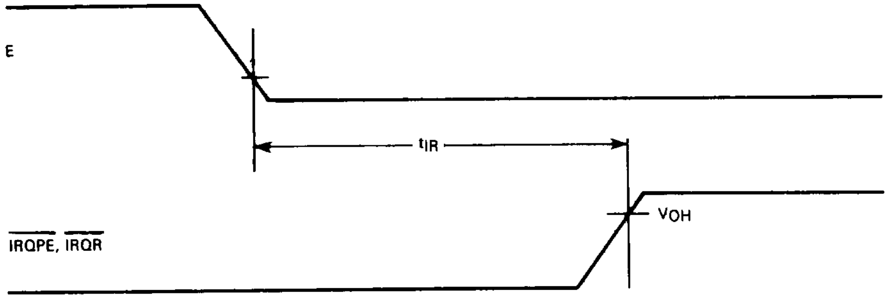


FIGURE 3 — INTERRUPT RELEASE TIME



Note: Timing measurements are referenced from a low voltage of 0.8 volts and a high voltage of 2.0 volts, unless otherwise noted.

BUS INTERFACE

The MC6859 Data Security Device (DSD) interfaces to the M6800 bus via an 8-bit bidirectional data bus, five chip select lines, a read/write (R/$\overline{\text{W}}$) line, an external $\overline{\text{RESET}}$ line, three register select lines, an Enable (System ϕ2) line, a 2XEnable (2XE) clock line, and two interrupt request lines. These signals permit the M6800 MPU to control the DSD and perform data transfers between the two.

Bidirectional Data Bus (D0-D7) – The bidirectional data lines (D0-D7) allow the transfer of information between the MPU and DSD. The data bus input/output drivers are three-state devices which remain in the high-impedance (off) state except when the MPU performs a DSD read or write operation.

Chip Select (CS0, $\overline{\text{CS1}}$, $\overline{\text{CS2}}$, CS3, and CS4) – These five signals are used to activate the data bus interface and allow DSD data transfers. When CS0 = CS3 = CS4 = 1 and $\overline{\text{CS1}}$ = $\overline{\text{CS2}}$ = 0, the device is selected.

Read/Write (R/$\overline{\text{W}}$) – With the DSD selected, this input controls the direction of data transfer on the data bus. When R/$\overline{\text{W}}$ is high, data in the DSD is read by the MPU on the trailing edge of E. A low state on the R/$\overline{\text{W}}$ line enables data transfer from the MPU on the trailing edge of the 2XE signal.

Enable (E) and 2XEnable (2XE) – The rising edge of the Enable input initiates data transfer from the DSD to the MPU during a read cycle. The falling edge of the Enable input latches MPU data into the DSD during a write cycle. The 2XE input is used in processing the encryption/decryption algorithm for all mask sets. E and 2XE are completely asynchronous. See section on Mask Sets for exceptions on prior revision of the DSD.

Reset ($\overline{\text{RESET}}$) – This input signal is used to initialize the internal control logic, status flags, and counters of the DSD. The contents of the active key register and major key register remain unchanged. The $\overline{\text{RESET}}$ function should be coupled with the system power-on reset to provide orderly system initialization. It may also be used as a master reset to the chip during system operation.

To abort the encryption algorithm before the required 320 clock cycles (2XE) have occurred, it is necessary to provide a $\overline{\text{RESET}}$ signal or a software reset command to the DSD. When this occurs, information in the data register and active key register is no longer valid. The contents of the major key register are unaffected.

Address Lines (A0, A1, A2) – These inputs are used in conjunction with the R/$\overline{\text{W}}$ line to select one of eleven possible DSD operations, as shown in Tables 1 and 2. The DSD is accessed via MPU read and write operations in much the same manner as a memory device.

NOTE:

Instructions performing operations directly on memory should not be used when the DSD is accessed. Since the DSD uses the R/$\overline{\text{W}}$ line as an additional register select input, read-modify-write type instructions will conflict with normal operation of the Data Security Device.

Modes – Operational and control modes are invoked by addressing DSD registers at the addresses in Tables 1 and 2.

