8294
DATA ENCRYPTION UNIT

- **80 byte/sec data conversion rate.**
- **64-bit data encryption using 56-bit key.**
- **DMA interface.**
- **3 interrupt outputs to aid in loading and unloading data.**
- **7-bit user output port.**
- **Single 5V ± 10% power supply.**
- **Peripheral to MCS-85™, MCS-80™ and MCS-48™ processors.**
- **Compatible with algorithm specified in Federal Information Processing Data Encryption Standard.**
- **Encrypt and decrypt modes available.**

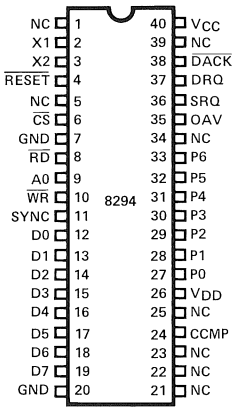
DESCRIPTION

The 8294 Data Encryption Unit (DEU) is a microprocessor peripheral device designed to encrypt and decrypt 64-bit blocks of data using the algorithm specified in the Federal Information Processing Data Encryption Standard. The DEU operates on 64-bit text words using a 56-bit user-specified key to produce 64-bit cipher words. The operation is reversible: if the cipher word is operated upon, the original text word is produced. The algorithm itself is permanently contained in the 8294; however, the 56-bit key is user-defined and may be changed at any time.

The 56-bit key and 64-bit message data are transferred to and from the 8294 in 8-bit bytes by way of the system data bus. A DMA interface and three interrupt outputs are available to minimize software overhead associated with data transfer. Also, by using the DMA interface two or more DEUs may be operated in parallel to achieve effective system conversion rates which are virtually any multiple of 80 bytes/second. The 8294 also has a 7-bit TTL compatible output port for user-specified functions.

Because the 8294 is compatible with the NBS encryption standard it can be used in a variety of Electronic Funds Transfer applications as well as other electronic banking and data handling applications where data must be encrypted.

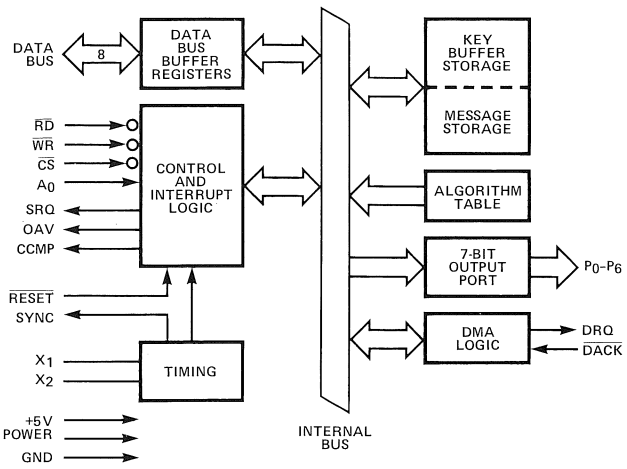
PIN CONFIGURATION



PIN NAMES

PIN NAME	FUNCTION
D_7-D_0	DATA BUS
$\overline{RD}$, $\overline{WR}$	READ, WRITE STROBES
$\overline{CS}$	CHIP SELECT
A_0	CONTROL/DATA SELECT
$\overline{RESET}$	RESET INPUT
X_1, X_2	FREQUENCY REFERENCE INPUT
SYNC	HIGH FREQUENCY OUTPUT
DRQ, $\overline{DACK}$	DMA REQUEST, DMA ACKNOWLEDGE
SRQ, OAV, CCMP	INTERRUPT REQUEST OUTPUTS
P_6-P_0	OUTPUT PORT LINES
V_{CC}, V_{DD}, GND	+5V POWER, GND

BLOCK DIAGRAM



COMMAND SUMMARY

1 — Enter New Key

OP CODE:

0	1	0	0	0	0	0	0

MSB LSB

This command is followed by 8 data inputs which are retained in the key buffer (RAM) to be used in encrypting and decrypting data.

2 — Encrypt Data

OP CODE:

0	0	1	1	0	0	0	0

MSB LSB

This command puts the 8294 into the encrypt mode.

3 — Decrypt Data

OP CODE:

0	0	1	0	0	0	0	0

MSB LSB

This command puts the 8294 into the decrypt mode.

