For faster operation, the subkey should be pre computed and stored in cache for faster encryption.

International Data encryption Algorithms (IDEA)

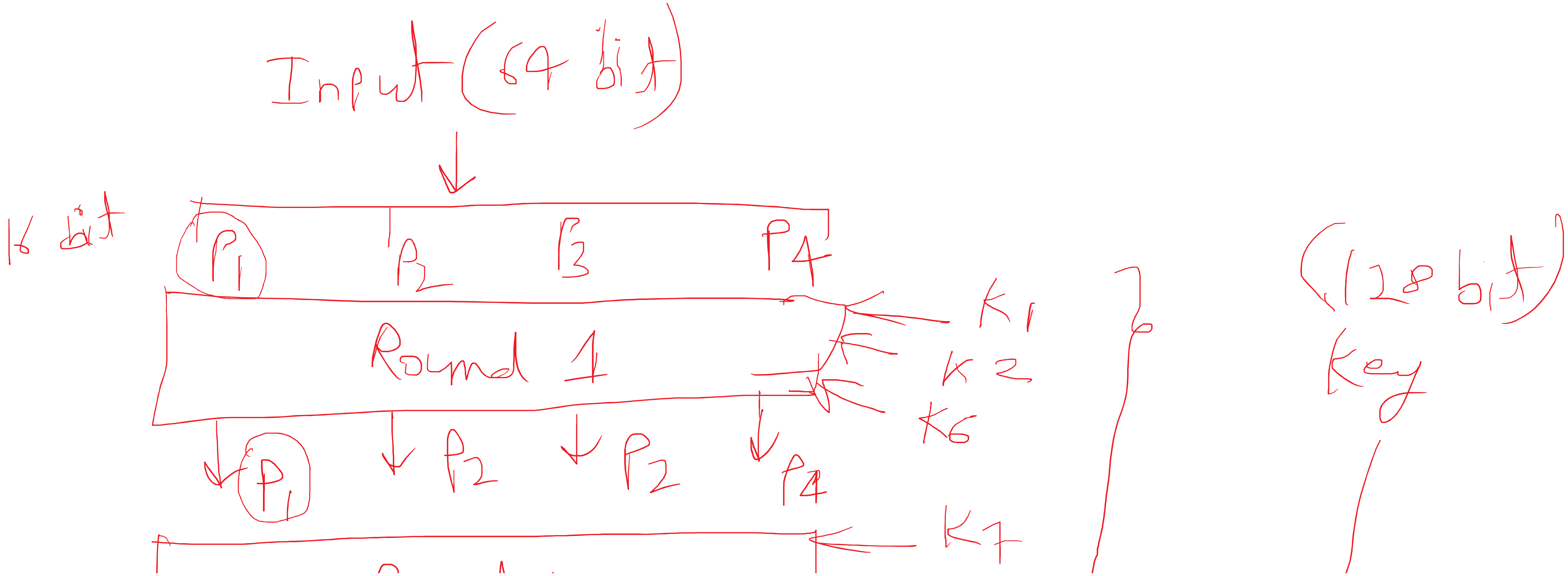
(James Massey, Lai - 1990)

