AN583

Implementation of the Data Encryption Standard Using PIC17C42

INTRODUCTION

In January 1977, The United States government adopted a product cipher developed by IBM® as its official encryption standard [1]. This algorithm, called the Data Encryption Standard (DES), has been adopted as a worldwide standard for data encryption by ISO (International Standards Organization) [2, 3]. This application note describes the implementation of the DES algorithm on PIC17C42.

THE DATA ENCRYPTION STANDARD

The DES algorithm is a substitution cipher which takes a block of 64 bits of input (plaintext) into a unique block of 64 bits of output (ciphertext), under the control of a 64 bit key, which is known only to the people intended to read the message. In this system, plaintext information is divided into several blocks which are then operated upon independently to generate a sequence of ciphertext blocks. The basic idea behind DES is to build a strong system out of simple, individually weak, components. The DES cryptosystem is based on a system of transpositions and permutations. The permutation box or P-box, is used to transpose, or map a sequence of input values to another sequence of values of the same length. Substitutions are performed by what is called S-boxes. A combination of the S-boxes and P-boxes can be viewed as a decoder/coder operation, where the output is simply a linear mapping of the input values. Each combination of the S-box and P-box comprises a single weak component of the algorithm. By including a sufficiently large number of stages in the product cipher, the output can be made to be a non-linear function of the input.

The mapping of input to output is one-to-one and invertible, since the encrypted messages can be decrypted. The DES has three distinct components: key schedule, cipher function and invertibility.

KEY SCHEDULE

The DES uses a 64-bit key for encryption and decryption process. Initially, the original 64-bit key is reduced to 56-bit by ignoring every eighth bit. In general these bits are used as parity bits to make sure that there were no errors when entering the key or during key transmission. After the 56-bit key is extracted a different 48-bit key, referred to as subkey, is generated for each of the 16 rounds of the DES. These keys, K_i, are determined as shown in Figure-1. The 56-bit key is divided into two 28-bit halves C_i and D_i which are then shifted left by either 1 or 2 digits, depending on the round. Table 1 shows the number of circular left shifts for C_i and D_i halves. After the shifting operation, 48 out of the 56 bits are selected. Since this operation permutes the order of the bits as well as selecting a subset of the original bits, it is called compression permutation or permuted choice. The permuted choice 1 and permuted choice 2 matrices are shown in Figure 2 and Figure 3 respectively.

TABLE 1: LEFT SHIFTS FOR KEY GENERATION

Iteration	# of left shifts
1	1
2	1
3	2
4	2
5	2
6	2
7	2
8	2
9	1
10	2
11	2
12	2
13	2
14	2
15	2
16	1