4 — Set Mode

OP CODE:

0	0	0	0	A	B	C	D

MSB LSB

where:

A is the OAV (Output Available) interrupt enable
B is the SRQ (Service Request) interrupt enable
C is the DMA (Direct Memory Access) transfer enable
D is the CCMP (Conversion Complete) interrupt enable

This command determines which interrupt outputs will be enabled. A "1" in bits A, B, or D will enable the OAV, SRQ, or CCMP interrupts respectively. A "1" in bit C will allow DMA transfers. When bit C is set the OAV and SRQ interrupts should also be enabled (bits A,B = 1). Following the command in which bit C, the DMA bit, is set the 8294 will expect one data byte to specify the number of 8-byte blocks to be converted using DMA.

5 — Write to Output Port

OP CODE:

1	P_6	P_5	P_4	P_3	P_2	P_1	P_0

MSB LSB

This command causes the 7 least significant bits of the command byte to be latched as output data on the 8294 output port.

FUNCTIONAL DESCRIPTION

In non-DMA mode, the conversion sequence is as follows:

1. A mode command is issued to enable the desired interrupt outputs.
2. A new key command is issued followed by 8 data inputs to initialize the key. Each byte must have odd parity.
3. The encrypt data or decrypt data command is issued to set the DEU in the desired mode.

After this, data conversions are made by writing 8 data bytes and then reading back 8 converted data bytes. Any of the above commands may be issued between data conversions to change the basic operation of the DEU; e.g., a decrypt data command could be issued to change the DEU from encrypt mode to decrypt mode without changing either the key or the interrupt outputs enabled.

COMMAND AND DATA TRANSFER

Four internal registers are addressable by the master: 2 for input, 2 for output. Access and function of these registers are described below.

$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	A_0	Register
1	0	0	0	Data input buffer
0	1	0	0	Data output buffer
0	1	0	1	Status output buffer
1	0	0	1	Command input buffer
X	X	1	X	Don't care

Data Input Buffer — Data written to this register is interpreted as part of a key, as data to be encrypted/decrypted, or as a DMA block count, depending on the command sequence preceding the write.

Data Output Buffer — Data read from this register will be the output of the encrypter/decrypter function.

Status Output Buffer — DEU status is available in this register at all times.

STATUS BIT:	7	6	5	4	3	2	1	0
FUNCTION:	XXX	XXX	XXX	KPE	HS	DEC	IBF	OBF

OBF — Output buffer full; OBF = 1 indicates that the output buffer contains encrypter/decrypter output data. It is set false when the data is read.

IBF — Input buffer full; IBF is set true when a command or data is written to the input buffer. The DEU sets this flag false when it has accepted the input byte. No data should be written when IBF = 1.

DEC — Decode; indicates whether the DEU is in encrypt or decrypt mode. Decrypt: DEC = TRUE; Encrypt: DEC = FALSE.

HS — Handshake flag; this flag is used in the data transfer protocol.

KPE — Key Parity Error; after a new key has been entered, the DEU will use this flag in conjunction with the HS flag to indicate correct or incorrect parity.

Command Input Buffer — Commands to the DEU are written to this register.

MASTER/SLAVE INTERFACE

Figures 1 through 4 illustrate four interface configurations used in Master/Slave data transfers. In all cases SRQ will be true (if enabled) and IBF will be false when the DEU is ready to accept data or commands.