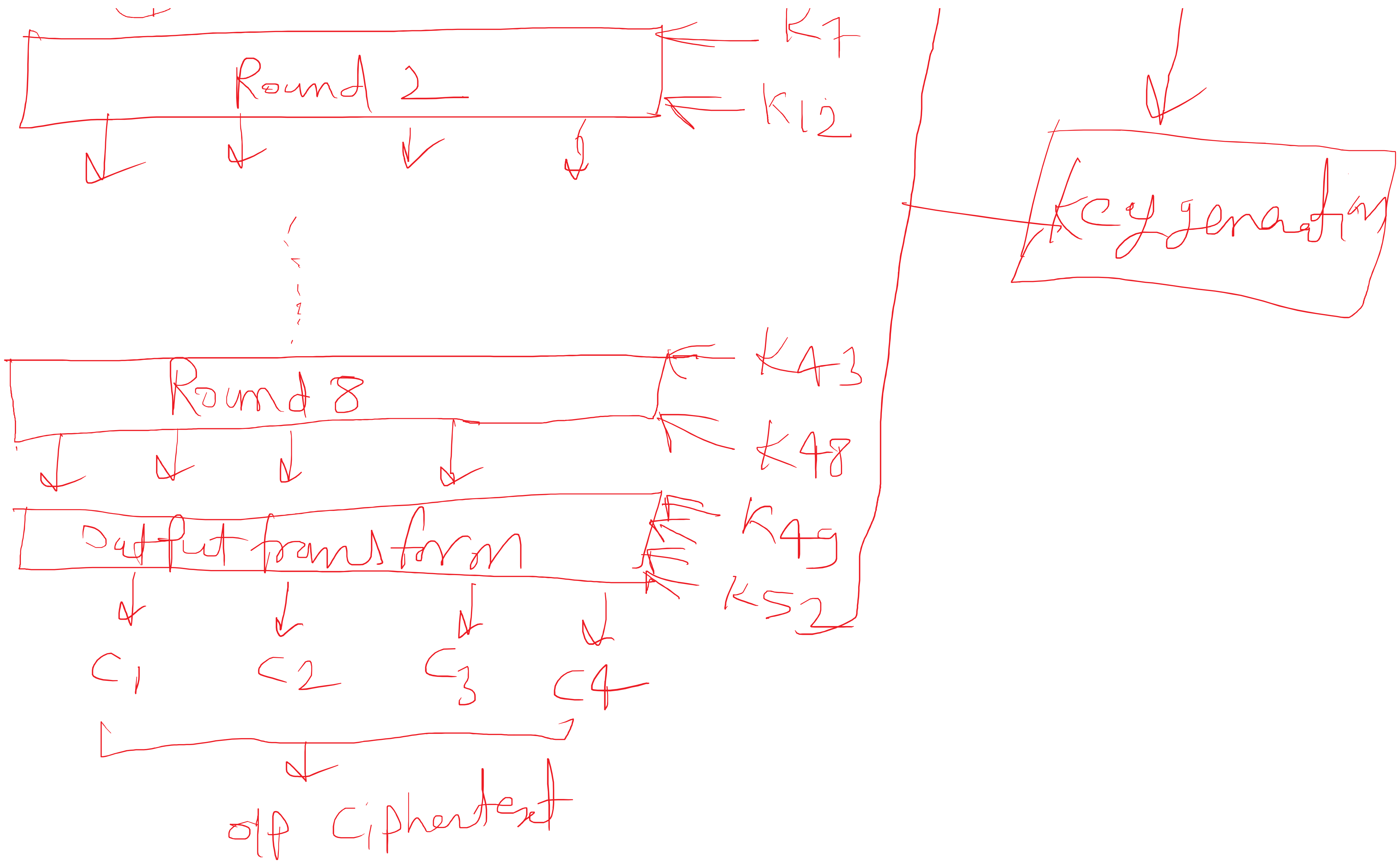
- Block cipher (64 bit)
- Symmetric (Sender & Receiver same key)
- Reversible like DES (Decry is just Reverse of Encryption)
- Design Principle behind IDEA is the mixing of Arithmatic operation from different algebraic

groups, which are easily implementable on H/W and S/W.

- Underline Operation are XOR, add, mult.
- It does not use S-Box (explicitly)
- Key size 128 → sub key (16 bit)

Structure IDEA : 8 Round





Round functionality :

each Round (P_1, P_2, P_3, P_4)

1. multiply $P_1 * K_1 \rightarrow S_1$
2. add $P_2 + K_2 \rightarrow S_2$
3. add $P_3 + K_3 \rightarrow S_3$
4. multiply $P_4 * K_4 \rightarrow S_4$

4 key

5. XOR $S_1 \oplus S_3 \rightarrow S_5$
6. XOR $S_2 \oplus S_4 \rightarrow S_6$

No key

7 mult $\rightarrow$ $S_5 * K_5 \rightarrow S_7$

8 add $S_6 + S_7 \rightarrow S_8$

9 mult $\rightarrow$ $S_8 * K_6 \rightarrow S_9$

10 add $S_7 + S_9 \rightarrow S_{10}$

2 Key

$$\left[\begin{array}{lll} 11 & \text{XOR} & S_1 \oplus S_9 \to S_{11} \to \text{New } P_1 \\ 12 & \text{XOR} & S_2 \oplus S_9 \to S_{12} \to \text{New } P_2 \\ 13 & \text{XOR} & S_3 \oplus S_{10} \to S_{13} \to \text{New } P_3 \\ 14 & \text{XOR} & S_4 \oplus S_{10} \to S_{14} \to \text{New } P_4 \end{array}\right\} \begin{array}{l} \text{O/P} \\ \text{of} \\ \text{Round 1} \end{array}$$

Repeat same up to Round 8, so total 48 sub key is used till Now.