FIGURE 1 - KEY GENERATION

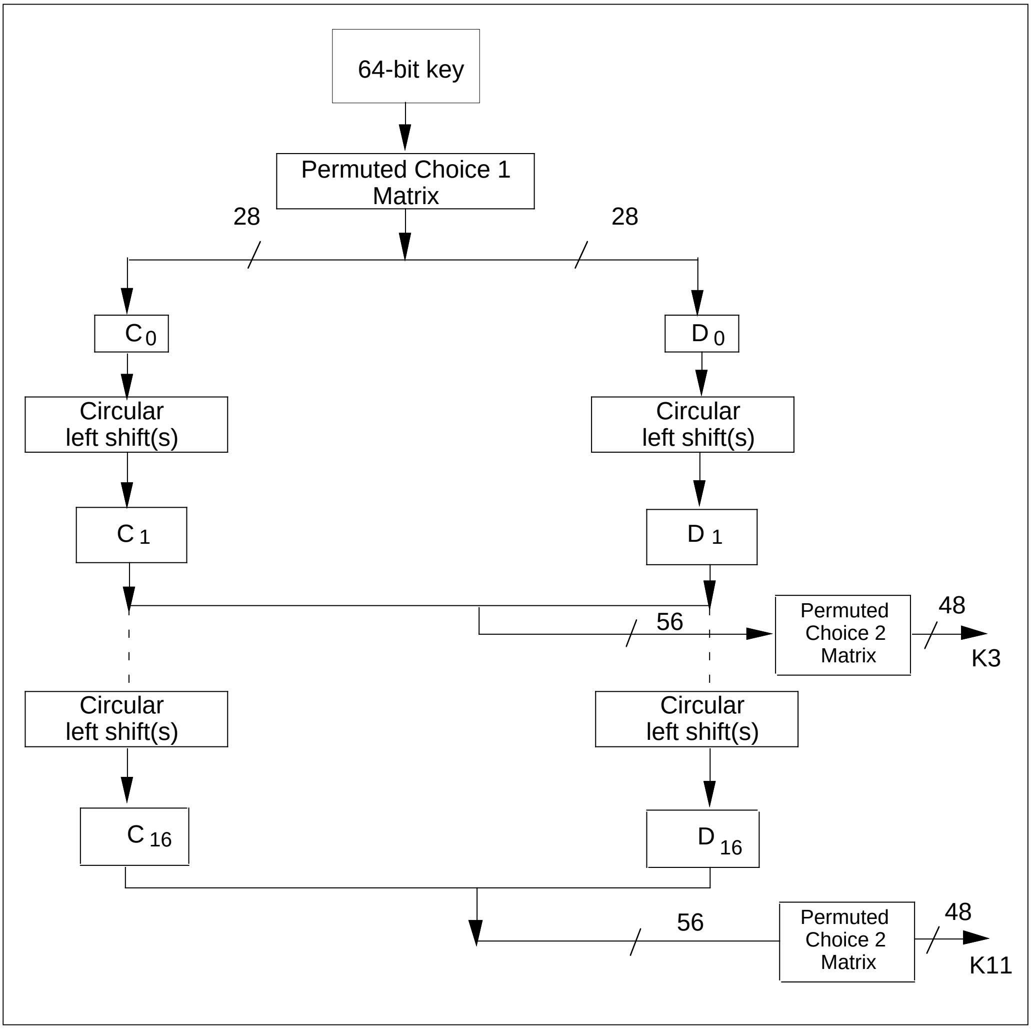


FIGURE 2 - PERMUTED CHOICE 1 MATRIX

57	49	41	33	25	17	9	1
58	50	42	34	26	18	10	2
59	51	43	35	27	19	11	3
60	52	44	36	63	55	47	39
31	23	15	7	62	54	46	38
30	22	14	6	61	53	45	37

FIGURE 3 - PERMUTED CHOICE 2 MATRIX

14	17	11	24	1	5	3	28
15	6	21	10	23	19	12	4
26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

CIPHER FUNCTION

The strength of the DES is based on the cipher function component. This is a fixed, highly non-linear function which guarantees that each bit of the ciphertext depend on every bit of the plaintext and every bit of the key.

After an initial permutation, the 64-bit block of plaintext is broken into a right half and left half, each 32 bits long. This step is followed by 16 identical rounds of operation, called function f, that combines the data with a 48-bit key, K_i. At each stage i, the inputs are the left block L_{i-1} and the right block R_{i-1} of the previous stage, and the outputs are the left shift block L_i and right block R_i of this stage. The outputs of L_i and R_i of each stage are computed from L_{i-1} and R_{i-1}, and a subkey K_i that is generated from the encryption key. In other words a round of the DES can be shown as:

$L_i = R_{i-1}$

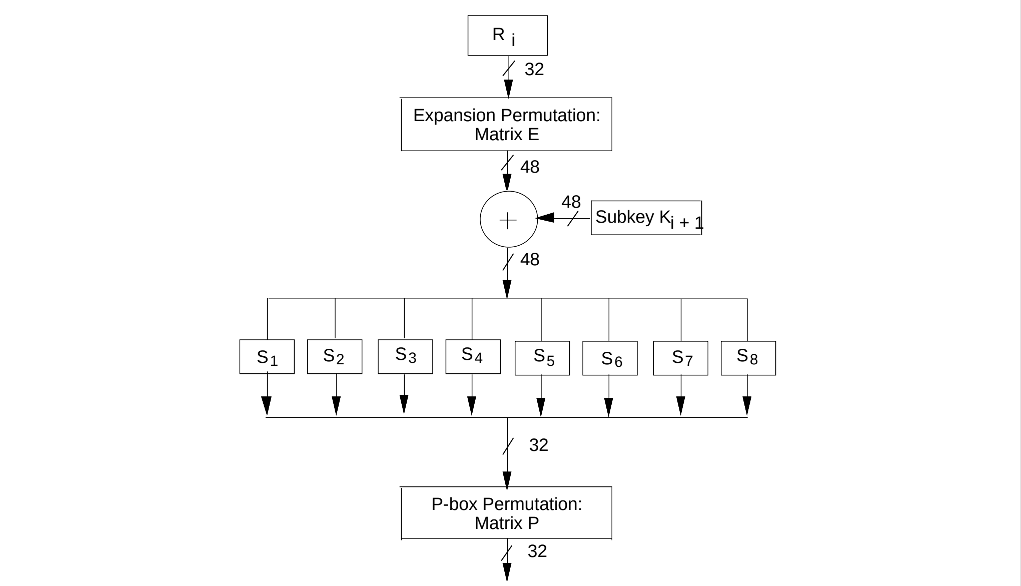
$R_i = L_{i-1}$ XOR $f(R_{i-1}, K_i)$

All the complexity of the DES algorithm lies in the function f, as shown in Figure-4. The function has four steps that are carried out in sequence. First a 48-bit number, E, is constructed by expanding the 32-bit previous right value, R_{i-1}, according to a fixed transposition and duplication rule. Then, K_i and R_i-1 are XORed together, generating a 48-bit result. This output is then partitioned into eight groups of 6 bits each, each of which is fed into a different S-box or substitution box. The S-boxes generate four instead of six outputs. In other words, the 64-bit input is mapped into a 32-bit output. Each S-box is a table of 4 rows and 16 columns. Each entry in the box is a 4-bit number. The six input bits of the S-box specify under which row and column number to look for the output. Figure-7 shows the 8 S-boxes.

The 6 input bits specify an entry in the S-box in a particular fashion as follows: the first and last bits of the sequence, taken together, represent a number between 0 and 3 (row entry), while the middle 4-bits represent a number between 0 and 15 (column entry). The output is simply the entry that corresponds to the (row, column) entry. For example, that the input to the first S-box is the binary sequence 110010. The first and last bits combine to form 10 which corresponds to the third row of the S-box. The middle four bits are combined to form 1001 which corresponds to the ninth column of the S-box. The corresponding entry in the first S-box is 12. Therefore the value of 1100 is substituted for 110010. The substitution boxes are the most critical step in the DES algorithm and more than anything else give DES its security.

Finally, the last stage consists of a permutation stage that generates a 32-bit output. After the 16 rounds, the left and right halves are joined, and a final permutation generates the ciphertext. The final permutation is the inverse of the initial permutation. Figure-8 shows a block diagram of the enciphering portion of the algorithm. While the reverse process of deciphering is shown in Figure 11. The initial permutation and inverse initial permutation matrices in Figure-8 are shown in Figures 9 and 10. Where the algorithm requires bit manipulation of a stream of data according to a matrix, the matrix is read from left to right, top to bottom, and interpreted as the bit position in the output block. For example, the initial permutation matrix transposes bit1 to bit58, bit2 to bit50, bit3 to bit42, etc.

FIGURE 4 - FUNCTION f (Ri, K_{i+1})



INVERTIBILITY

The DES cipher function is not necessarily invertible, meaning to decode a message, it is not necessary to recover the input to the cipher function from its output and a knowledge of the key. In fact the cipher function must be highly non-linear to be resistant to plaintext attack (a method used for breaking a given algorithm). Invertibility of the DES is that one half-word of the output is precisely the bit configuration which was used to encode the other half, with the aid of the particular stage subkey. Therefore, by using the subkeys in reverse order, the encryption process can be reversed. This is really the reason that one half-word is always passed through unchanged - to provide the means of decrypting the other half-word.

PIC17C42 IMPLEMENTATION OF DES

CPU processing is required to generate the encryption key into the DES subkeys. The 64-bit encryption key is reduced to 56-bits, by ignoring every eighth bit, usually used as parity bit.

The majority of the DES code is for the Implementation of the permutation of the block of bits. The 56-bits of the key, stored in K1 through K8, scrambled-bit output is stored in the eight bytes D0-D7. The scrambling is accomplished by constructing D0-D7, one bit at a time. This is accomplished by initializing the D0 to D7 locations to a known state (cleared). Then the 64-bits of plain text are processed through the Initial Permutation Matrix (IP), which permutes the plain text and divides the information into two 32-bit blocks.

The use of Indirect addressing and the PIC16/17's single word instructions allows tight efficient coding of the DES algorithm. These bit testing capabilities allows the same code structure to generate the different subkey blocks. This permutation macro looks like:

```
Permute   Macro    KEY,TEST,BIT
          BTFS C   KEY,TEST
          BSF      INDOF0,BIT
          endm
```

Where KEY is the DES key and TEST is the bit is the KEY being tested. If the KEY<TEST> bit is set, then the bit position (BIT) in the data location pointed to by INDF0 is set.