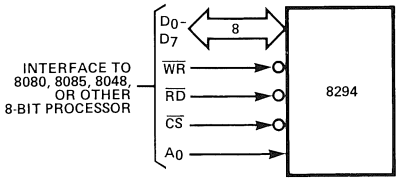


Figure 1. Polling Interface

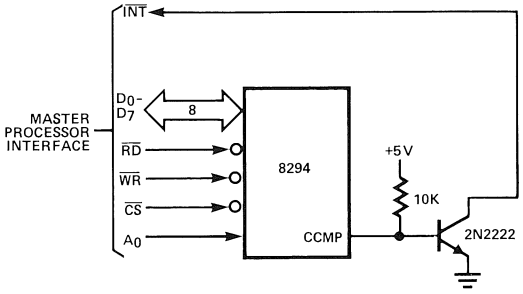


Figure 2. Single Interrupt Interface

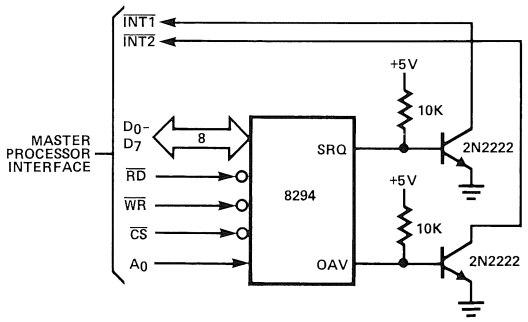
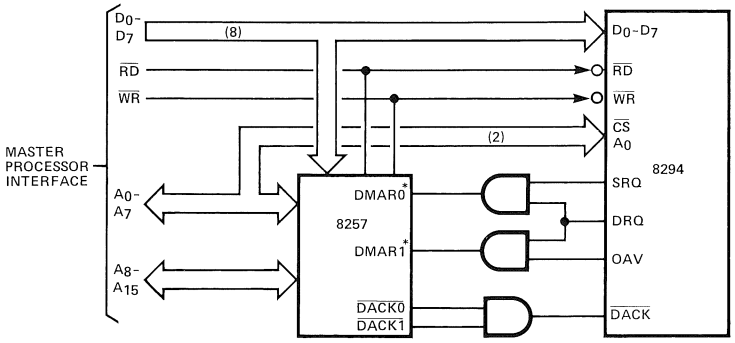


Figure 3. Dual Interrupt Interface



*DMAR0 IS FOR MEMORY TO DEU DATA TRANSFER
DMAR1 IS FOR DEU TO MEMORY DATA TRANSFER

Figure 4. DMA Interface

INTERFACE TIMING

Figures 5 through 8 illustrate recommended protocol sequences and timing for transferring commands and data between the master processor and the 8294.

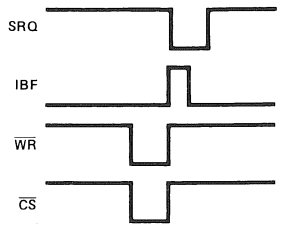


Figure 5. Single Byte Command

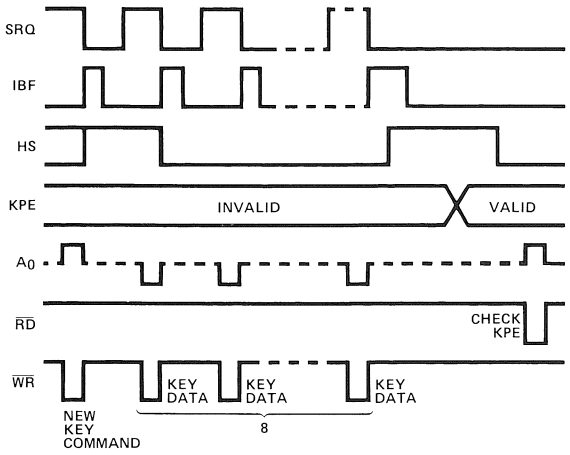


Figure 6. New Key Command

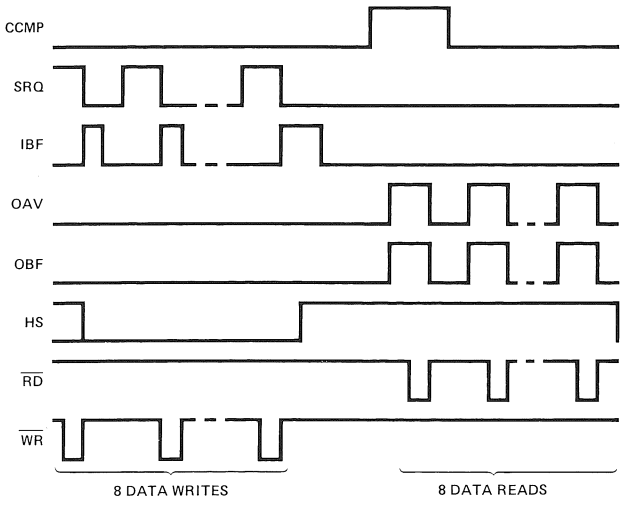


Figure 7. Encrypt/Decrypt Data

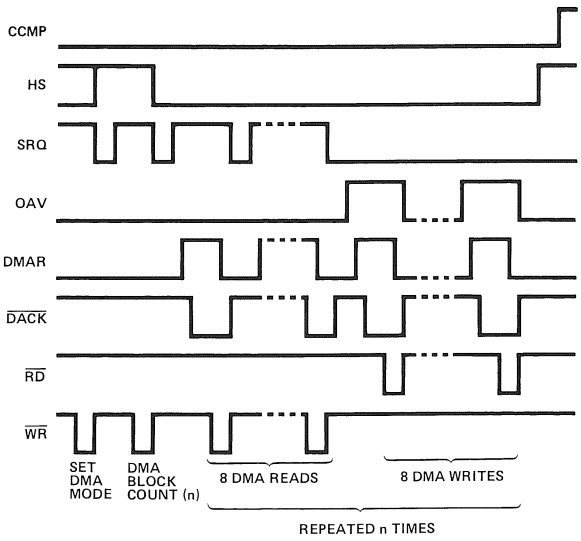


Figure 8. DMA Sequence