# T7000A Digital Encryption Processor

## Features

- Programmable DES ciphering modes
  - Electronic codebook (ECB)
  - Cipher block chaining (CBC)
  - 1-, 8-, or 64-bit cipher feedback (CFB)
  - Output feedback (OFB)
- Ciphering rates of 235,000 operations/s for any of the DES modes.
- Data throughput of 1.882 Mbytes/s using 64-bit DES output block
- On-chip RAM and ROM program memory
- Flags readable on the data bus or independent output pins
- Four sets of key and initial value registers
- Separate plain text and cipher text parallel (8-bit) ports
- Separate plain text and cipher text serial ports
- Separate serial key input port
- ECB program available in ROM

## Description

The T7000A Digital Encryption Processor (DEP) is a programmable integrated circuit that provides a low-cost, high-security, cryptographic system for encrypting and decrypting digital signals. It is manufactured using CMOS technology, requires a single 5 V supply, and is supplied in a 40-pin plastic DIP. It implements four data encryption standard (DES) modes and is capable of performing multiple encryption operations or multiplexed key and initial value ciphering.

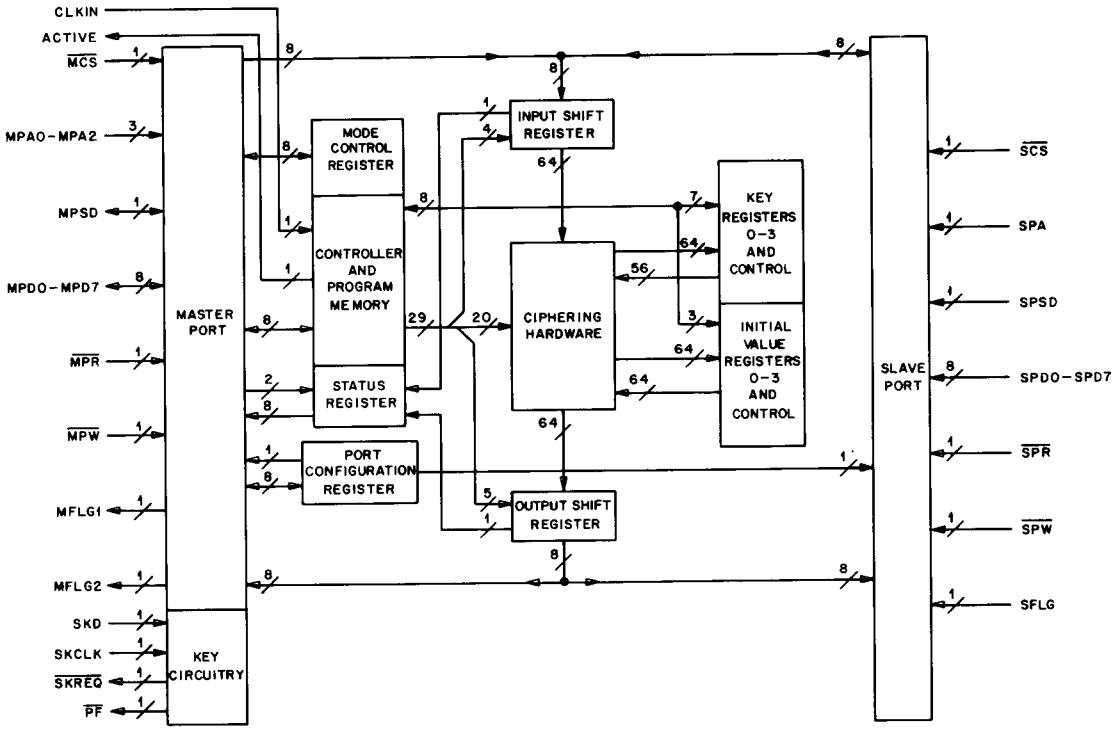
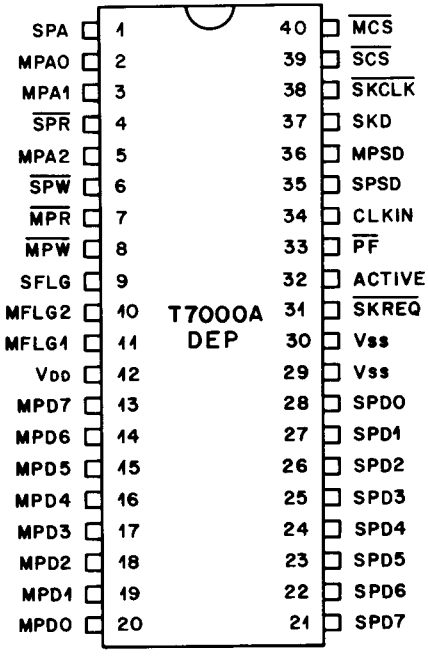



**Figure 1. Block Diagram**

# User Information

## Pin Descriptions




| Symbol | Pin | Symbol | Pin |
|---|---|---|---|
| ACTIVE | 32 | SCS | 39 |
| CLKIN | 34 | SFLG | 9 |
| MCS | 40 | SKCLK | 38 |
| MFLG1 | 11 | SKD | 37 |
| MFLG2 | 10 | SKREQ | 31 |
| MPA0 | 2 | SPA | 1 |
| MPA1 | 3 | SPD0 | 28 |
| MPA2 | 5 | SPD1 | 27 |
| MPD0 | 20 | SPD2 | 26 |
| MPD1 | 19 | SPD3 | 25 |
| MPD2 | 18 | SPD4 | 24 |
| MPD3 | 17 | SPD5 | 23 |
| MPD4 | 16 | SPD6 | 22 |
| MPD5 | 15 | SPD7 | 21 |
| MPD6 | 14 | SPR | 4 |
| MPD7 | 13 | SPSD | 35 |
| MPR | 7 | SPW | 6 |
| MPSD | 36 | VDD | 12 |
| MPW | 8 | VSS | 29 |
| PF | 33 | VSS | 30 |

**Figure 2. Pin Function Diagram and Alphabetical Listing of Symbols**

**Table 1. Pin Descriptions**

| Pin | Symbol | Type | Name/Function |
|---|---|---|---|
| 1 | SPA | I | **Slave Port Address.** When high (1), the contents of the status register can be read, but not written, to the slave port data bus. When low (0), either the input shift register (ISR) or output shift register (OSR) is accessed, depending on the port configuration programmed. |
| 2<br>3 | MPA0<br>MPA1 | I<br>I | **Master Port Address Bits 0 and 1.** Used with MPA2 (pin 5) for internal register selection. |
| 4 | SPR | I | **Slave Port Read (Active Low).** Used with SPA (pin 1) to read from the output shift register (if the slave port is programmed as an output) or from the status register. Data is available on the slave port data bus following the falling edge of the read pulse and remains on the bus as long as SPR is low. SPW (pin 6) should be held high during the read pulse. |

**Table 1. Pin Descriptions (Continued)**

| Pin | Symbol | Type | Name/Function |
|---|---|---|---|
| 5 | MPA2 | I | **Master Port Address Bit 2.** Used with MPA0 and MPA1 (pins 2 and 3) for internal register selection. |
| 6 | $\overline{SPW}$ | I | **Slave Port Write (Active Low).** Used with SPA (pin 1) to write to the input shift register if the slave port has been programmed as an input. The data input is latched on the rising edge of the write pulse. $\overline{SPR}$ (pin 4) should be held high during the write pulse. |
| 7 | $\overline{MPR}$ | I | **Master Port Read (Active Low).** Used with the master port address bus to read one of the internal registers. Data is available on the master port data bus following the falling edge of the read pulse and remains on the bus as long as $\overline{MPR}$ is low. $\overline{MPW}$ (pin 8) should be held high during the read pulse. |
| 8 | $\overline{MPW}$ | I | **Master Port Write (Active Low).** This lead is used with the master port address bus to write to one of the internal registers. The data input is latched into the addressed register on the rising edge of the write pulse. The $\overline{MPR}$ lead should be held high during the write pulse. |
| 9 | SFLG | O | **Slave Flag.** This output indicates the status of either the input or output shift register, depending on the port configuration programmed (see Table 5). If the slave port is programmed as an input, the slave flag reflects the contents of the ISRFULL flag (status register, bit 4). If the slave port is programmed as an output, the slave flag reflects the contents of the OSREMPTY flag (status register, bit 5). Both of these conditions can be read from the status register. |
| 10 | MFLG2 | O | **Master Flag 2.** This output indicates the status of the ISRFULL flag (status register, bit 4). This condition may also be read from the status register (see Table 5). |
| 11 | MFLG1 | O | **Master Flag 1.** This output indicates the status of either the input or output shift register, depending on the port configuration programmed (see Table 5). If the master port is programmed as an input, this lead reflects the contents of the ISRFULL flag (status register, bit 4). If the master port is programmed as an output, this pin indicates the status of the OSREMPTY flag (status register, bit 5). If the master port is programmed as both input and output, this pin indicates the status of the OSRFULL flag and MFLG2 (pin 10) indicates the status of the ISRFULL flag. The status of the input and output shift register can also be read from the status register. |
| 12 | $V_{DD}$ | — | **5 V Supply.** |
| 13 | MPD7 | I/O | **Master Port Data Bit 7.** Bidirectional, 18-bit master port I/O bus. |
| 14 | MPD6 | | **Master Port Data Bit 6.** |
| 15 | MPD5 | | **Master Port Data Bit 5.** |
| 16 | MPD4 | | **Master Port Data Bit 4.** |
| 17 | MPD3 | | **Master Port Data Bit 3.** |
| 18 | MPD2 | | **Master Port Data Bit 2.** |
| 19 | MPD1 | | **Master Port Data Bit 1.** |
| 20 | MPD0 | | **Master Port Data Bit 0.** |

