

# 1. Penjelasan Apa itu Enigma

Enigma adalah sebuah mesin kriptografi elektromekanis yang dikembangkan oleh Jerman pada awal abad ke-20. Mesin ini digunakan secara luas oleh militer Jerman selama Perang Dunia II untuk mengenkripsi dan mendekripsi pesan rahasia. Enigma menjadi salah satu alat penyandian paling kuat pada masanya dan merupakan langkah penting dalam perkembangan ilmu kriptografi.

Contoh penggunaan Enigma dalam sejarah yang paling terkenal adalah oleh tentara Jerman selama Perang Dunia II. Mereka menggunakan mesin Enigma untuk mengirim pesan rahasia yang tidak dapat dengan mudah dipecahkan oleh pihak musuh. Misalnya, U-boat Jerman (kapal selam) menggunakan Enigma untuk mengirim pesan terenkripsi ke markas mereka. Pihak Sekutu berusaha keras untuk memecahkan kode Enigma untuk mendapatkan informasi intelijen tentang rencana dan gerakan musuh.

Namun, upaya untuk memecahkan kode Enigma terbukti sangat sulit karena kekuatan kriptografi mesin ini. Baru setelah pengembangan dan penggunaan mesin pemecah kode yang dikenal sebagai "Bombe" oleh Alan Turing dan tim Bletchley Park di Inggris, Sekutu berhasil memecahkan kode Enigma secara sistematis dan signifikan mengurangi efektivitas rahasia Jerman. Kontribusi Alan Turing dan tim Bletchley Park dalam memecahkan kode Enigma dianggap sebagai salah satu pencapaian terbesar dalam sejarah komputasi dan kriptografi. Enigma telah menjadi simbol dari perang kriptografi pada masa itu dan menjadi fondasi bagi perkembangan lebih lanjut dalam ilmu kriptografi modern.

## 2. Penjelasan Cara Kerja Enigma

Enigma machine terdiri dari empat komponen utama yang bekerja bersama untuk melakukan enkripsi dan dekripsi pesan:

### 1. Tahap 1: Keyboard

Keyboard digunakan untuk memasukkan input pengguna. Enigma machine merupakan mesin enkripsi simetris, yang berarti dapat digunakan baik untuk mengenkripsi maupun mendekripsi pesan dengan pengaturan yang sama. Keyboard digunakan untuk memasukkan teks biasa (plaintext) yang perlu dienkripsi atau teks terenkripsi (ciphertext) yang perlu didekripsi. Keyboard terdiri dari 26 tombol, masing-masing mewakili huruf abjad. Perhatikan bahwa keyboard dimulai dengan huruf QWERTZ bukan QWERTY, hal ini disesuaikan dengan kebiasaan penulisan huruf dalam bahasa Jerman.

### 2. Tahap 2: Plugboard

Setelah tombol ditekan pada keyboard, sinyal masuk ke plugboard yang menyediakan tahap pertama dalam proses enkripsi. Plugboard menggunakan prinsip dari substitusi cipher, yaitu bentuk enkripsi transposisi. Dalam plugboard, pasangan huruf ditukar posisinya menggunakan kabel pendek. Misalnya, huruf W akan digantikan dengan huruf D dan huruf D dengan huruf W, dengan menggunakan kabel (berwarna merah) yang menghubungkan kedua huruf tersebut. Pengaturan plugboard dicatat dalam buku kode, contohnya "DW VZ" berarti W diganti dengan D dan D diganti dengan W, serta V diganti dengan Z dan Z diganti dengan V.

### 3. Tahap 3: Rotors

Setelah melewati plugboard, huruf masuk ke tiga rotor berurutan dari kanan ke kiri. Setiap rotor mengubah huruf masukan dengan cara yang berbeda menggunakan kombinasi substitusi cipher dan Caesar cipher. Terdapat lima jenis rotor yang dapat dipilih dan diposisikan dalam mesin Enigma. Setiap rotor diidentifikasi menggunakan angka Romawi I hingga V. Pengaturan ini mencakup penggunaan rotor dan urutan posisinya. Setiap rotor memiliki pengaturan awal yang dapat diubah dengan memutar roda yang sesuai.