TABLE 1 – OPERATIONAL MODES

Control Address		Operational Mode
A0, A1, A2	R/$\overline{\text{W}}$	
0 0 0	W	Write Data/"C" Key Operation (1st 7 bytes)
*1 0 1	W	Encipher Data
*0 0 1	W	Decipher Data
0 0 1	R	Read Data
0 1 0	R	Read Status

TABLE 2 – CONTROL MODES

Control Address		Control Mode
A0, A1, A2	R/$\overline{\text{W}}$	
1 0 0	W	Reset/Initialize
0 1 0	W	Enter Major Key
1 1 0	W	Enter Plain Secondary Key
*0 1 1	W	Decipher Secondary Key
*1 1 1	W	Encipher Secondary Key
1 0 0	R	Transfer Major Key

*Instruction initiated after eighth byte of Key Block entry.

Interrupt Requests – These open drain outputs are used to convey internal DSD status information to the MPU.

Ready Interrupt Request ($\overline{\text{IRQR}}$) – This active low output signals the MPU that the DSD is ready to initiate another operation. The $\overline{\text{IRQR}}$ signal will be inactive during encryption/decryption or key transfer.

Parity Error Interrupt Request ($\overline{\text{IRQPE}}$) – This active low output is used to signal the MPU that the DSD has detected a parity error. The $\overline{\text{IRQPE}}$ signal will remain low until a hardware or software reset is received.

DSD FUNCTIONAL DESCRIPTION

The MC6859 Data Security Device appears to an MPU system as an interface adapter device. An example of a system with the encryption function is shown in Figure 4.

Internal construction of the DSD is illustrated by the block diagram. The device consists of a single 8-bit data bus buffer with three-state operation, through which data may be entered into:

1) the 56-bit active key register
2) the 64-bit major key register
3) the 64-bit data register

Output data from the status register or the data register is also switched through the data bus buffers.

At the bus interface, the DSD data register appears as eight addressable memory locations to the MPU, through which the operational mode of the chip may be selected, chip status monitored, key or data written into.the device, and data read from the device.

OPERATING MODES

As shown in Table 1, the operation of the DSD is split into five major modes:

1) status readout
2) loading of data or encrypted key
3) data encryption
4) data decryption
5) data readout

These and additional control modes are activated by three address input lines and a read/write input line.

Read Status — Only two bits are used in the status readout, D7 = Parity Error (PE) and D6 = $\overline{\text{READY}}$. The remaining six bits are always read as logic zeros. A read of the status register does not change these bits.

The PE flag is set when a parity error is detected while loading either a major or secondary key or when the active key is checked during algorithm operation. The PE flag remains set and the $\overline{\text{IRQPE}}$ signal will remain low until a hardware/software reset is received.

The $\overline{\text{READY}}$ flag is set and the $\overline{\text{IRQR}}$ output goes high whenever the device is processing a block of data. The flag is cleared, pulling the $\overline{\text{IRQR}}$ output low, whenever the DSD is not encoding/decoding data or transferring major key. $\overline{\text{IRQR}}$ may be tied to $\overline{\text{IRQ}}$ of a M6800 family processor for interrupt-driven encryption if no other peripherals share the $\overline{\text{IRQ}}$ line.

Encipher Data — To encipher an 8 byte block of data, the first seven bytes are written to the Write Data/"C" Key register. The eighth byte is written to the Encipher Data register. This automatically initiates the encryption process. Data is always processed using the current Active Key. During algorithm operation, the DSD constantly performs parity checking on the contents of the active key register. The busy flag will be set during encryption and then reset when the algorithm has finished. Completion requires 320 cycles of 2XE. During this time the DSD will ignore all external commands except status read, hardware reset and software reset.

Decipher Data — This process is identical to encipher data except that the eighth byte is written to the Decipher Data register. During decipher or encipher only a read status register, hardware reset, or software reset will be recognized. All other commands will be ignored.