**Table 1. Pin Descriptions (Continued)**

| Pin | Symbol | Type | Name/Function |
|---|---|---|---|
| 21 | SPD7 | I/O | **Slave Port Data Bit 7.** Bidirectional, 8-bit slave port I/O bus. |
| 22 | SPD6 | | **Slave Port Data Bit 6.** |
| 23 | SPD5 | | **Slave Port Data Bit 5.** |
| 24 | SPD4 | | **Slave Port Data Bit 4.** |
| 25 | SPD3 | | **Slave Port Data Bit 3.** |
| 26 | SPD2 | | **Slave Port Data Bit 2.** |
| 27 | SPD1 | | **Slave Port Data Bit 1.** |
| 28 | SPD0 | | **Slave Port Data Bit 0.** |
| 29 | VSS | — | **Ground.** |
| 30 | VSS | — | **Ground.** |
| 31 | $\overline{\text{SKREQ}}$ | O | **Serial Key Request (Active Low).** This output indicates that the DEP is expecting a key input. Active when IO SERIAL ACT is programmed. The condition of this flag can be read from the status register. |
| 32 | ACTIVE | O | **Active.** This output flag is set by the microcode instruction IO ACT. |
| 33 | $\overline{\text{PF}}$ | O | **Parity Fail (Active Low).** When this output is low it indicates that one or more key input bytes had even parity. This flag is set on the 8th $\overline{\text{MPW}}$ pulse (pin 8) when the key is loaded through the parallel master port and on the 64th SKCLK pulse (pin 38) when the key is loaded serially. The status of this flag can be read from the status register. |
| 34 | CLKIN | I | **Clock Input.** The clock signal input at this lead determines all internal timing. A microcode instruction is executed every two clock cycles. The master and slave ports' read and write signals do not have to be synchronous with this clock signal. The frequency range of this clock is 10 kHz to 8 MHz. |
| 35 | SPSD | I/O | **Slave Port Serial Data.** Used to write data to the input shift register or read data from the output shift register, depending on the programmed port configuration. The first bit read or written is the most significant. When this port is selected by the port configuration register, slave port signals $\overline{\text{SPW}}$, $\overline{\text{SPR}}$, and SFLG (pins 6, 4, and 9) are used for control. This port cannot be used to read or write to any of the other six registers. |
| 36 | MPSD | I/O | **Master Port Serial Data.** Used to write data to the input shift register or read data from the output shift register, depending on the programmed port configuration. The first bit read or written is the most significant. When this port is selected by the port configuration register and master port address 0 is addressed, master port signals $\overline{\text{MPW}}$, $\overline{\text{MPR}}$, MFLG1, and MFLG2 (pins 8, 7, 11, and 10) are used for control. |

**Table 1. Pin Descriptions (Continued)**

| Pin | Symbol | Type | Name/Function |
|---|---|---|---|
| 37 | SKD | I | **Serial Key Data.** This input port is used to load key variables serially. The data on this pin is latched into key memory on the falling edge of the serial key clock during the execution of a serial load key program. The key is entered with the most significant bit first and every 8th bit is treated as an odd parity bit. A parity failure does not prevent the 56-bit key from being loaded. |
| 38 | SKCLK | I | **Serial Key Clock.** This clock is used to latch key data into key memory on the falling edge of the clock. The key input circuitry is inhibited after the 64th clock is received. |
| 39 | $\overline{SCS}$ | I | **Slave Chip Select (Active Low).** This input enables the slave port inputs and outputs. When high, all slave port outputs are placed in a high-impedance state. The $\overline{SPW}$, $\overline{SPR}$, SPSD, and SPD0—SPD7 signals are affected. |
| 40 | $\overline{MCS}$ | I | **Master Chip Select (Active Low).** This input enables the master port input and output leads. When high, all master port outputs are placed in a high-impedance state and all inputs are disabled. The $\overline{MPW}$, $\overline{MPR}$, MPSD, and MPD0—MPD7 signals are affected. |

## Overview

Figure 1 is a block diagram of the DEP device. There are three major sections: the ciphering hardware and peripheral circuitry, the controller and program memory, and the ports.

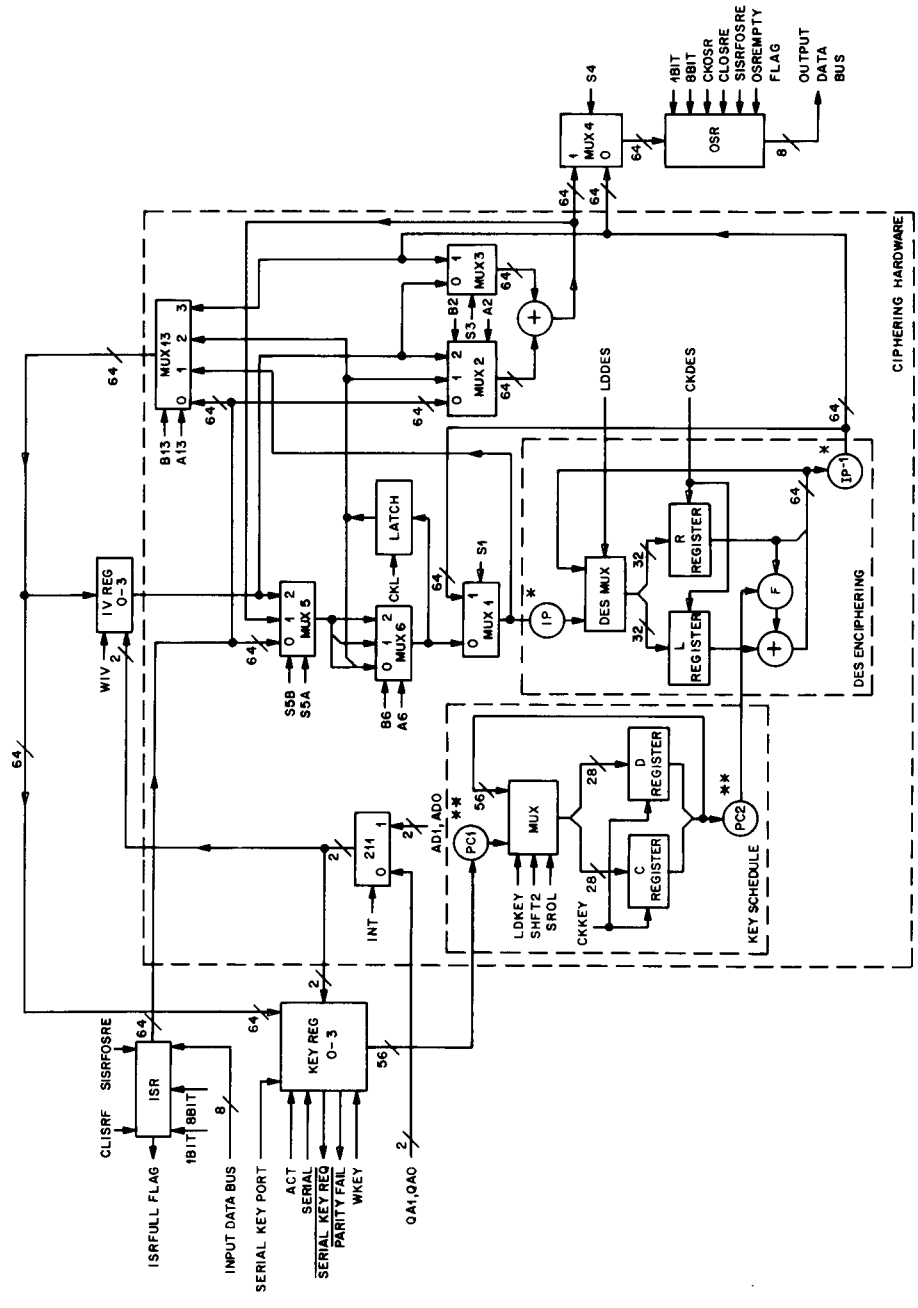
The ciphering hardware contains a high-speed hardware implementation of the National Bureau of Standards Data Encryption Algorithm (DEA) and the necessary hardware to configure the DES operating modes (see Figure 3). Both the key schedule and DES enciphering circuitry are part of the DEA algorithm. The remaining circuitry (seven multiplexers, an exclusive-OR gate, and a latch) is used for the DES operating modes. An input shift register, four key registers, four initial value registers, and an output shift register support the ciphering hardware.