Setiap rotor memiliki sebuah notch (biasanya pada posisi tertentu, seperti pada huruf Q), yang berfungsi sebagai indikator untuk rotor di sebelah kirinya. Ketika notch pada rotor saat ini sesuai dengan posisi window rotor tersebut, rotor di sebelah kirinya juga akan berputar satu langkah. Dengan demikian, setiap kali sebuah tombol huruf ditekan dan rotor saat ini berputar, rotornya akan menggerakkan rotor-rotor di sebelah kirinya. Proses stepping rotor mempengaruhi posisi window rotor dan mengubah permutasi koneksi internal rotor. Ini menghasilkan pengaturan enkripsi yang terus berubah saat pesan dienkripsi, menjadikannya sulit untuk diprediksi atau ditembus.

Ada beberapa kondisi khusus di mana rotor dapat melakukan "double stepping" atau "dual stepping". Double stepping terjadi ketika sebuah rotor berputar dua kali pada satu tekanan tombol. Pada kasus mesin enigma 3 rotor dengan single notch, hal ini akan terjadi ketika rotor tengah berada di posisi notchnya sendiri.

Ring setting pada rotor adalah pengaturan posisi awal dari cincin di sekitar rotor. Cincin ini memiliki posisi angka atau huruf yang dapat diubah oleh pengguna. Pengaturan ini mempengaruhi hubungan antara window rotor (posisi yang terlihat pada lingkaran luar rotor) dan penghubung internal di dalam rotor. Ring setting menggeser hubungan koneksi internal rotor dengan posisi eksternal rotor. Dengan mengubah ring setting, pengguna dapat mengubah posisi awal enkripsi rotor, memberikan variasi tambahan pada proses enkripsi.

### 4. Tahap 4: Reflector

Setelah melewati tiga rotor dari kanan ke kiri, huruf masukan dipantulkan kembali (reflected) oleh rotor reflektor. Rotor reflektor memantulkan aliran listrik melalui rotor-rotor secara terbalik, yaitu dari kiri ke kanan, untuk tiga tahap enkripsi tambahan (Mengulangi kembali tahap 3 -> 2-> 1). Setelah keluar dari rotor reflektor, huruf terenkripsi kembali ke plugboard untuk substitusi cipher terakhir. Berbagai versi Enigma menggunakan reflector yang berbeda dengan menerapkan substitusi cipher yang berbeda pula.

### 5. Tahap 5: Lampboard

Setelah melalui proses enkripsi di rotor-rotor dan reflektor, huruf terenkripsi akan ditampilkan pada lampboard. Lampboard terdiri dari 26 lampu, masing-masing mewakili huruf abjad. Setiap lampu akan menyala untuk menunjukkan huruf terenkripsi yang sesuai.

Enigma machine menjadi sulit untuk ditembus karena setiap kali tombol ditekan, rotor paling kanan akan berputar satu langkah. Hal ini berarti pengaturan enkripsi terus berubah untuk setiap huruf dalam pesan.

### 3. Contoh step by step enkripsi Enigma

Akan dienkripsi teks "SUDO" dengan:

- Urutan rotor I, II, III
- Initial Key AAU.
- Plugboard SU
- Ring Offset AAU

Wiring rotor setiap notch adalah sebagai berikut:

```
'I': {'forward': 'EKMFLGDQVZNTOWYHXUSPAIBRCJ',  
      'backward': 'UWYGADFPVZBECKMTHXSLRINQOJ'},  
'II': {'forward': 'AJDKSIRUXBLHWTMCQGZNPYFVOE',  
       'backward': 'AJPCZWRLFBDKOTYUQGENHXMIVS'},  
'III': {'forward': 'BDFHJLCPRTXVZNYEIWGAKMUSQO',  
        'backward': 'TAGBPCSDQEUFVNZHYIXJWLRKOM'},
```

Reflektor:

```
{ 'A': 'Y', 'B': 'R', 'C': 'U', 'D': 'H', 'E': 'Q', 'F': 'S', 'G': 'L', 'H': 'D',  
  'I': 'P', 'J': 'X', 'K': 'N', 'L': 'G', 'M': 'O', 'N': 'K', 'O': 'M', 'P': 'I',  
  'Q': 'E', 'R': 'B', 'S': 'F', 'T': 'Z', 'U': 'C', 'V': 'W', 'W': 'V', 'X': 'J',  
  'Y': 'A', 'Z': 'T'  
}
```