The main algorithm requires that the 16 subkeys, each 48-bits long, be generated. These 16 subkeys are then used at the 16 stages of the algorithm.

Using the generated subkeys, the incoming stream of bits can be encrypted or decrypted. Table 2 shows the requirements of the DES algorithm.

TABLE 2: DES ALGORITHM REQUIREMENTS

Function	Program Memory (words)	Execution time	
		Instruction cycles	ms
Key management and subkey generation	382	2729	0.436
Encryption	798	7714	1.234

A bit rate of about 51 kbps baud can be achieved, with a device utilization of 100%. This makes the PIC17C42 a price/performance leader for DES algoritms.

CONCLUSION

We have demonstrated the implementation of the DES algorithm on the PIC17C42 microcontroller. The 160 ns cycle time of the PIC17C42 makes possible a half-duplex rate of approximately 51 kbps for the DES. This rate is as good or superior to other implementations of the algorithm. The high performance of the PIC17C42 provides a low cost alternative to many dedicated solutions resulting in minimum system cost because of the programmability of the device.

References

1. NBS FIPS PUB 46, “Data Encryption Standard,” National Bureau of Standards, US Department of Commerce, January 1977.
2. ISO DIS 8730, “Banking Requirements for Message Authentication (Wholesale),” Association for Payment Clearing Services, London, July 1987.
3. ISO DIS 8732, “Banking Key Management (Wholesale),” Association for Payment Clearing Services, London, December 1987.

Authors: *Al Lovrich*
Mark Palmer
Logic Products Division

FIGURE 5 - MATRIX E

32	1	2	3	4	5	4	5
6	7	8	9	8	9	10	11
12	13	12	13	14	15	16	17
16	17	18	19	20	21	20	21
22	23	24	25	24	25	26	27
28	29	28	29	30	31	32	1

FIGURE 6 - MATRIX P

16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

FIGURE-7 - S MATRICES

```
/* S1 */
14  4 13  1  2 15 11  8  3 10  6 12  5  9  0  7
 0 15  7  4 14  2 13  1 10  6 12 11  9  5  3  8
 4  1 14  8 13  6  2 11 15 12  9  7  3 10  5  0
15 12  8  2  4  9  1  7  5 11  3 14 10  0  6 13

 /* S2 */
15  1  8 14  6 11  3  4  9  7  2 13 12  0  5 10
 3 13  4  7 15  2  8 14 12  0  1 10  6  9 11  5
 0 14  7 11 10  4 13  1  5  8 12  6  9  3  2 15
13  8 10  1  3 15  4  2 11  6  7 12  0  5 14  9

/* S3 */
10  0  9 14  6  3 15  5  1 13 12  7 11  4  2  8
13  7  0  9  3  4  6 10  2  8  5 14 12 11 15  1
13  6  4  9  8 15  3  0 11  1  2 12  5 10 14  7
 1 10 13  0  6  9  8  7  4 15 14  3 11  5  2 12

/* S4 */
 7 13 14  3  0  6  9 10  1  2  8  5 11 12  4 15
13  8 11  5  6 15  0  3  4  7  2 12  1 10 14  9
10  6  9  0 12 11  7 13 15  1  3 14  5  2  8  4
 3 15  0  6 10  1 13  8  9  4  5 11 12  7  2 14

/* S5 */
 2 12  4  1  7 10 11  6  8  5  3 15 13  0 14  9
14 11  2 12  4  7 13  1  5  0 15 10  3  9  8  6
 4  2  1 11 10 13  7  8 15  9 12  5  6  3  0 14
11  8 12  7  1 14  2 13  6 15  0  9 10  4  5  3

 /* S6 */
12  1 10 15  9  2  6  8  0 13  3  4 14  7  5 11
10 15  4  2  7 12  9  5  6  1 13 14  0 11  3  8
 9 14 15  5  2  8 12  3  7  0  4 10  1 13 11  6
 4  3  2 12  9  5 15 10 11 14  1  7  6  0  8 13

/* S7 */
 4 11  2 14 15  0  8 13  3 12  9  7  5  0  6  1
13  0 11  7  4  9  1 10 14  3  5 12  2 15  8  6
 1  4 11 13  2  3  7 14 10 15   6  8  0  5  9  2
 6 11 13  8  1  4 10  7  9  5  0 15 14  2  3 12

/* S8 */
13  2  8  4  6 15 11  1 10  9  3 14  5  0 12  7
 1 15 13  8 10  3  7  4 12  5  6 11  0 14  9  2
 7 11  4  1  9 12 14  2  0  6 10 13 15  3  5  8
 2  1 14  7  4 10  8 13 15 12  9  0  3  5  6 11
```