An internal hardware controller executes a 22-bit machine instruction every two clock cycles, thereby setting up the ciphering multiplexers and clocking the appropriate registers. Within the controller, a program counter is used to address the machine instruction stored in either RAM or ROM program memory. On-chip ROM (29 X 22 bits) contains a subroutine controlling the DES hardware, a load initial value program, a load key program, a serial load key program, and an ECB encrypt and decrypt program. These short programs are located at hexadecimal address 00 through 1c (see Figure 5). User-accessible on-chip RAM (32 X 22 bits) allows the user to tailor the ciphering operation to meet system requirements and thus eliminates the need for external hardware. These ciphering programs must start at hex address 20 and cannot exceed hex address 3F.

Master and slave ports are provided so that the plain text and cipher text can be on separate buses. These ports have both serial and 8-bit parallel bidirectional data buses. When using the 8-bit parallel data bus, master or slave, the most significant data or key byte should be written/read first. In the serial mode, the most significant bit is written/read first.

## Registers

Eight addressable internal registers control device operation. Table 2 shows the register assignments for both the master and slave ports during either a read or write operation.




* Initial Permutation.
** Permuted Choice.

**Figure 3. Ciphering Hardware Block Diagram**

**Table 2. Register Assignments**

| Master Port (MP) | | |
|---|---|---|
| **Address** | **Register** | **Size (Bytes)** |
| 0 (write) | Input shift | 8 |
| 0 (read) | Output shift | 8 |
| 1 (read/write) | Status | 1 |
| 2 (read/write) | Port configuration | 1 |
| 3 (read/write) | Mode control | 1 |
| 4 (read/write) | M1 | 1 |
| 5 (read/write) | M2 | 1 |
| 6 (read/write) | M3 | 1 |
| **Slave Port (SP)** | | |
| **Address** | **Register** | **Size (Bytes)** |
| 0 (write) | Input shift | 8 |
| 0 (read) | Output shift | 8 |
| 1 (read) | Status | 1 |

**Input and Output Shift Registers.** Both registers (master or slave port address 0) can be accessed from MPD, MPSD, SPD, or SPSD. The input shift register is a 64-bit, write-only, shift register. The output shift register is a 64-bit, read-only, shift register. The port configuration register controls which port, master or slave, is associated with the input or output shift register. These shift registers are used to input and output data and normally are not accessed until the other registers are loaded.

If a parallel port is used, 8 bytes are read or written to empty or load these registers, except when the 1- or 8-bit cipher feedback (CFB) mode has been programmed. In these cases, a single byte is expected. For 1-bit CFB, only the most significant bit of the byte is used.

If a serial port is used, 64 bits are read or written to empty or load these registers, except when the 1- or 8-bit CFB mode has been programmed. One bit is expected for 1-bit CFB and eight bits for 8-bit CFB.

**Status Register.** This register (master or slave port address 1) can be read or written from the master port data bus but only read from the slave port data bus (see Figure 4).

Bits 1 and 0 (QA1, QA0) are read/write address lines that are used to select key and initial value register pairs 0—3 when the microcode instruction bit, INT, is not set. Key and initial value registers are matched sets, e.g., 00 selects key register 0 and initial value register 0 (see Table 3). The values are loaded into these registers by executing the appropriate program in ROM.

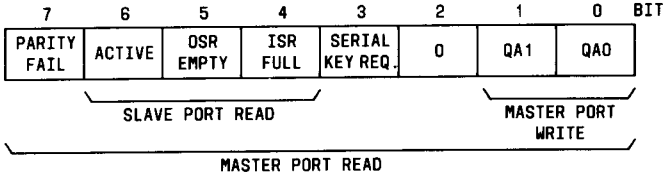



**Figure 4. Status Register**

**Table 3. Key and Initial Value Register Addresses**

| Bits | | Key and Initial Value Register Number |
|---|---|---|
| QA1 | QA0 | |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 2 |
| 1 | 1 | 3 |

Bit 2 of this register is not used.

Bit 3 is a read-only, active-high, serial key request (SKREQ) flag. The complement of this flag ($\overline{\text{SKREQ}}$) is available at output pin 31. SKREQ is microcode-controlled and goes active when the SERIAL and ACT instructions are executed simultaneously.

Bit 4 is a read-only, active-high, input shift register full (ISRFULL) flag. This flag appears on an output pin, the specific pin (MFLG1 , MFLG2 , or SFLG) being determined by the port configuration. An active signal indicates that the ISR is full and additional information written to that register will be ignored. The ISRFULL flag is set automatically whenever the mode control register is written or after the microcode instruction SISRFOSRE is executed. It is cleared by microcode instruction CLISRF.

Bit 5 is a read-only, active-high, output shift register empty (OSREMPTY) flag. This flag appears on an output pin, the specific pin (MFLG1 or SFLG) being determined by the port configuration. An active signal indicates that the OSR is empty and additional attempts to read that register will be ignored. OSREMPTY is set automatically whenever the mode control register is written or after the microcode instruction SISRFOSRE is executed. It is cleared by the microcode instruction CLOSRE.

Bit 6 is a read-only, active-high, activity (ACTIVE) flag. This flag appears on output pin 32 (ACTIVE). It is set by the microcode instruction IO ACT and indicates processor activity. The ACTIVE flag has no effect on device operation.

Bit 7 is a read-only, active-high, parity fail flag. The complement of this flag is available at output pin 33 ($\overline{\text{PF}}$). This flag is latched whenever the WKEY instruction is executed. An active condition indicates that one or more of the key bytes entered had even parity. Device operation is not inhibited by the parity fail flag.

**Port Configuration Register.** This register (master port address 2) is a read/write register accessible only through the master port data bus. Table 4 defines the possible port configurations and associated hex code for data encryption and decryption.

**Table 4. Port Configuration (MP Address = 2)**

| Port Type | Input | Output | Hex Code* | |
|---|---|---|---|---|
| | | | Encrypt | Decrypt |
| Parallel | MPD | SPD | 04 | 84 |
| Parallel | SPD | MPD | 11 | 91 |
| Parallel | MPD | MPD | 01 | 81 |
| Serial | MPSD | SPSD | 28 | A8 |
| Serial | SPSD | MPSD | 62 | E2 |
| Parallel to serial | MPD | SPSD | 08 | 88 |
| Serial to parallel | SPSD | MPD | 61 | E1 |

* The most significant bit in the hex code for the port configuration is an input flag. It is tested by the microcode mnemonic LT?. In the microcode for the standard modes given in this document, this bit is tested to determine the order in which the DES key schedule should be used (encrypt or decrypt).

The conditions indicated by the master and slave port flags are determined by the port configuration (see Table 5).

Bit 7 of the port configuration register is an input flag that is tested by microcode instruction LT?. This bit can be used to indicate the order in which the key schedule is used (encrypt or decrypt) or as a general-purpose conditional jump.