Notch setiap rotor adalah sebagai berikut:

```
'I': 'Q',  
'II': 'E',  
'III': 'V',
```

**Note: Diadaptasi dari output program dengan penambahan penjelasan agar menjaga sanity saat liburan**

#### 1. Tahap 1, Current Key: AAV (+1 dari initial key)

INPUT: S

PLUGBOARD: U (S -> U)

Rotor III:

OFFSET: 21(V)

RING\_OFFSET: 20

input = U(20)

Process: WIRING[forward][(21 + 20 - 20) % 26] = M(12)

Output: (12 - 21 + 20) % 26 = 11 (L)

Rotor II:

OFFSET: 0(A)

RING\_OFFSET: 0

input = L(11)

Process: WIRING[forward][(0 + 11 - 0) % 26] = H(7)

Output: (7 - 0 + 0) % 26 = 7 (H)

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = H(7)  
Process: WIRING[forward][ $(0 + 7 - 0) \% 26$ ] = Q(16)  
Output:  $(16 - 0 + 0)\%26 = 16$  (Q)

REFLEKTOR: Q -> E

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = E(4)  
Process: WIRING[backward][ $(0 + 4 - 0) \% 26$ ] = A(0)  
Output:  $(0 - 0 + 0)\%26 = 0$  (A)

Rotor II:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = A(0)  
Process: WIRING[backward][ $(0 + 0 - 0) \% 26$ ] = A(0)  
Output:  $(0 - 0 + 0)\%26 = 0$  (A)

Rotor III:  
OFFSET: 21(V)  
RING\_OFFSET: 20  
input = A(0)  
Process: WIRING[backward][ $(21 + 0 - 20) \% 26$ ] = A(0)  
Output:  $(0 - 21 + 20)\%26 = 25$  (Z)

RESULT: Z

2. Tahap 2, Current Key: ABW (+ 1 dari initial key, Rotor tengah berputar karena transisi notch V->W pada ring 3)

INPUT: U  
PLUGBOARD: (U-> S) S

Rotor III:  
OFFSET: 22(W)  
RING\_OFFSET: 20  
input = S(18)  
Process: WIRING[forward][ $(22 + 18 - 20) \% 26$ ] = K(10)  
Output:  $(10 - 22 + 20)\%26 = 8$  (I)

Rotor II:  
OFFSET: 1(B)  
RING\_OFFSET: 0

input = I(8)  
Process: WIRING[forward][ $(1 + 8 - 0) \% 26$ ] = B(1)  
Output:  $(1 - 1 + 0)\%26 = 0$  (A)

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = A(0)  
Process: WIRING[forward][ $(0 + 0 - 0) \% 26$ ] = E(4)  
Output:  $(4 - 0 + 0)\%26 = 4$  (E)

REFLEKTOR: E->Q

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = Q(16)  
Process: WIRING[backward][ $(0 + 16 - 0) \% 26$ ] = H(7)  
Output:  $(7 - 0 + 0)\%26 = 7$  (H)

Rotor II:  
OFFSET: 1(B)  
RING\_OFFSET: 0  
input = H(7)  
Process: WIRING[backward][ $(1 + 7 - 0) \% 26$ ] = F(5)  
Output:  $(5 - 1 + 0)\%26 = 4$  (E)

Rotor III:  
OFFSET: 22(W)  
RING\_OFFSET: 20  
input = E(4)  
Process: WIRING[backward][ $(22 + 4 - 20) \% 26$ ] = S(18)  
Output:  $(18 - 22 + 20)\%26 = 16$  (Q)

RESULT: Q

### 3. Tahap 3, Current Key: ABX (+ 1)

INPUT: D

Rotor III:  
OFFSET: 23(X)  
RING\_OFFSET: 20  
input = D(3)  
Process: WIRING[forward][ $(23 + 3 - 20) \% 26$ ] = C(2)  
Output:  $(2 - 23 + 20)\%26 = 25$  (Z)

Rotor II:

OFFSET: 1(B)  
RING\_OFFSET: 0  
input = Z(25)  
Process: WIRING[forward][ $(1 + 25 - 0) \% 26$ ] = A(0)  
Output:  $(0 - 1 + 0)\%26 = 25$  (Z)