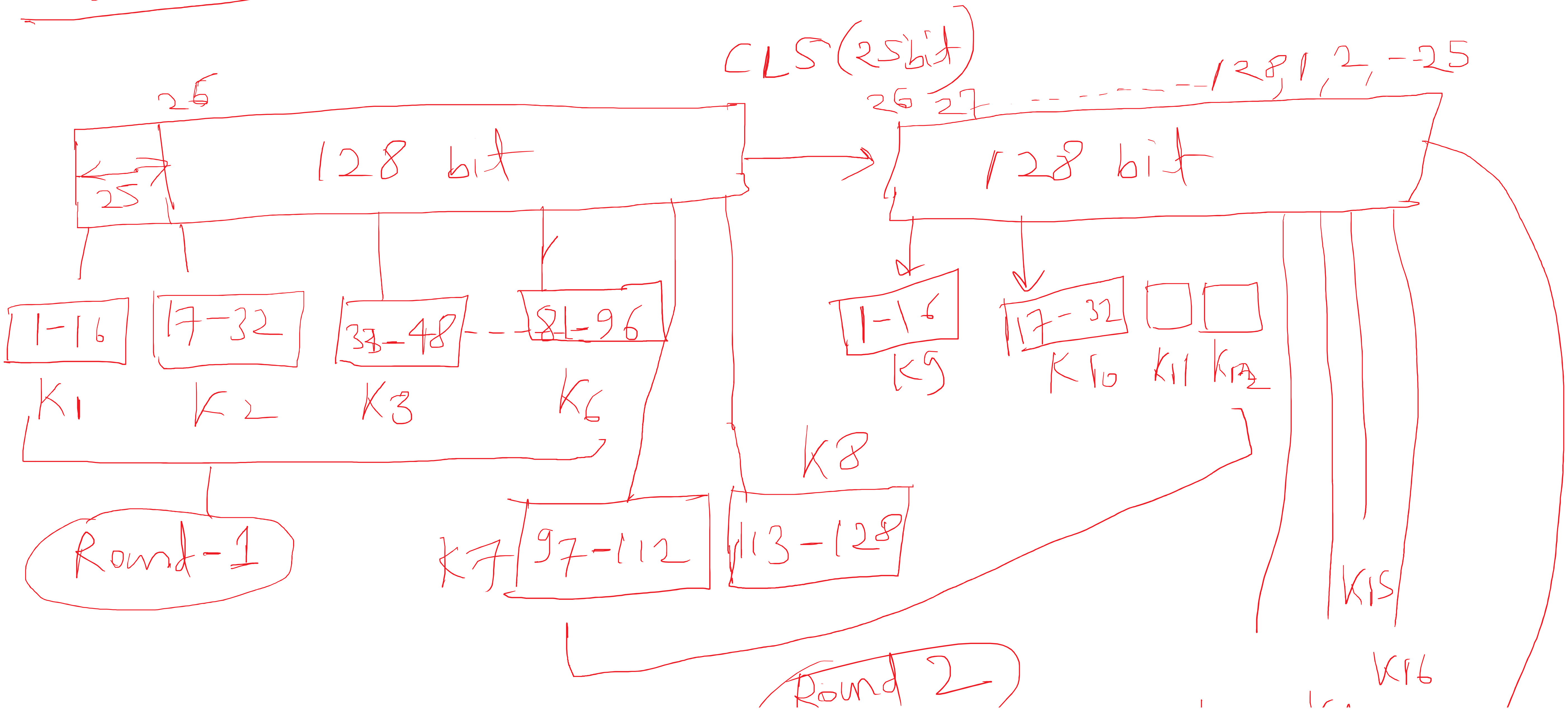
after 8th Round 4 more operation is need to apply in output transform step.