**Table 5. Master and Slave Port Flag Conditions**

| Port Configuration | | Flag Condition | | |
|---|---|---|---|---|
| Input | Output | MFLG1 | MFLG2 | SFLG |
| MPD or MPSD | SPD or SPSD | ISRFULL | — | OSREMPTY |
| SPD or SPSD | MPD or MPSD | OSREMPTY | — | ISRFULL |
| MPD | MPD | OSREMPTY | ISRFULL | — |

**Mode Control and M1, M2, and M3 Registers.** The mode control register (master port address 3) is a read/write register accessible only through the master port data bus. This register is used to address on-chip memory for read/write operations and to begin program execution. Only the six least significant bits are used in this register.

To run a microcode program, write the starting address for the set of instructions to be executed into the mode control register. On the next instruction cycle, load this address into a program counter to begin execution.

To read/write the program memory, load the address of the instruction into the mode control register and read/write one of the three hex bytes (M1, M2, or M3) that make up an instruction on a subsequent $\overline{MPR}/\overline{MPW}$ pulse. Use the master port address bus to select M1, M2, or M3.

The M1, M2, and M3 registers (master port addresses 4—6, respectively) are accessible only through the master port data bus. These three bytes define a 22-bit microcode instruction stored in on-chip program memory. The two most significant bits of register M3 are not used.

## Operation

It is important to use the following operating sequence with the DEP. Deviations from this sequence (e.g., loading the key before loading the ciphering program) may cause unpredictable results:

1. Load the ciphering program.
2. Configure the ports.
3. Load key and initial value register data.
4. Execute the program.

**Loading the Ciphering Program.** Microcode instructions can be entered for any of the DES mode programs (Figures 6—8), multiple programs, multiplexed programs, or the user's own unique cipher program. Thirty-two 22-bit instructions, starting at hex address 20, can be entered. Microcode instructions are loaded into RAM, a byte at a time, through the master port data bus to the address designated by the mode control register. The microcode address is written to the mode control register (MP address 3) and then followed by the three hex bytes (M1, M2, and M3). These three bytes (MP addresses 4—6, respectively) constitute a 22-bit instruction.

**Configuring the Ports.** Data flow, port selection, and the DES key schedule selection (encrypt and decrypt) are programmed by writing the appropriate hex code to the port configuration register (MP address 2). Table 4 shows the various port configuration options.

**Loading Key and Initial Value Register Data.** There are four key and initial value registers that must be externally loaded. A key/initial value register address is written to the status register (see Tables 2 and 3 and Figure 4) and the load initial value program or one of the two load key programs is executed. The following is a description of the load key and load initial value programs. The assembly language listings for these programs are shown in Figure 5.

After the starting address of the load initial value program (hex address 06) is written to the mode control register, the ISRFULL flag becomes inactive and the ACTIVE flag goes active. The eight initial value bytes can then be written to the input shift register through the master port data bus. After the eighth byte is written, the ISRFULL flag goes active and the content of the input shift register is copied to the addressed initial value register. The next internal machine instruction clears the ACTIVE flag.

After the starting address of the parallel load key program (hex address 0B) is written to the mode control register, the ISRFULL flag becomes inactive and the ACTIVE flag goes active. The eight key bytes can then be written to the input shift register through the master port data bus. After the eighth byte is written, the ISRFULL flag goes active and the content of the input shift register is copied to the addressed key register. Coincident with the program's WKEY instruction, the PARITY FAIL flag is set active high if any of the key bytes entered had even parity. The next internal machine instruction clears the ACTIVE flag.

After the starting address of the serial load key program (hex address 10) is written to the mode control register, the ACTIVE and serial key request (SKREQ) flags become active. The 64-bit key must then be clocked into the input shift register through the serial key port. After the last bit is entered, the content of the input shift register is copied into the addressed key register. One internal machine instruction cycle after the key is entered, the ACTIVE and SKREQ flags become inactive. Coincident with the program's WKEY instruction, the PARITY FAIL flag is set active high if any of the key bytes entered had even parity.

**Executing the Program.** After the microcode program is loaded, the desired port configuration is set up, and the key and initial value registers are loaded, the device is ready to begin a ciphering operation. The starting address of the microcode program is written to the mode control register. On the next internal machine cycle, this address is loaded into a program counter and execution begins. To execute the ECB mode, no microcode has to be loaded since it already exists in ROM. For this DES mode, step 1 should be omitted.

Input and output to the DEP device does not have to be synchronous with the input clock. The ISRFULL and OSREMPTY flags signal the host processor to write and read data. When these flags are

inactive, data can be loaded into the input shift register and read from the output shift register by the port associated with these registers. These flags, tested in program memory by conditional machine instructions, determine when to start or stop ciphering data. A typical ciphering program contains the following steps:

1. Multiplexer set-up
2. Wait for input data
3. DES subroutine call
4. Wait until previous output data has been read
5. Latch output data and return to step 2.

## DES Mode Descriptions

The DEP is capable of performing all four DES operating modes: electronic codebook; cipher block chaining; 1-, 8-, or 64-bit cipher feedback; and output feedback. Code for the ECB mode is stored in ROM, beginning at location hexadecimal 12. The DEP can be programmed for the other modes via the RAM. Each mode can be used independently, combined with another mode, or used with multiple keys. For a detailed description of the DES modes refer to *Federal Information Processing Standards Publication 81.*

**Electronic Codebook (ECB) Mode.** This mode is used primarily to encrypt or decrypt keys or initial values through the use of a master key. It is a direct implementation of the DES algorithm. A 64-bit input data block results in a 64-bit output block. Consecutive data blocks are cryptographically independent. Figure 5b contains the assembly language listing for the ECB mode, beginning at hexadecimal address 12.

**Cipher Block Chaining (CBC) Mode.** This mode uses the DES algorithm in a 64-bit feedback mode, which results in consecutive output data blocks being cryptographically dependent. This dependence provides an error-extension characteristic useful in protecting against an active system attack. Figure 6 contains the assembly language listing for the CBC mode.

**Cipher Feedback (CFB) Mode.** This mode is an additive stream cipher in which the DES algorithm is used to generate pseudorandom blocks. This mode provides cryptographic dependence of data blocks and error-extension. It is not necessary for the input block to be 64 bits; it may be 1, 8, or 64 bits. If the 1- or 8-bit mode is selected, a DES operation must be performed for every input bit or byte; consequently, the data rate is reduced by a factor of 64 or 8, respectively. Figure 7 contains the assembly language listings for 1-, 8-, and 64-bit CFB modes.

**Output Feedback (OFB) Mode.** This mode uses the DES algorithm as a pseudorandom number generator. Encryption and decryption are identical operations, and the security of the algorithm is dependent on the proper management of the initial value blocks. This mode has no error-extension property: a 1-bit transmission error results in a 1-bit decryption error. This is an important property when transmitting over a noisy channel. Figure 8 contains the assembly language listing for the OFB mode.

These standard DES modes, after set-up, can be executed in a minimum of 17 instructions. With an 8-MHz input clock, the instruction period is 250 ns, yielding a maximum of 235,000 ciphering operations/s. If the entire output block (all 64 bits) is used, the data throughput rate is 1.882 Mbytes/s.

Multiple encryption can be easily implemented with the DEP device. By using different keys, any of the previously mentioned DES modes can be cascaded to provide multiple encryption.

Figure 9 contains the assembly language listing for the ECB mode using 3 keys for encryption and decryption. Decryption is similar to encryption except that the key schedules are used in reverse order, e.g., the last key register used for encrypting is used first for decrypting.

In addition to using the four DES operating modes, the multiple modes, and the multiplexed modes, the user can choose to program a unique encryption method.