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = Z(25)  
Process: WIRING[forward][ $(0 + 25 - 0) \% 26$ ] = J(9)  
Output:  $(9 - 0 + 0)\%26 = 9$  (J)

REFLEKTOR: J->X

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = X(23)  
Process: WIRING[backward][ $(0 + 23 - 0) \% 26$ ] = Q(16)  
Output:  $(16 - 0 + 0)\%26 = 16$  (Q)

Rotor II:  
OFFSET: 1(B)  
RING\_OFFSET: 0  
input = Q(16)  
Process: WIRING[backward][ $(1 + 16 - 0) \% 26$ ] = G(6)  
Output:  $(6 - 1 + 0)\%26 = 5$  (F)

Rotor III:  
OFFSET: 23(X)  
RING\_OFFSET: 20  
input = F(5)  
Process: WIRING[backward][ $(23 + 5 - 20) \% 26$ ] = Q(16)  
Output:  $(16 - 23 + 20)\%26 = 13$  (N)  
RESULT: N

#### 4. Tahap 4, Current Key: ABY (+ 1)

INPUT: O

Rotor III:  
OFFSET: 24(Y)  
RING\_OFFSET: 20  
input = O(14)  
Process: WIRING[forward][ $(24 + 14 - 20) \% 26$ ] = G(6)  
Output:  $(6 - 24 + 20)\%26 = 2$  (C)

Rotor II:

OFFSET: 1(B)  
RING\_OFFSET: 0  
input = C(2)  
Process: WIRING[forward][ $(1 + 2 - 0) \% 26$ ] = K(10)  
Output:  $(10 - 1 + 0)\%26 = 9$  (J)

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = J(9)  
Process: WIRING[forward][ $(0 + 9 - 0) \% 26$ ] = Z(25)  
Output:  $(25 - 0 + 0)\%26 = 25$  (Z)

REFLEKTOR: Z->T

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = T(19)  
Process: WIRING[backward][ $(0 + 19 - 0) \% 26$ ] = L(11)  
Output:  $(11 - 0 + 0)\%26 = 11$  (L)

Rotor II:  
OFFSET: 1(B)  
RING\_OFFSET: 0  
input = L(11)  
Process: WIRING[backward][ $(1 + 11 - 0) \% 26$ ] = O(14)  
Output:  $(14 - 1 + 0)\%26 = 13$  (N)

Rotor III:  
OFFSET: 24(Y)  
RING\_OFFSET: 20  
input = N(13)  
Process: WIRING[backward][ $(24 + 13 - 20) \% 26$ ] = I(8)  
Output:  $(8 - 24 + 20)\%26 = 4$  (E)

RESULT: E

**Encrypted Text: ZQNE**

## 4. Contoh step by step dekripsi Enigma

Config: sama seperti step 3  
Teks: ZQNE

### 1. Tahap 1, Current Key: AAV (+1)

INPUT: Z  
Rotor III:

OFFSET: 21(V)  
RING\_OFFSET: 20  
input = Z(25)  
Process: WIRING[forward][ $(21 + 25 - 20) \% 26$ ] = B(1)  
Output:  $(1 - 21 + 20)\%26 = 0$  (A)

Rotor II:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = A(0)  
Process: WIRING[forward][ $(0 + 0 - 0) \% 26$ ] = A(0)  
Output:  $(0 - 0 + 0)\%26 = 0$  (A)

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = A(0)  
Process: WIRING[forward][ $(0 + 0 - 0) \% 26$ ] = E(4)  
Output:  $(4 - 0 + 0)\%26 = 4$  (E)

REFLEKTOR: E->Q

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = Q(16)  
Process: WIRING[backward][ $(0 + 16 - 0) \% 26$ ] = H(7)  
Output:  $(7 - 0 + 0)\%26 = 7$  (H)

Rotor II:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = H(7)  
Process: WIRING[backward][ $(0 + 7 - 0) \% 26$ ] = L(11)  
Output:  $(11 - 0 + 0)\%26 = 11$  (L)