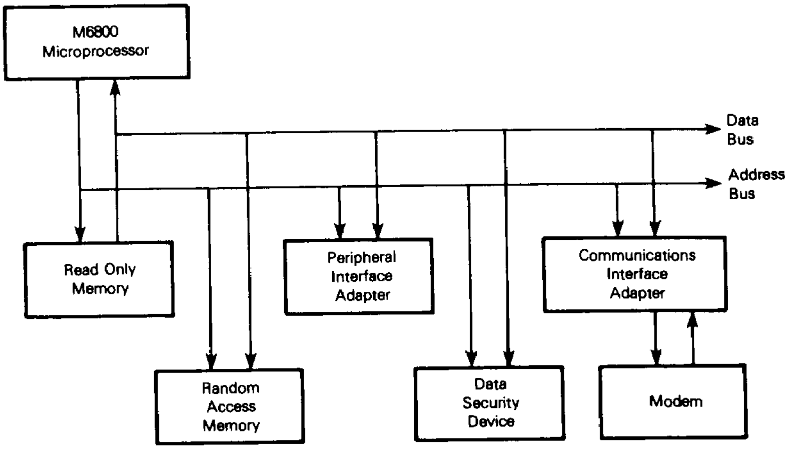
Read Data — This command is normally executed upon completion of the encipher/decipher algorithm (indicated by $\overline{\text{READY}}$ = 0). A read prior to completion of busy will result in all zeros being read from D0-D7. As each byte of data is read, zeros are automatically shifted into the data register to ensure data security.

CONTROL MODES

Shown in Table 2 are the control modes which facilitate programming of the primary and secondary keys.

Reset/Initialize — The DSD may be software reset by writing the reset/initialize command at any time the data bus is ignored. Like the hardware reset, this command initializes the internal control logic, status flags, and counters without altering the contents of the active key register or the major key register. If a hardware or software reset is issued during the algorithm processing, the information in the data register and active key register will no longer be valid. However, the contents of the major key register are not affected.

FIGURE 4 — M6800 MICROCOMPUTER FAMILY BLOCK DIAGRAM



Load Major Key – An unencrypted key will be entered into both the active key register and the major key register when eight consecutive bytes are written into the Enter Major Key Register. Parity error checking is automatically performed.

Load Plain Secondary Key – An unencrypted key may be loaded into the active key register and simultaneously checked for parity errors by writing eight consecutive bytes into the Enter Plain Secondary Key Register. The Major Key Register is unaffected.

Encipher Secondary Key – After a secondary key is loaded, it can be enciphered or deciphered (the source of an encrypted key is usually another DSD). A secondary key may be enciphered by loading the first seven bytes of plain text to the Write Data/"C" Key register. The eighth byte is entered to the Encipher Secondary Key register. This causes the secondary key to be enciphered using the current major key and automatically loaded into the Active Key register and checked for parity. This operation requires 328 cycles of 2XE.

Decipher Secondary Key – This function is similar to the Encipher Secondary Key operation. The first seven bytes of the key are loaded into the Write Data/"C" Key register. The eighth byte is entered by addressing the Decipher Secondary Key register. The secondary key is then deciphered using the current major key and automatically loaded into the Active Key register and checked for parity. This operation requires 328 cycles of 2XE.

Transfer Major Key – The contents of the Major Key register will be transferred to the Active Key register by a read of the Transfer Major Key register. The data bus is ignored. The Major Key register remains unchanged. This operation requires eight cycles of 2XE.

KEY CONVENTIONS

The key used for coding is a 56-bit data word plus eight bits of odd parity. In the DSD seven bits of key and the parity bit make up a key character. Eight key characters make up the total key information required by the DSD if parity errors are to be checked via the PE signal. If parity is not needed for some reason, then the parity bit need not be calculated and can be left as a zero. An example key with parity is shown in Table 3.