Figure 5b contains an assembly language listing (in ROM) for the ECB DES mode. Figures 6—9 contain assembly language listings for three DES modes and multiple-key ECB. Each listing in Figures 6—9 begins at RAM hexadecimal address 20. When combining programs, program labels may have to be changed to prevent incorrect addressing. Duplicate code in some programs can be combined.

| ADDRESS | 22-Bit Instruction M1 | M2 | M3 | Program Mnemonics |
|---|---|---|---|---|
| **DES Subroutine** | | | | |
| 0 | c2 | 1f | 0 | :00 LDDES CKDES CKKEY |
| 1 | 42 | 10 | 5 | :01 CKDES CKKEY LLC 5 |
| 2 | 52 | 11 | 2 | :02 CKDES SHFT2 CKKEY ILC 02 |
| 3 | 42 | 10 | 5 | CKDES CKKEY LLC 5 |
| 4 | 52 | 11 | 4 | :03 CKDES SHFT2 CKKEY ILC 03 |
| 5 | 42 | 13 | 0 | CKDES CKKEY RET 0 |
| **Load Initial Value** | | | | |
| 6 | 1 | b | 3 | B6 IO LDMP ACT<br>DES INPUT = ISR OSR INPUT = DESOUT<br>IV INPUT = ISR LATCH INPUT = ISR |
| 7 | 1 | 1a | 0 | CLISRF ADD |
| 8 | 0 | 15 | 8 | :10 ISRFT? 10 |
| 9 | 0 | 3c | 0 | WIV CLEAR |
| a | 0 | 14 | a | :20 GTO 20 |
| **Parallel Load Key** | | | | |
| b | 1 | b | 3 | B6 IO LDMP ACT<br>DES INPUT = ISR OSR INPUT = DESOUT<br>IV INPUT = ISR LATCH INPUT = ISR |
| c | 1 | 1a | 0 | :25 CLISRF ADD |
| d | 0 | 15 | d | :30 ISRFT? 30 |
| e | 8 | 1c | 0 | WKEY CLEAR |
| f | 0 | 14 | f | :40 GTO 40 |
| **Serial Load Key** | | | | |
| 10 | 1 | b | 7 | B6 IO LDMP SERIAL ACT<br>DES INPUT = ISR OSR INPUT = DESOUT<br>IV INPUT = ISR LATCH INPUT = ISR |
| 11 | 0 | 14 | c | GTO 25 |

**Figure 5a. ROM Programs**

| ADDRESS | M1 | M2 | M3 | Program Mnemonics |
|---|---|---|---|---|
| | | 22-Bit Instruction | | |
| **ECB Encrypt or Decrypt** | | | | |
| 12 | 1 | c | 0 | B6 CLEAR<br>DES INPUT = ISR OSR INPUT = DESOUT<br>IV INPUT = ISR LATCH INPUT = ISR |
| 13 | 7 | 18 | 15 | LDKEY CKKEY CLISRF LT? 100 |
| 14 | 2 | 19 | 1 | CKKEY SROL SHFTR |
| 15 | 0 | 15 | 15 | :100 ISRFT? 100 |
| 16 | c3 | 12 | 1 | CLISRF LDDES CKDES CKKEY SUB 01 |
| 17 | 0 | 17 | 1a | ISRFOSRET? 120 |
| 18 | 0 | 16 | 18 | :110 OSRET? 110 |
| 19 | 0 | d4 | 15 | CLOSRE CKOSR GTO 100 |
| 1a | c3 | d2 | 1 | :120 CLISRF CLOSRE CKOSR LDDES CKDES CKKEY SUB 01 |
| 1b | 0 | 17 | 1a | :130 ISRFOSRET? 120 |
| 1c | 0 | 14 | 18 | GTO 110 |

**Figure 5b. ROM Programs**

| ADDRESS | 22-Bit Instruction M1 | M2 | M3 | Program Mnemonics |
|---|---|---|---|---|
| **CBC Encrypt** | | | | |
| 20 | 3 | c | 0 | S5A B6 CLEAR<br>DES INPUT = ISR^IV OSR INPUT = DESOUT<br>IV INPUT = ISR LATCH INPUT = ISR^IV |
| 21 | 7 | 18 | 23 | LDKEY CKKEY CLISRF LT? 200 |
| 22 | 2 | 19 | 1 | CKKEY SROL SHFTR |
| 23 | 0 | 15 | 23 | :200 ISRFT? 200 |
| 24 | c3 | 12 | 1 | CLISRF LDDES CKDES CKKEY SUB 01 |
| 25 | 13 | 4 | 2c | :210 S3 S5A B6 GTO 130<br>DES INPUT = ISR^DESOUT OSR INPUT = DESOUT<br>IV INPUT = ISR LATCH INPUT = ISR^DESOUT |
| 26 | 0 | 15 | 26 | :100 ISRFT? 100 |
| 27 | c3 | 12 | 1 | CLISRF LDDES CKDES CKKEY SUB 01 |
| 28 | 0 | 17 | 2b | ISRFOSRET? 120 |
| 29 | 0 | 16 | 29 | :110 OSRET? 110 |
| 2a | 0 | d4 | 26 | CLOSRE CKOSR GTO 100 |
| 2b | c3 | d2 | 1 | :120 CLISRF CLOSRE CKOSR LDDES CKDES CKKEY SUB 01 |
| 2c | 0 | 17 | 2b | :130 ISRFOSRET? 120 |
| 2d | 0 | 14 | 29 | GTO 110 |
| **CBC Decrypt** | | | | |
| 2e | 7 | 1c | 0 | LDKEY CKKEY CLISRF CLEAR |
| 2f | 59 | 48 | 31 | B2 S3 S4 B6 B13 LT? 250<br>DES INPUT = ISR OSR INPUT = IV^DESOUT<br>IV INPUT = Qn LATCH INPUT = ISR |
| 30 | 2 | 19 | 1 | CKKEY SROL SHFTR |
| 31 | 0 | 15 | 31 | :250 ISRFT? 250 |
| 32 | e3 | 12 | 1 | CLISRF CKL LDDES CKDES CKKEY SUB 01 |
| 33 | 0 | 17 | 36 | ISRFOSRET? 230 |
| 34 | 0 | 16 | 34 | :220 OSRET? 220 |
| 35 | 0 | f4 | 31 | CLOSRE CKOSR WIV GTO 250 |
| 36 | e3 | f2 | 1 | :230 CLISRF CKL WIV CLOSRE CKOSR LDDES CKDES CKKEY SUB 01 |
| 37 | 0 | 17 | 36 | ISRFOSRET? 230 |
| 38 | 0 | 14 | 34 | GTO 220 |

**Figure 6. Assembly Language Listing for CBC Mode**

| ADDRESS | 22-Bit Instruction M1 | M2 | M3 | Program Mnemonics |
|---|---|---|---|---|
| **64-bit CFB Encrypt** | | | | |
| 20 | 1d | c | 0 | S3 S4 S5B B6 CLEAR<br>DES INPUT = IV OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = IV |
| 21 | 7 | 12 | 0 | LDKEY CKKEY CLISRF SUB 00 |
| 22 | 1b | 4 | 24 | S3 S4 S5A B6 GTO 102<br>DES INPUT = ISR^DESOUT OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = ISR^DESOUT |
| 23 | e3 | d2 | 1 | :101 LDDES CKDES CKL CKKEY CLISRF CLOSRE CKOSR SUB 01 |
| 24 | 0 | 17 | 23 | :102 ISRFOSRET? 101 |
| 25 | 0 | 14 | 24 | GTO 102 |
| **64-bit CFB Decrypt** | | | | |
| 26 | 1d | c | 0 | S3 S4 S5B B6 CLEAR<br>DES INPUT = IV OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = IV |
| 27 | 7 | 12 | 0 | LDKEY CKKEY CLISRF SUB 00 |
| 28 | 19 | 4 | 24 | S3 S4 B6 GTO 102<br>DES INPUT = ISR OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = ISR |
| **8-bit CFB Encrypt** | | | | |
| 29 | 1d | b | 10 | S3 S4 S5B B6 IO 8BIT<br>DES INPUT = IV OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = IV |
| 2a | 27 | 12 | 0 | CKL LDKEY CKKEY CLISRF SUB 00 |
| 2b | 1a | 84 | 24 | S3 S4 S5A A6 GTO 102<br>DES INPUT = Qn<<8 \| \| ISR^DESOUT OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = Qn<<8 \| \| ISR^DESOUT |

**Figure 7a. Assembly Language Listing for 1-, 8-, and 64-Bit CFB Modes**

| ADDRESS | 22-Bit Instruction M1 | M2 | M3 | Program Mnemonics |
|---|---|---|---|---|
| **8-bit CFB Decrypt** | | | | |
| 2c | 1d | b | 10 | S3 S4 S5B B6 IO 8BIT<br>DES INPUT = IV OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = IV |
| 2d | 27 | 12 | 0 | CKL LDKEY CKKEY CLISRF SUB 00 |
| 2e | 18 | 84 | 24 | S3 S4 A6 GTO 102<br>DES INPUT = Qn<<8 \| \| ISR OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = Qn<<8 \| \| ISR |
| **1-bit CFB Encrypt** | | | | |
| 2f | 1d | b | 8 | S3 S4 S5B B6 IO 1BIT<br>DES INPUT = IV OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = IV |
| 30 | 27 | 12 | 0 | CKL LDKEY CKKEY CLISRF SUB 00 |
| 31 | 1a | 4 | 24 | S3 S4 S5A GTO 102<br>DES INPUT = Qn<<1 \| \| ISR^DESOUT OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = Qn<<1 \| \| ISR^DESOUT |
| **1-bit CFB Decrypt** | | | | |
| 32 | 1d | b | 8 | S3 S4 S5B B6 IO 1BIT<br>DES INPUT = IV OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = IV |
| 33 | 27 | 12 | 0 | CKL LDKEY CKKEY CLISRF SUB 00 |
| 34 | 18 | 4 | 24 | S3 S4 GTO 102<br>DES INPUT = Qn<<1 \| \| ISR OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = Qn<<1 \| \| ISR |