Rotor III:  
OFFSET: 21(V)  
RING\_OFFSET: 20  
input = L(11)  
Process: WIRING[backward][ $(21 + 11 - 20) \% 26$ ] = V(21)  
Output:  $(21 - 21 + 20)\%26 = 20$  (U)

RESULT: U  
PLUGBOARD: U->S



## 2. Tahap 2, Current Key: ABW (+1, Rotor tengah berputar karena transisi V->W ring 3)

INPUT: Q

Rotor III:

OFFSET: 22(W)

RING\_OFFSET: 20

input = Q(16)

Process: WIRING[forward][(22 + 16 - 20) % 26] = G(6)

Output: (6 - 22 + 20)%26 = 4 (E)

Rotor II:

OFFSET: 1(B)

RING\_OFFSET: 0

input = E(4)

Process: WIRING[forward][(1 + 4 - 0) % 26] = I(8)

Output: (8 - 1 + 0)%26 = 7 (H)

Rotor I:

OFFSET: 0(A)

RING\_OFFSET: 0

input = H(7)

Process: WIRING[forward][(0 + 7 - 0) % 26] = Q(16)

Output: (16 - 0 + 0)%26 = 16 (Q)

REFLEKTOR: Q->E

Rotor I:

OFFSET: 0(A)

RING\_OFFSET: 0

input = E(4)

Process: WIRING[backward][(0 + 4 - 0) % 26] = A(0)

Output: (0 - 0 + 0)%26 = 0 (A)

Rotor II:

OFFSET: 1(B)

RING\_OFFSET: 0

input = A(0)

Process: WIRING[backward][(1 + 0 - 0) % 26] = J(9)

Output: (9 - 1 + 0)%26 = 8 (I)

Rotor III:

OFFSET: 22(W)

RING\_OFFSET: 20

input = I(8)

Process: WIRING[backward][(22 + 8 - 20) % 26] = U(20)

Output: (20 - 22 + 20)%26 = 18 (S)

RESULT: S  
PLUGBOARD: U

### 3. Tahap 3, Current Key: ABX (+1)

INPUT: N

Rotor III:  
OFFSET: 23(X)  
RING\_OFFSET: 20  
input = N(13)  
Process: WIRING[forward][(23 + 13 - 20) % 26] = I(8)  
Output: (8 - 23 + 20)%26 = 5 (F)

Rotor II:  
OFFSET: 1(B)  
RING\_OFFSET: 0  
input = F(5)  
Process: WIRING[forward][(1 + 5 - 0) % 26] = R(17)  
Output: (17 - 1 + 0)%26 = 16 (Q)

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = Q(16)  
Process: WIRING[forward][(0 + 16 - 0) % 26] = X(23)  
Output: (23 - 0 + 0)%26 = 23 (X)

REFLEKTOR: J

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = J(9)  
Process: WIRING[backward][(0 + 9 - 0) % 26] = Z(25)  
Output: (25 - 0 + 0)%26 = 25 (Z)

Rotor II:  
OFFSET: 1(B)  
RING\_OFFSET: 0  
input = Z(25)  
Process: WIRING[backward][(1 + 25 - 0) % 26] = A(0)  
Output: (0 - 1 + 0)%26 = 25 (Z)

Rotor III:  
OFFSET: 23(X)  
RING\_OFFSET: 20  
input = Z(25)

Process:  $\text{WIRING}[\text{backward}][[(23 + 25 - 20) \% 26]] = \text{G}(6)$   
Output:  $(6 - 23 + 20) \% 26 = 3 \text{ (D)}$   
RESULT: D

#### 4. Tahap 4, Current Key: ABY (+1)

INPUT: E  
Rotor III:  
OFFSET: 24(Y)  
RING\_OFFSET: 20  
input = E(4)  
Process:  $\text{WIRING}[\text{forward}][[(24 + 4 - 20) \% 26]] = \text{R}(17)$   
Output:  $(17 - 24 + 20) \% 26 = 13 \text{ (N)}$

Rotor II:  
OFFSET: 1(B)  
RING\_OFFSET: 0  
input = N(13)  
Process:  $\text{WIRING}[\text{forward}][[(1 + 13 - 0) \% 26]] = \text{M}(12)$   
Output:  $(12 - 1 + 0) \% 26 = 11 \text{ (L)}$