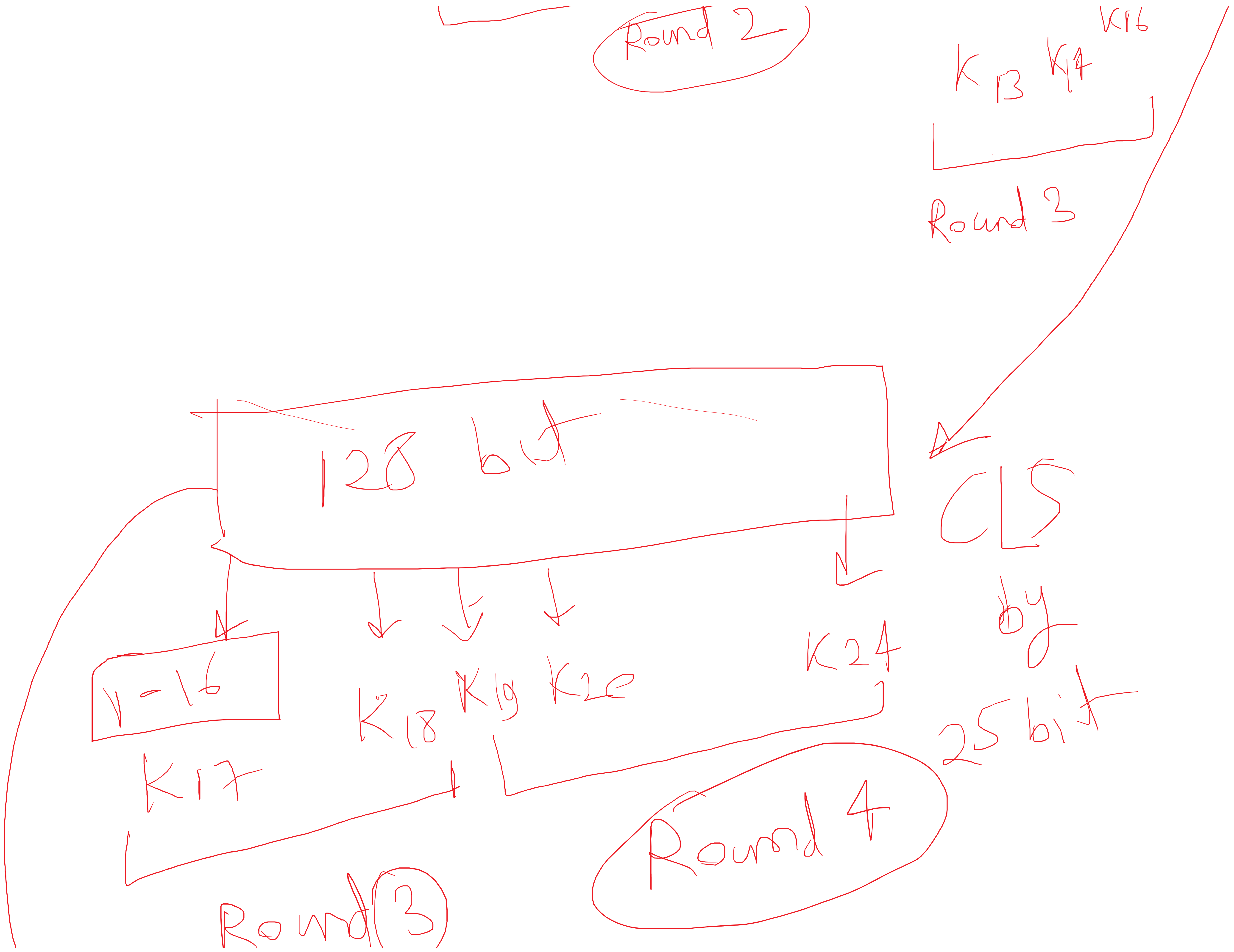
$$\left.\begin{array}{l} P_1 * K_{49} \longrightarrow C_1 \\ P_2 + K_{50} \longrightarrow C_2 \\ P_3 + K_{51} \longrightarrow C_3 \\ P_4 * K_{52} \longrightarrow C_4 \end{array}\right\} \begin{array}{l} \text{final} \\ \text{ciphertext} \\ \text{(64 bit)} \end{array}$$

P4 * K52 —— J64 (64 bit)

Total sub subkey = 48 + 4 = 52 key.

Key generation: given Key K : 128 bit




Round 2
K13 K14 K16
Round 3
128 bit
CLS
by
25 bit
1-16
K17
K18 K19 K20
K24
Round 3
Round 4

Round 3

CLS by 25 bit and so on to generate all 52 sub key.

CLS (3bit) 10110101 = 11010101

10110101

01101011

CLS

11010101

CLS
by 3 bit

CL 11010110

CL 11010101