**Figure 7b. Assembly Language Listing for 1-, 8-, and 64-Bit CFB Modes**

| ADDRESS | 22-Bit Instruction M1 | M2 | M3 | Program Mnemonics |
|---|---|---|---|---|
| **OFB Encrypt and Decrypt** | | | | |
| 20 | 1d | c | 0 | S3 S4 S5B B6 CLEAR<br>DES INPUT = IV OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = IV |
| 21 | 7 | 12 | 0 | LDKEY CKKEY CLISRF SUB 00 |
| 22 | 98 | 4 | 24 | S1 S3 S4 GTO 102<br>DES INPUT = DESOUT OSR INPUT = ISR^DESOUT<br>IV INPUT = ISR LATCH INPUT = Qn<<1 \| \| ISR |
| 23 | e3 | d2 | 1 | :101 LDDES CKDES CKL CKKEY CLISRF CLOSRE CKOSR SUB 01 |
| 24 | 0 | 17 | 23 | :102 ISRFOSRET? 101 |
| 25 | 0 | 14 | 24 | GTO 102 |

**Figure 8. Assembly Language Listing for OFB Mode**

| ADDRESS | 22-Bit Instruction M1 | M2 | M3 | Program Mnemonics |
|---|---|---|---|---|
| **Subroutine for ECB with 3 Keys** | | | | |
| 20 | 0 | 19 | 0 | :20 SROL SHFTL |
| 21 | c6 | 18 | 24 | LDDES CKDES LDKEY CKKEY LT? 25 |
| 22 | 2 | 1f | 0 | CKKEY |
| 23 | 0 | 19 | 1 | SROL SHFTR |
| 24 | 2 | 14 | 1 | :25 CKKEY GTO 01 |
| **3 Key ECB Encrypt** | | | | |
| /* | | | | |
| 25 | 1 | 1c | 0 | CLISRF CLEAR |
| 26 | 1 | a | 1 | :100 B6 ADD INT<br>DES INPUT = ISR OSR INPUT = DESOUT<br>IV INPUT = ISR LATCH INPUT = ISR |
| 27 | 0 | 15 | 27 | :110 ISRFT? 110 |
| 28 | 1 | 12 | 20 | CLISRF SUB 20 |
| 29 | 81 | a | 3 | B6 S1 ADD INT ADD0<br>DES INPUT = DESOUT OSR INPUT = DESOUT<br>IV INPUT = ISR LATCH INPUT = ISR |
| 2a | 0 | 12 | 20 | SUB 20 |
| 2b | 0 | 1a | 5 | ADD INT ADD1 |
| 2c | 0 | 12 | 20 | SUB 20 |
| 2d | 0 | 16 | 2d | :140 OSRET? 140 |
| 2e | 0 | d4 | 26 | CLOSRE CKOSR GTO 100 |

**Figure 9a. Assembly Language Listing for the ECB Mode Using 3 Keys**

**3 Key ECB Decrypt**

```
2f   1  1c   0   CLISRF CLEAR
30   1   a   5   :200 B6 ADD INT ADD1
                   DES INPUT = ISR     OSR INPUT = DESOUT
                   IV INPUT = ISR     LATCH INPUT = ISR
31   0  15  31   :210 ISRFT? 210
32   1  12  20   CLISRF SUB 20
33  81   a   3   B6 S1 ADD INT ADD0
                   DES INPUT = DESOUT     OSR INPUT = DESOUT
                   IV INPUT = ISR     LATCH INPUT = ISR
34   0  12  20   SUB 20
35   0  1a   1   ADD INT
36   0  12  20   SUB 20
37   0  16  37   :240 OSRET? 240
38   0  d4  30   CLOSRE CKOSR GTO 200
```

**Figure 9b. Assembly Language Listing for the ECB Mode Using 3 Keys**

## Instruction Set

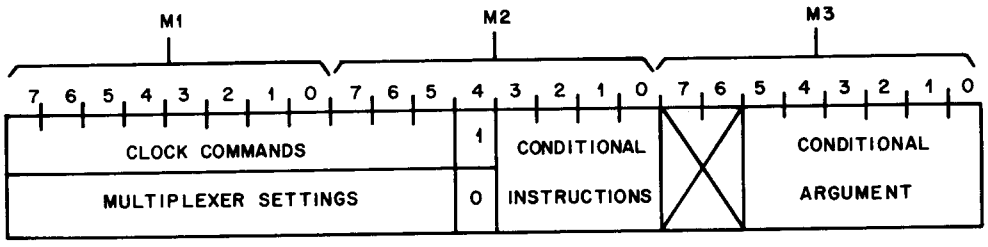



**Figure 10. 22-Bit Instruction Diagram**

Bytes M1, M2, and M3 constitute a 22-bit instruction. Bit 4 of byte M2 determines which set of instructions are used in bits 0—7 of byte M1 and bits 5—7 of byte M2. If bit 4 of byte M2 is high, the clock command instructions are used. If this bit is low, the multiplexer setting instructions are used.

Bits 0—3 of byte M2 are decoded to one of thirteen conditional instructions. With the exception of RET and CLEAR, these instructions use the third byte, M3, as an argument. A description of each instruction is given in Table 6.

The timing diagram for the instruction set is shown in Figure 14. An instruction is executed every two clock cycles. The ciphering rate can be computed by multiplying the number of instructions in the ciphering operation by twice the CLKIN period.

**Table 6. Instruction Set — Clock Commands and Multiplexer Settings**

| Clock Commands (M2, Bit 4 = 1) | | | |
|---|---|---|---|
| **Byte** | **Bit** | **Mnemonic** | **Description** |
| M1 | 7 | LDDES | Enables the DES multiplexer to receive the output from MUX 1 when high or from the DES itself when low. |
| M1 | 6 | CKDES | Clocks the DES L and R registers. |
| M1 | 5 | CKL | Clocks the latch register. |
| M1 | 4 | SHFT2 | Enables the key circuitry to rotate 2 positions when high and 1 position when low. |
| M1 | 3 | WKEY | Latches the key register currently addressed. |
| M1 | 2 | LDKEY | Enables the key schedule C and D registers to be loaded from the addressed key register when high. When low, the contents of the C and D registers can be rotated 1 or 2 positions, left or right, depending on the state of the instructions SHFT2, SROL, and CKKEY. These two registers are used in the key schedule generation for the DES algorithm. |
| M1 | 1 | CKKEY | Clocks the key schedule C and D registers. |
| M1 | 0 | CLISRF | Clears the ISRFULL flag and allows data to be written into the ISR. |
| M2 | 7 | CLOSRE | Clears the OSREMPTY flag and allows data to be read from the OSR. |
| M2 | 6 | CKOSR | Clocks the output from MUX 4 into the OSR. |
| M2 | 5 | WIV | Writes the output of MUX 13 into the initial value memory. |
| **Multiplexer Settings (M2, Bit 4 = 0)** | | | |
| M1 | 7 | S1 | Selects the input line for MUX 1. A low selects input line 0; a high selects input line 1. |
| M1<br>M1 | 6<br>5 | B2<br>A2 | Selects the input line for MUX 2:<br>**B2 A2 Input Line**<br>0 0 0<br>0 1 1<br>1 0 2<br>1 1 Illegal<br>An error occurs if both B2 and A2 are high. |
| M1 | 4 | S3 | Selects the input line for MUX 3. A low selects input line 0; a high selects input line 1. |
| M1 | 3 | S4 | Selects the input line for MUX 4. A low selects input line 0; a high selects input line 1. |
| M1<br>M1 | 2<br>1 | S5B<br>S5A | Selects the input line for MUX 5:<br>**S5B S5A Input Line**<br>0 0 0<br>0 1 1<br>1 0 2<br>1 1 Illegal<br>An error occurs if both S5B and S5A are high. |