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = L(11)  
Process:  $\text{WIRING}[\text{forward}][[(0 + 11 - 0) \% 26]] = \text{T}(19)$   
Output:  $(19 - 0 + 0) \% 26 = 19 \text{ (T)}$

REFLEKTOR: T->Z

Rotor I:  
OFFSET: 0(A)  
RING\_OFFSET: 0  
input = Z(25)  
Process:  $\text{WIRING}[\text{backward}][[(0 + 25 - 0) \% 26]] = \text{J}(9)$   
Output:  $(9 - 0 + 0) \% 26 = 9 \text{ (J)}$

Rotor II:  
OFFSET: 1(B)  
RING\_OFFSET: 0  
input = J(9)  
Process:  $\text{WIRING}[\text{backward}][[(1 + 9 - 0) \% 26]] = \text{D}(3)$   
Output:  $(3 - 1 + 0) \% 26 = 2 \text{ (C)}$

Rotor III:  
OFFSET: 24(Y)  
RING\_OFFSET: 20  
input = C(2)  
Process:  $\text{WIRING}[\text{backward}][[(24 + 2 - 20) \% 26]] = \text{S}(18)$

Output:  $(18 - 24 + 20) \% 26 = 14$  (O)

RESULT: O

Decrypted Text: SUDO

## 5. Screenshot hasil program dibandingkan dengan enigma di internet

Plug boards:

- S->U

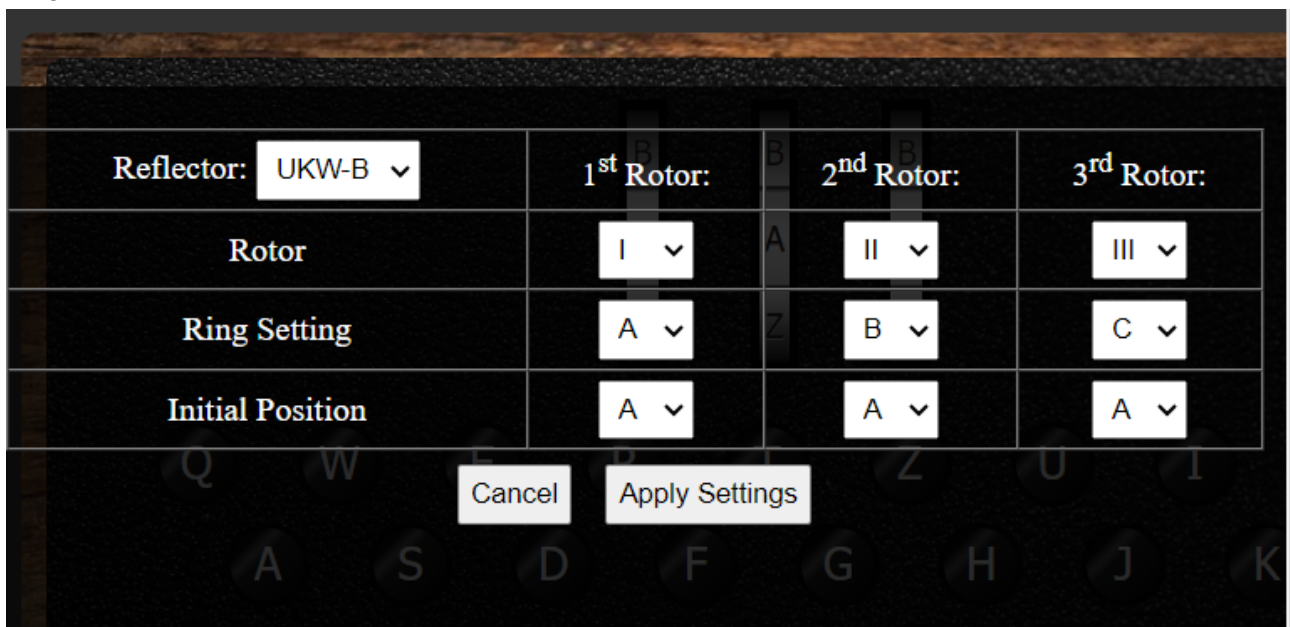
- D->O

Rotor Position:

I-II-III

Rotor initial position: AAA

Ring offset: ABC



N o	Teks	Hasil Web	Hasil Program
1	sudo mau libur		
2	BPRQ ZRW HLPBW		

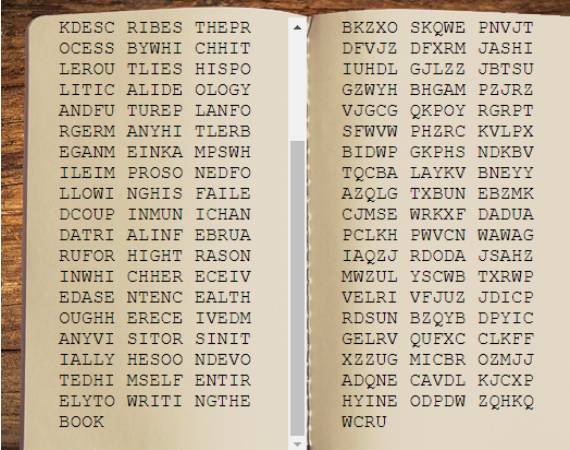
MEINK AMPFI  
SANAU TOBIO  
GRAPH YBYNA  
ZIPAR TYLEA  
DERAD ULFHI  
TLERT HEWOR  
KDESC RIBES  
THEPR OCESS  
BYWHI CHHIT  
LEROU TLIES  
HISPO LITIC ALIDE  
OLOGY ANDFU  
TUREP LANFO  
RGERM ANYHI  
TLERB EGANM  
EINKA MPSWH  
ILEIM PROSO  
NEDFO LLOWI  
NGHIS FAILE  
DCOUP INMUN  
ICHAN DATRI  
ALINF EBRUA  
RUFOR HIGHT  
RASON INWHI  
CHHER ECEIV  
EDASE NTENC  
EALTH OUGHH  
ERECE IVEDM  
ANYVI SITOR SINIT  
IALLY HESOO  
NDEVO TEDHI  
MSELF ENTIR  
ELYTO WRITI  
NGTHE BOOK

Plaintext:

MEINK AMPFI SANAU  
TOBIO GRAPH YBYNA  
ZIPAR TYLEA DERAD  
ULFHI TLERT HEWOR  
KDESC RIBES THEPR  
OCESS BYWHI CHHIT  
LEROU TLIES HISPO  
LITIC ALIDE OLOGY  
ANDFU TUREP LANFO  
RGERM ANYHI TLERB  
EGANM EINKA MPSWH  
ILEIM PROSO NEDFO  
LLOWI NGHIS FAILE  
DCOUP INMUN ICHAN  
DATRI ALINF EBRUA  
RUFOR HIGHT RASON  
INWHI CHHER ECEIV  
EDASE NTENC EALTH

Ciphertext:

XIVZW RSDMX IGYIG  
KZGHA DVMBF MCXYO  
AQJWN RGFNU PBGCZ  
NMGZU EBCJG AXLWJ  
BKZXO SKQWE PNVJT  
DFVJZ DFXRM JASHI  
IUHDL GJLZZ JBTSU  
GZWYH BHGAM PZJRZ  
VJGCG QKPOY RGRPT  
SFWVW PHZRC KVLPH  
BIDWP GKPHS NDKBV  
TQCBA LAYKV BNEY Y  
AZQLG TXBUN EBZMK  
CJMSE WRKXF DADUA  
PCLKH PWVCN WAWAG  
IAQZJ RDODA JSAHZ  
MWZUL YSCWB TXRWP  
VELRI VFJUZ JDICP