FIGURE 9 - INITIAL PERMUTATION MATRIX

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

FIGURE 10 - INVERSE PERMUTATION MATRIX

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

FIGURE 8 - DES ENCRYPTION BLOCK DIAGRAM

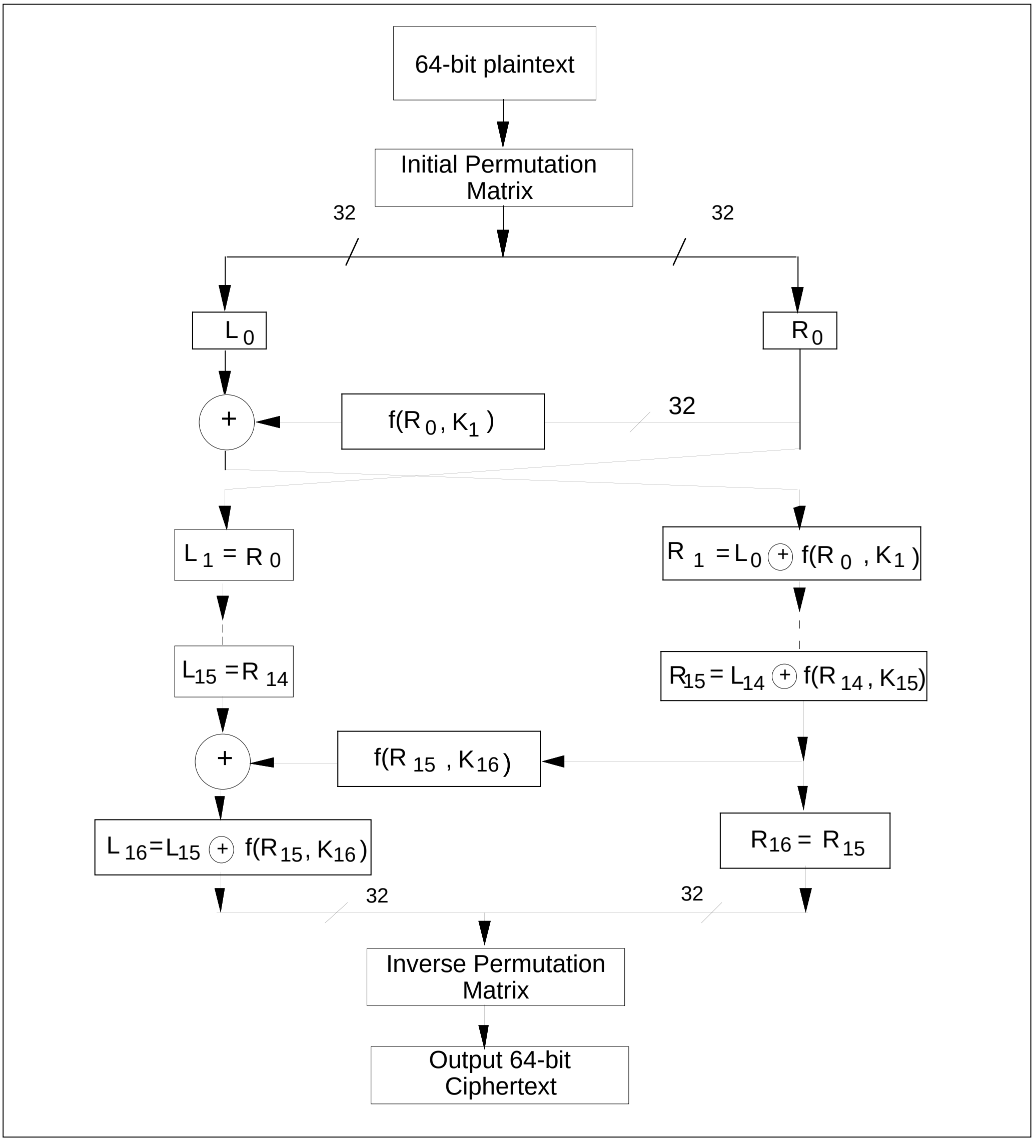
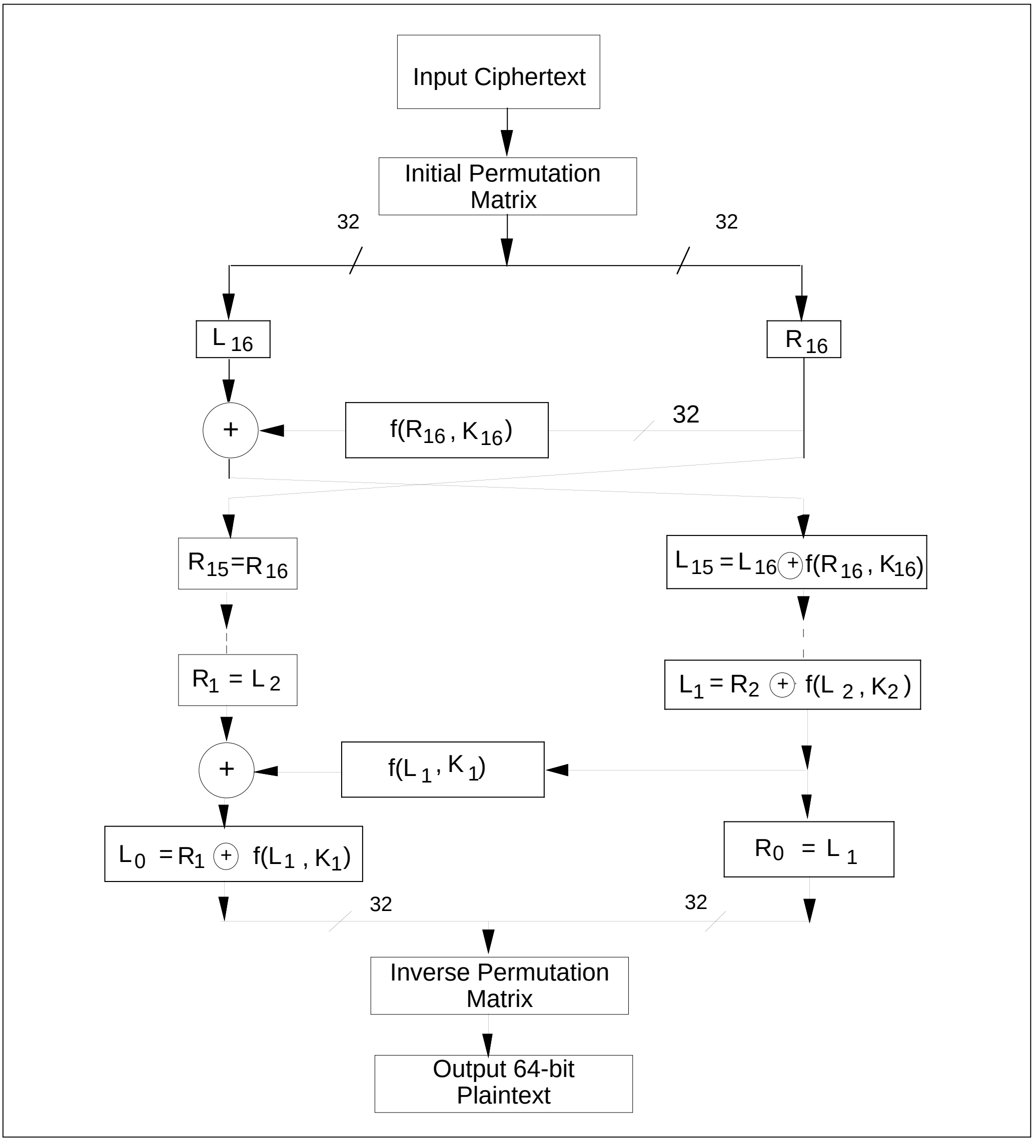

64-bit plaintext
Initial Permutation Matrix
32
32
L_0
R_0
+
$f(R_0, K_1)$
32
$L_1 = R_0$
$R_1 = L_0 \oplus f(R_0, K_1)$
$L_{15} = R_{14}$
$R_{15} = L_{14} \oplus f(R_{14}, K_{15})$
+
$f(R_{15}, K_{16})$
$L_{16} = L_{15} \oplus f(R_{15}, K_{16})$
$R_{16} = R_{15}$
32
32
Inverse Permutation Matrix
Output 64-bit Ciphertext

