DES function

\[ f(A, J) \leq \]
\[ A' = E(A) \]
\[ B = A' \oplus J = b_1b_2 \ldots b_8 \]
\[ \text{for } (i = 1; i < 8; i++) \]
\[ c_i = S_i (b_i) \rightarrow 6 \text{ bits} \]
\[ C = c_1c_2 \ldots c_8 \]
\[ \text{return } P(C) \]

\[ \exists \]

S-box Criteria

P0) each row is a perm of 0,1,...,15
P1) no S-box is a linear or affine function of its inputs.
P2) Changing 1 input bit to an S-box causes at least 2 output bits to change.
P3) For any S-box any input \( x \), \( S(x) \) and \( S(x \oplus 001100) \) differ in at least 2 bits.

High Level

\[ X_0 = IP(x) = L_0R_0 \]
\[ \text{for } (i = 1; i < (6;} i++) \]
\[ L_i = R_{i-1} \]
\[ R_i = L_{i-1} \oplus f(k_i, y) \]
\[ \exists \]
\[ \text{return } IP^{-1}(R_{16}L_{16}) \]
For any S-box, any input \( x \), and for \( e, f \in \{0,1\} \\
S(x) \neq S((x \oplus 1) \oplus e \oplus f) \\
S(x) \neq S(x \oplus 110000) \\
S(x) \neq S(x \oplus 110100) \\
S(x) \neq S(x \oplus 111000) \\
S(x) \neq S(x \oplus 111100) \\

1.5) For any S-box, if 1 input bit is fixed, and we look at one output bit, the # of 0s and 1s needs to be 
"pretty close" (\( |\text{#0s} - \text{#1s}| \leq 6 \))

There are 32 inputs with 1 bit fixed.
Consider looking at the 32 outputs of 4
bits + choosing a specific bit to look at
(say bit 3), then we must have in between 13 0s and 19 0s in those 32 outputs at that position.

Key Schedule

Key is 56 bits expressed in 64 bits
w/parity bits \( K_8, K_{16}, K_{24}, \ldots, K_{64} \) (06h, 07h, 08h, 09h)

\[
\begin{align*}
K_1 & K_2 K_3 \ldots K_8 \\
K_9 & K_{10} K_{11} \ldots K_{16} \\
& \vdots \\
K_{57} & K_{58} K_{59} \ldots K_{64} \\
\end{align*}
\]

\[
\begin{align*}
11011001 & \text{ sum(1) = 5} \\
00011100 & \text{ sum(1) = 3}
\end{align*}
\]

We use the 56 bits to create 16 Round Keys, \( K_1, K_2, \ldots, K_{16} \) each of which are 48 bits.
Key Schedule Algorithm (DES)

1. Permute key bits according to PC-1.
   \[ PC-1(K) = C_0 \quad D_0 \]
   
   
   \[
   \begin{array}{c}
   28 \text{ bits} \\
   28 \text{ bits} \\
   \end{array}
   \]

2. 16 Rounds
   
   for \( i = 1 \) to \( 16 \) do
     \[
     C_i = LS_i(C_{i-1})
     \]
     \[
     D_i = \overline{LS_i}(D_{i-1})
     \]
   
   \[
   K_i = PC-2(C_i \quad D_i)
   \]

\[ \exists \]

\( LS_i \) is a cyclic left shift of either 1 or 2 bits.

if \( i = 1, 2, 9 \) or \( 16 \)
   \[ LS_i = 1 \]
else
   \[ LS_i = 2 \]

12 rounds \times 26 bits + 4 \times \text{round} \times 1 \text{ bit} = 28 \text{ bits}
57, 49, 41, 33, 25 .... (After Step 1)

LS 1 bit on 1st 28 values:
\[ C_1 = (49, 41, 33, 25, ... , 44, 36, 57) \]
\[ D_1 = (55, 47, 39, ... , 12, 4, 63) \]

PC-2:
\[ 14, 17, 11, 24, 1, 5, 3, 28 \]

\[ 10, 51, 34, 60 \]

Location in the original key:
\[ 14, 17, 11, 24, 1, 5, 3, 28 \]