MEINK AMPFI SANAU TOBIO GRAPH YBYNA ZIPAR TYLEA DERAD ULFHI TLERT HEWOR KDESC RIBES THEPR OCESS BYWHI CHHIT LEROU TLIES  
HISPO LITIC ALIDE OLOGY ANDFU TUREP LANFO RGERM ANYHI TLERB EGANM EINKA MPSWH ILEIM PROSO NEDFO LLOWI NGHIS FAILE DCOUP  
INMUN ICHAN DATRI ALINF EBRUA RUFOR HIGHT RASON INWHI CHHER ECEIV EDASE NTENC EALTH OUGHH ERECE IVEDM ANYVI SITOR SINIT  
IALLY HESOO NDEVO TEDHI MSELF ENTIR ELYTO WRITI NGTHE BOOK  
Encrypted Text: XIVZW RSDMX IGYIG KZGHA DVMBF MCXYO AQJWN RGFNU PBGCZ NMGZU EBCJG AXLWJ BKZXO SKQWE PNVJT DFVJZ DFXRM JASHI  
IUHDL GJLZZ JBTSU GZWYH BHGAM PZJRZ VJGCG QKPOY RGRPT SFWVW PHZRC KVLPH BIDWP GKPHS NDKBV TQCBA LAYKV BNEY Y AZQLG TXBUN  
EBZMK CJMSE WRKXF DADUA PCLKH PWVCN WAWAG IAQZJ RDODA JSAHZ MWZUL YSCWB TXRWP VELRI VFJUZ JDICP RDSUN BZQYB DPYIC GELRV  
QUFXC CLKFF XZZUG MICBR OZMJJ ADQNE CAVDL KJCKP HYINE ODPDW ZQHKQ WCRU

4

XIVZW  
RSDMX IGYIG  
KZGHA  
DVMBF  
MCXYO  
AQJWN  
RGPNU  
PBGCZ  
NMGZU  
EBCJG  
AXLWJ  
BKZXO  
SKQWE  
PNVJT DFBVJZ  
DFXRM JASHI  
IUHDL GJLZZ  
JBTSU  
GZWHY  
BHGAM  
PZJRZ  
VJGCG  
QKPOY  
RGRPT  
SFWVW  
PHZRC  
KVLXP  
BIDWP  
GKPHS  
NDKBV  
TQCBA  
LAYKV  
BNEYV  
AZQLG  
TXBUN  
EBZMK  
CJMSE  
WRKXF  
DADUA  
PCLKH  
PWVCN  
WAWAG  
IAQZJ RDODA  
JSAHZ  
MWZUL  
YSCWB  
TXRWP  
VELRI VFJUZ  
JDICP RDSUN  
BZQYB DPYIC  
GELRV  
QUFXC  
CLKFF  
XZZUG  
MICBR OZMJJ  
ADQNE  
CAVDL

<u>Ciphertext:</u>	<u>Plaintext:</u>
XIVZW RSDMX IGYIG	MEINK AMPFO SANAU
KZGHA DVMBF MCXYO	TOBIO GRAPH YBYNA
AQJWN RGPNU PBGVZ	ZIPAR TYLEA DERDD
NMGZU	ULFHI

(... mager 😊)

XIVZW RSDMX IGYIG KZGHA DVMBF MCXYO AQJWN RGPNU PBGCZ NMGZU EBCJG AXLWJ BKZXO SKQWE PNVJT DFBVJZ DFXRM JASHI IUHDL GJLZZ  
JBTSU GZWHY BHGAM PZJRZ VJGCG QKPOY RGRPT SFWVW PHZRC KVLXP BIDWP GKPHS NDKBV TQCBA LAYKV BNEYV AZQLG TXBUN EBZMK CJMSE  
WRKXF DADUA PCLKH PWVCN WAWAG IAQZJ RDODA JSAHZ MWZUL YSCWB TXRWP VELRI VFJUZ JDICP RDSUN BZQYB DPYIC GELRV QUFXC CLKFF  
XZZUG MICBR OZMJJ ADQNE CAVDL KJCPX HYINE ODPDW ZQHKQ WCRU  
Decrypted Text: MEINK AMPFI SANAU TOBIO GRAPH YBYNA ZIPAR TYLEA DERAD ULFHI TLERT HEMOR KDESC RIBES THEPR OCESS BYWHI CHHIT  
LEROU TLIES HISPO LITIC ALIDE OLOGY ANDFU TUREP LANFO RGERM ANYHI TLERB EGAMM EINKA MPSWH ILEIM PROSO NEDFO LLOWI NGHIS  
FAILE DCOUP IMMUN ICHAN DATRI ALINF EBRUA RUFOR HIGHT RASON INWHI CHHER ECEIV EDASE NTENC EALTH OUGHH ERECE IVEDM ANVVI  
SITOR SINIT IALLY HESOO NDEVO TEDHI MSELF ENTIR ELYTO WRITTI NGTHE BOOK

	KJCXP HYINE ODPDW ZQHKQ WCRU		
--	---------------------------------------	--	--