FIGURE 11 - DES DECRYPTION BLOCK DIAGRAM


Input Ciphertext
Initial Permutation Matrix
32
32
L_{16}
R_{16}
+
$f(R_{16}, K_{16})$
32
$R_{15}=R_{16}$
$L_{15} = L_{16} \oplus f(R_{16}, K_{16})$
$R_1 = L_2$
$L_1 = R_2 \oplus f(L_2, K_2)$
+
$f(L_1, K_1)$
$L_0 = R_1 \oplus f(L_1, K_1)$
$R_0 = L_1$
32
32
Inverse Permutation Matrix
Output 64-bit Plaintext

APPENDIX A

NOTE: The PIC17C42 code implementing the DES is not published because it falls within the U.S. Department of State Export Control Regulations.

Please contact your local Microchip sales office to obtain a copy of the code.

WORLDWIDE SALES & SERVICE

AMERICAS

Corporate Office

Microchip Technology Inc.
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 602 786-7200 Fax: 602 786-7277
Technical Support: 602 786-7627
Web: http://www.mchip.com/microhip

Atlanta

Microchip Technology Inc.
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770 640-0034 Fax: 770 640-0307

Boston

Microchip Technology Inc.
5 Mount Royal Avenue
Marlborough, MA 01752
Tel: 508 480-9990 Fax: 508 480-8575

Chicago

Microchip Technology Inc.
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 708 285-0071 Fax: 708 285-0075

Dallas

Microchip Technology Inc.
14651 Dallas Parkway, Suite 816
Dallas, TX 75240-8809
Tel: 214 991-7177 Fax: 214 991-8588

Dayton

Microchip Technology Inc.
35 Rockridge Road
Englewood, OH 45322
Tel: 513 832-2543 Fax: 513 832-2841

Los Angeles

Microchip Technology Inc.
18201 Von Karman, Suite 455
Irvine, CA 92715
Tel: 714 263-1888 Fax: 714 263-1338

New York

Microchip Technology Inc.
150 Motor Parkway, Suite 416
Hauppauge, NY 11788
Tel: 516 273-5305 Fax: 516 273-5335

AMERICAS (continued)

San Jose

Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408 436-7950 Fax: 408 436-7955

ASIA/PACIFIC

Hong Kong

Microchip Technology
Unit No. 3002-3004, Tower 1
Metroplaza
223 Hing Fong Road
Kwai Fong, N.T. Hong Kong
Tel: 852 2 401 1200 Fax: 852 2 401 3431

Korea

Microchip Technology
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku,
Seoul, Korea
Tel: 82 2 554 7200 Fax: 82 2 558 5934

Singapore

Microchip Technology
200 Middle Road
#10-03 Prime Centre
Singapore 188980
Tel: 65 334 8870 Fax: 65 334 8850

Taiwan

Microchip Technology
10F-1C 207
Tung Hua North Road
Taipei, Taiwan, ROC
Tel: 886 2 717 7175 Fax: 886 2 545 0139

EUROPE

United Kingdom

Arizona Microchip Technology Ltd.
Unit 6, The Courtyard
Meadow Bank, Furlong Road
Bourne End, Buckinghamshire SL8 5AJ
Tel: 44 0 1628 851077 Fax: 44 0 1628 850259

France

Arizona Microchip Technology SARL
2 Rue du Buisson aux Fraises
91300 Massy - France
Tel: 33 1 69 53 63 20 Fax: 33 1 69 30 90 79

Germany

Arizona Microchip Technology GmbH
Gustav-Heinemann-Ring 125
D-81739 Muenchen, Germany
Tel: 49 89 627 144 0 Fax: 49 89 627 144 44

Italy

Arizona Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Pegaso Ingresso No. 2
Via Paracelso 23, 20041
Agrate Brianza (MI) Italy
Tel: 39 039 689 9939 Fax: 39 039 689 9883

JAPAN

Microchip Technology Intl. Inc.
Benex S-1 6F
3-18-20, Shin Yokohama
Kohoku-Ku, Yokohama
Kanagawa 222 Japan
Tel: 81 45 471 6166 Fax: 81 45 471 6122

9/22/95

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights. The Microchip logo and name are registered trademarks of Microchip Technology Inc. All rights reserved. All other trademarks mentioned herein are the property of their respective companies.