TABLE 3 – EXAMPLE KEY

Key Character	Hex Value	Binary Value							Parity
Byte 1	7C	0	1	1	1	1	1	0	0
Byte 2	A1	1	0	1	0	0	0	0	1
Byte 3	10	0	0	0	1	0	0	0	0
Byte 4	45	0	1	0	0	0	1	0	1
Byte 5	4A	0	1	0	0	1	0	1	0
Byte 6	1A	0	0	0	1	1	0	1	0
Byte 7	6E	0	1	1	0	1	1	1	0
Byte 8	57	0	1	0	1	0	1	1	1
Data Lines		**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**

TYPICAL SYSTEM OPERATION

For a communications link between a sender and one or more receivers, the following typical sequence might be used to transmit confidential data:

1) A software reset is issued to each DSD by its MPU.
2) The sending MPU loads a major key (eight bytes) into its DSD. This will serve as the active key if a secondary key is not entered.
3) The receiving station must also load this same major key before data transmission can begin. If the current major (or secondary) key is not known in advance, it can be transmitted by the sending MPU, but may not be encoded as the receiving MPU system has no key to decode it by. The MPU at the receiving station must be programmed with the mode and format being used for data transmission so its DSD can process the data correctly. At this point both the transmitting and receiving stations are ready for data transfer.
4) The sending MPU writes eight bytes of data into its DSD which enciphers them.
5) The sending MPU retrieves eight bytes of encrypted data from its DSD and transmits them to the receiving MPU.
6) The receiving MPU writes these eight bytes of data into its DSD to be deciphered.
7) The receiving MPU retrieves eight bytes of data from its DSD in the original plain text form.

Steps four through seven are repeated for each 8-byte block of data to be transmitted. If the major key or secondary key is to be changed, steps two and three must also be carried out.

SECURITY CONSIDERATIONS

The security of a system employing the NBS Data Encryption Standard (DES) depends only upon the key used, not the availability of the algorithm or of equipment used to implement the algorithm. The key is the most critical piece of information in the system and security of the key itself must be maintained both inside and outside the system.

Guidelines to be used in selecting a key are:

- Consider the key to be a single 56-bit number
- Avoid bias in selecting the key
- Change key as frequently as practical

One way to help ensure the security of the key is to make frequent use of secondary keys. Secondary keys can be generated by the sender and distributed selectively to one or more receivers. Since the MC6859 can encipher or decipher secondary keys using the major key, the sender can transmit the secondary key in encrypted form to further ensure system security. However, the receiver must be aware that a secondary key is being transmitted and must decrypt the key if it was sent in encrypted form.

Assuming that secrecy of the key is maintained, it is nearly impossible for an unauthorized user to decode an intercepted message into its original form. Since the DES algorithm utilizes a 56-bit active key, there are 2^{56} (or about 7×10^{16}) possible encrypted messages which must be searched to retrieve the original message. In addition, if the key were changed regularly only a small portion of the message would be retrieved for each successful exhaustive search. Therefore, the basic "block cipher" technique described in the Typical System Operation section is adequate for today's data security applications.

If additional security is required for some reason, several techniques can be used to increase data security. These include:

- Perform multiple encryption and/or decryption using the same key or different keys
- Reverse the algorithm (decipher-transmit-encipher)
- Utilize cipher feedback or other feedback techniques

The process of multiple encryption or decryption is an easy way to effectively increase the size of the key to any desired length. For example, the sender might successively encipher, decipher, and encipher a block of data using one key for the encipher operations and another for the decipher operation. The receiver would then have to decipher, encipher, and decipher the data using the same pair of keys. This technique would greatly increase data security while reducing throughput by a factor of three. Many such multiple encryption combinations are possible.

An easy way to increase security without reducing throughput is to perform the DES algorithm "in reverse." In other words, data or keys can be deciphered by the sender and then enciphered by the receiver to yield the original message. This technique works because the enciphering and deciphering algorithms are "mirror images" of each other.

Many different feedback techniques are available as alternatives to the basic 64-bit block cipher. One of these, known as cipher feedback (CFB), is described below. CFB is a byte-oriented implementation in that only one byte is transmitted at a time. Thus, throughput is reduced by a factor of eight (excluding software overhead). Implementation of the CFB technique is more dependent upon the system configuration than is the block cipher.

CFB ENCIPHER

The basic flow of the CFB encipher procedure is shown in Figure 5.

An initial eight byte fill of the RAM buffer must be done prior to accepting plain text bytes for enciphering. This information can be considered to be a data subset of the key, but may be any combination of eight-bit bytes as long as the deciphering device uses the same initial fill.