**Table 6. Instruction Set — Clock Commands and Multiplexer Settings (Continued)**

| Multiplexer Settings (M2, Bit 4 = 0) (Continued) | | | |
|---|---|---|---|
| **Byte** | **Bit** | **Mnemonic** | **Description** |
| M1<br>M2 | 0<br>7 | B6<br>A6 | Selects the input line for MUX 6:<br>**B6 A6 Input Line**<br>0 0 0<br>0 1 1<br>1 0 2<br>1 1 Illegal<br>An error occurs if both B6 and A6 are high. |
| M2<br>M2 | 6<br>5 | B13<br>A13 | Selects the input line for MUX 13:<br>**B13 A13 Input Line**<br>0 0 0<br>0 1 1<br>1 0 2<br>1 1 3 |

**Table 7. Instruction Set — Conditional Instructions**

| M2 Bits | | | | | |
|---|---|---|---|---|---|
| **3** | **2** | **1** | **0** | **Mnemonic** | **Description** |
| 0 | 0 | 0 | 0 | LLC | Loads the loop counter with the least significant nibble in M3. There is only one loop counter. |
| 0 | 0 | 0 | 1 | ILC | Decrements the loop counter and jumps to the address in M3 if the loop counter is not 0. |
| 0 | 0 | 1 | 0 | SUB | The current program counter instruction address is incremented and latched before the program jumps to the address specified by M3. Only one level of subroutine call is allowed. |
| 0 | 0 | 1 | 1 | RET | Return from subroutine. The program jumps to the address latched when the preceding SUB command is executed. |
| 0 | 1 | 0 | 0 | GTO | The program jumps to the address in M3. |
| 0 | 1 | 0 | 1 | ISRFT? | If the ISR is not full, the program jumps to the address specified by M3. |
| 0 | 1 | 1 | 0 | OSRET? | If the OSR is not empty, the program jumps to the address specified by M3. |
| 0 | 1 | 1 | 1 | ISRFOSRET? | If the ISR is full and the OSR is empty, the program jumps to the address specified by M3. |
| 1 | 0 | 0 | 0 | LT? | If bit 7 of the port configuration register is low, the program jumps to the instruction address in M3. This bit can be used to select the order in which the key schedule is used (encrypt or decrypt). |

**Table 7. Instruction Set — Conditional Instructions (Continued)**

| M2 Bits | | | | Mnemonic | M3 Bit | Mnemonic | Description |
|---|---|---|---|---|---|---|---|
| 3 | 2 | 1 | 0 | | | | |
| 1 | 1 | 1 | 1 | UI | — | — | Unconditional increment to next instruction. |
| 1 | 1 | 0 | 1 | — | — | — | Not used. |
| 1 | 1 | 1 | 0 | — | — | — | Not used. |
| 1 | 0 | 0 | 1 | SROL | 0 = 1 | SHFTR | Latches a right-key schedule rotation. |
| | | | | | 0 = 0 | SHFTL | Latches a left-key schedule rotation. |
| 1 | 0 | 1 | 0 | ADD | 2<br>1 | ADD1<br>ADD0 | Latches the key/initial value register address:<br>**ADD1 ADD0 Reg Pair**<br>0 0 0<br>0 1 1<br>1 0 2<br>1 1 3 |
| | | | | | 0 | INT | A high specifies the internal key/initial value address bus; a low specifies the key/initial value address specified by bits 0 and 1 of the status register. |
| 1 | 0 | 1 | 1 | IO | 5 | SISRFOSRE | A high sets both the ISRFULL flag and the OSREMPTY flag active. |
| | | | | | 4<br>3 | 8BIT<br>1BIT | Selects 1-, 8-, or 64-bit CFB mode:<br>**1-Bit 8-Bit CFB Mode**<br>0 0 64-bit<br>0 1 8-bit<br>1 0 1-bit<br>1 1 Illegal |
| | | | | | 2 | SERIAL | Sets the key circuitry for a serial key input when high and parallel key input when low. |
| | | | | | 1 | LDMP | A high sets the input circuitry to receive data from the master port regardless of the conditions programmed in the port configuration register. |
| | | | | | 0 | ACT | A high sets both the ACTIVE flag in the status register and output pin 32. |
| 1 | 1 | 0 | 0 | CLEAR | NA | NA | Initializes control logic in the DEP. Specifically, this instruction clears the following bits: ACT, LDMP, SERIAL, 1BIT, 8BIT, INT, ADD0, ADD1, SHFTL, SHFTR. This instruction is typically used in the first line of a program. |

NA = not applicable.

# Characteristics

## Clocks

CLKIN: 10 kHz to 8 MHz
SKCLK: 10 kHz to 1.6 MHz

## On-Chip Memory

ROM: 29 X 22 bits (hex address 00—1C)
RAM: 32 X 22 bits (hex address 20—3F)

| ROM Address Map | |
|---|---|
| **Address** | **Program** |
| 00 | DES hardware subroutine |
| 06 | Load initial value |
| 0B | Parallel load key |
| 10 | Serial load key |
| 12 | ECB encrypt or decrypt |

## Electrical Characteristics

$T_A$ = 0 to 70 °C, $V_{DD}$ = 5 V ± 10%, $V_{SS}$ = 0 V

| Parameter | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|
| Supply current | $I_{DD}$ | — | — | 90 | mA | 0 °C, $V_{DD}$ = 5.5 V |
| Input voltage: | | | | | | |
| low | $V_{IL}$ | — | — | 0.8 | V | — |
| high | $V_{IH}$ | 2.0 | — | — | V | — |
| Output voltage: | | | | | | |
| low | $V_{OL}$ | — | — | 0.4 | V | $I_{OL}$ = 1.6 mA |
| high | $V_{OH}$ | 2.4 | — | — | V | $I_{OH}$ = 400 μA |
| Power dissipation | PD | — | 0.3 | 0.5 | W | 0 °C, $V_{DD}$ = 5.5 V |
| | | — | — | 0.4 | W | 70 °C, $V_{DD}$ = 5.5 V |

### Maximum Ratings

Voltage ($V_{SS}$) range on any pin with respect to ground ..... –0.5 to $V_{DD}$ + 0.5 V
Storage temperature (Tstg) range ..... –65 to +125 °C

Maximum ratings are the limiting conditions that can be applied under all variations of circuit and environmental conditions without the occurrence of permanent damage.

External leads can be bonded or soldered safely at temperatures up to 300 °C.

## Timing Characteristics

| Symbol | Description | Min | Max | Unit |
|---|---|---|---|---|
| tAVRL | Address set-up time (read) | 70 | — | ns |
| tAVWL | Address set-up time (write) | 70 | — | ns |
| tCLKINHCLKINH | CLKIN period | 0.125 | 100 | μs |
| tDVWH | Data valid to write pulse rising edge | 80 | — | ns |
| tPCHPCH | Instruction period | 2tCLKINHCLKINH | — | ns |
| tRHDX | Read pulse to data bus float | — | 80 | ns |
| tRHFLGH | Last read pulse to rising $\overline{\text{MFLG}}$ or $\overline{\text{SFLG}}$ | — | 80 | ns |
| tRHRH | $\overline{\text{MPR}}$ or $\overline{\text{SPR}}$ period | 2tCLKINHCLKINH | — | ns |
| tRLDV | Read pulse to data valid | — | 70 | ns |
| tSKCLKHSKCLKH | SKCLK period | 0.625 | — | μs |
| tSKCLKLSKDX | Serial key data hold time | 70 | — | ns |
| tSKCLKLSKREQH | Last falling serial key clock to rising serial key request | — | 4tCLKINHCLKINH + tWHFLGH | ns |
| tSKDVSKCLKL | Serial key data set-up time | 70 | — | ns |
| tSKREQLSKCLKL | Serial key request to first falling serial key clock | 4tCLKINHCLKINH | — | ns |
| tWHDX | Write pulse data hold | 15 | — | ns |
| tWHFLGH | Last write pulse to rising $\overline{\text{MFLG}}$ or $\overline{\text{SFLG}}$ | — | 60 | ns |
| tWHWH | $\overline{\text{MPW}}$ or $\overline{\text{SPW}}$ period | 2tCLKINHCLKINH | — | ns |

### Timing Diagram Nomenclature

| Term | Definition | Term | Definition | Term | Definition |
|---|---|---|---|---|---|
| ADR | Address | M1D | M1 data | PD | Port data |
| CD | Cipher data | M2D | M2 data | SD | Status data |
| MD | Mode data | M3D | M3 data | UD | Unciphered data (plain text) |

## Timing Diagrams

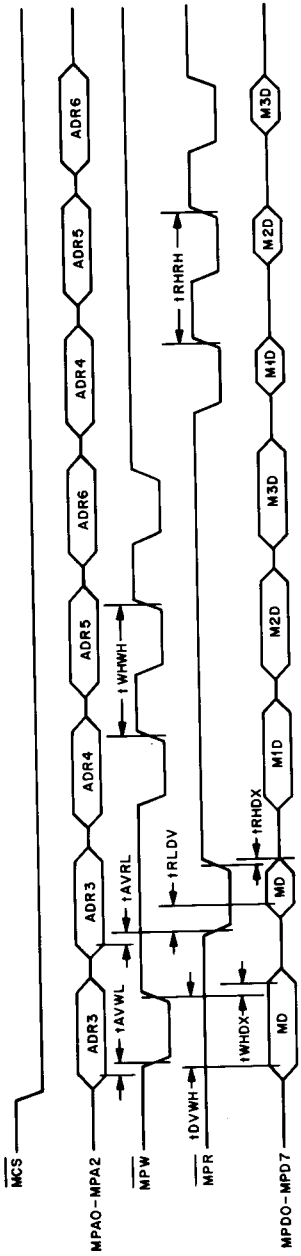



Figure 11. Memory Load Timing

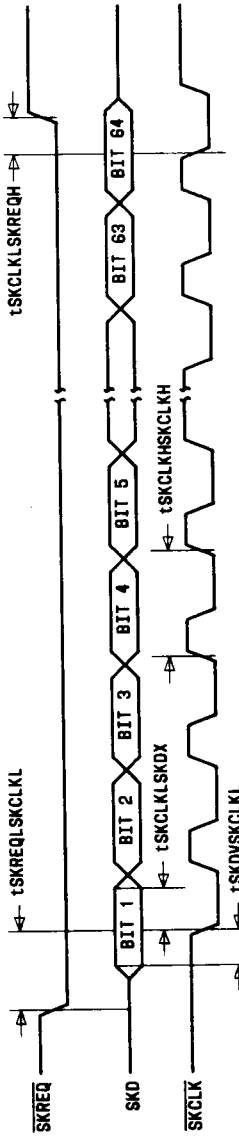
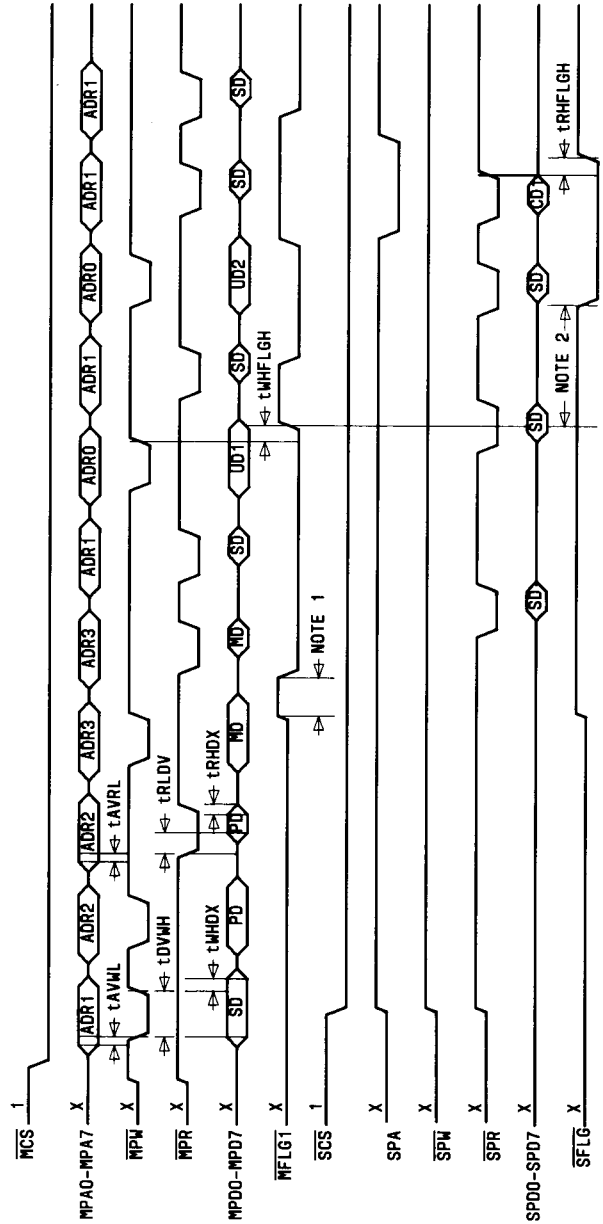
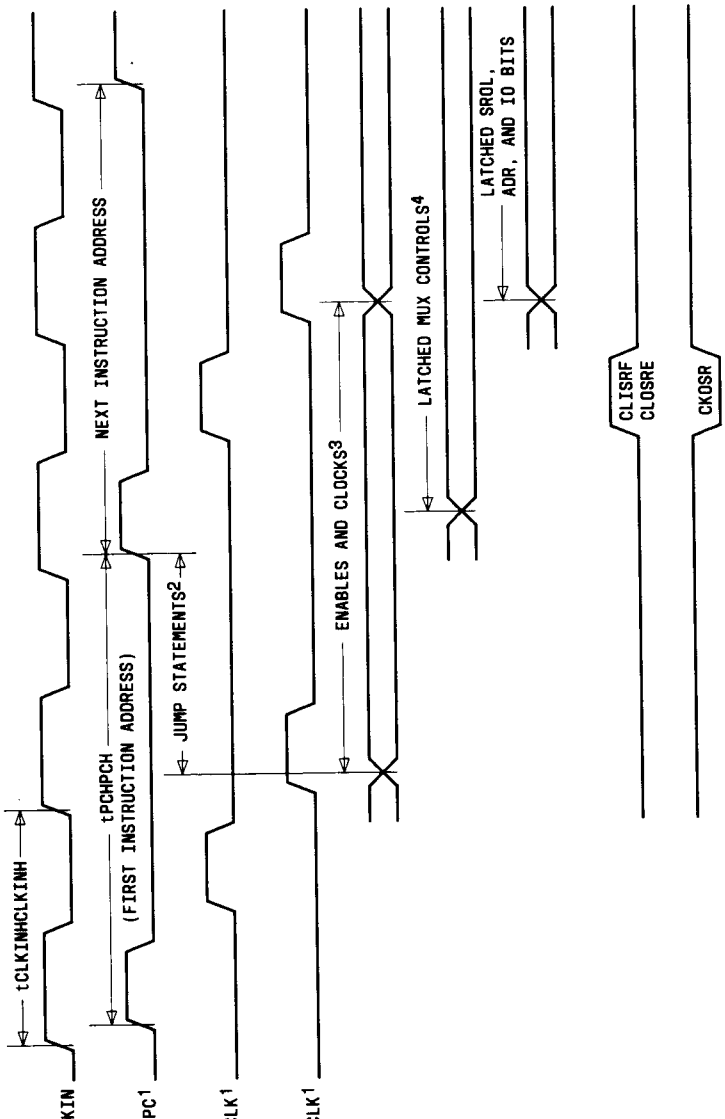



Figure 12. Serial Key Timing




Notes:
1 This time is determined by the number of program instructions executed before instruction CLISRF.
2 This time is determined by the number of program instructions executed between the time MFLG1 goes high and instruction CLOSRE executes (ciphering algorithm time).

**Figure 13. 8-bit CFB Mode Timing with the Master Port Configured for Input and the Slave Port as an Output**




Notes:

1 PC (Program Counter), MCLK, and SCLK are internal nonoverlapping clocks generated from CLKIN.
2 LLC, ILC, SUB, RET, GTO, ISRFT?, OSRET?, ISRFOSRET?, LT?
3 LDDES, CKDES, CKL, SHFT2, WKEY, LDKEY, CKKEY, WIV.
4 S1, A2, B2, S3, S4, S5A, S5B, A6, B6, A13, B13.

**Figure 14. Internal Machine Instruction Timing**