After the block of data in the RAM buffer is enciphered, one byte of enciphered data is read from the DSD. This byte is the key byte (Kt+1). The plain text byte (Pt+1) is exclusive ORed with the key byte and the result is the cipher text byte (Ct+1). The cipher text byte is shifted into the bottom of the RAM buffer and now is the newest byte in the block. The oldest previous byte is discarded. The cipher text byte is now available for use. The new RAM buffer block is loaded into the DSD for enciphering and yields the next key for further processing.

CFB DECIPHER

The basic flow of the decipher CFB operation is shown in Figure 6.

The same initial fill as used for enciphering must be used to initialize the decipher RAM buffer. The same key used to encipher must also be used to load the DSD active key register prior to receiving cipher text bytes. When a cipher text byte is received it is exclusive ORed with the key byte generated by the DSD and the result is the plain text data byte. The received cipher text byte is shifted into the RAM buffer and becomes the newest RAM buffer byte. The oldest RAM buffer byte is discarded and the eight byte RAM buffer is loaded into the DSD for block deciphering. One byte of the DSD data register is read out and this byte becomes the key byte for the next cipher text byte received.

FIGURE 5 — CFB ENCIPHER DATA FLOW (TRANSMITTING)

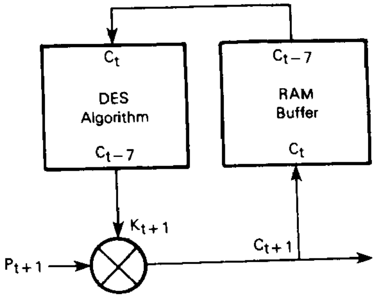
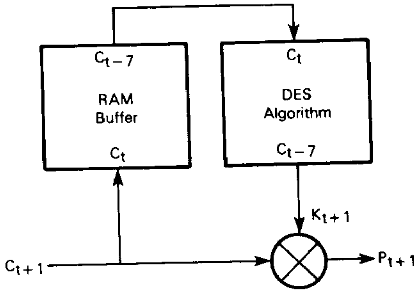


FIGURE 6 — CFB ENCIPHER DATA FLOW (RECEIVING)



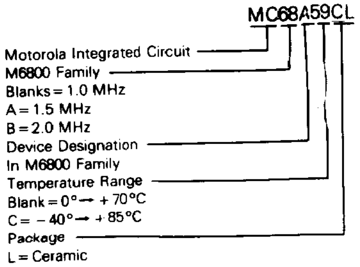
To purchase a copy of the NBS Data Encryption Standard ask for the Federal Information Processing Standards (FIPS) Publication, FIPSP 46 at the following address:

National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

Circuit diagrams external to or containing Motorola products are included as a means of illustration only. Complete information sufficient for construction purposes may not be fully illustrated. Although the information herein has been carefully checked and is believed to be reliable, Motorola assumes no responsibility for inaccuracies. Information herein does not convey to the purchaser any license under the patent rights of Motorola or others.

The information contained herein is for guidance only, with no warranty of any type, expressed or implied. Motorola reserves the right to make any changes to the information and the product(s) to which the information applies and to discontinue manufacture of the product(s) at any time.

ORDERING INFORMATION



MC68A59CL

Motorola Integrated Circuit

M6800 Family

Blanks = 1.0 MHz

A = 1.5 MHz

B = 2.0 MHz

Device Designation

In M6800 Family

Temperature Range

Blank = 0° → +70°C

C = −40° → +85°C

Package

L = Ceramic

BETTER PROGRAM

Better program processing is available on all types listed. Add suffix letters to part number.

Level 1 add "S"   Level 2 add "D"   Level 3 add "DS"

Level 1 "S" = 10 Temp Cycles − (−25 to 150°C);
Hi Temp testing at T_A max.
Level 2 "D" = 168 Hour Burn-in at 125°C
Level 3 "DS" = Combination of Level 1 and 2.

Speed	Device	Temperature Range
1.0 MHz	MC6859L	0 to 70°C