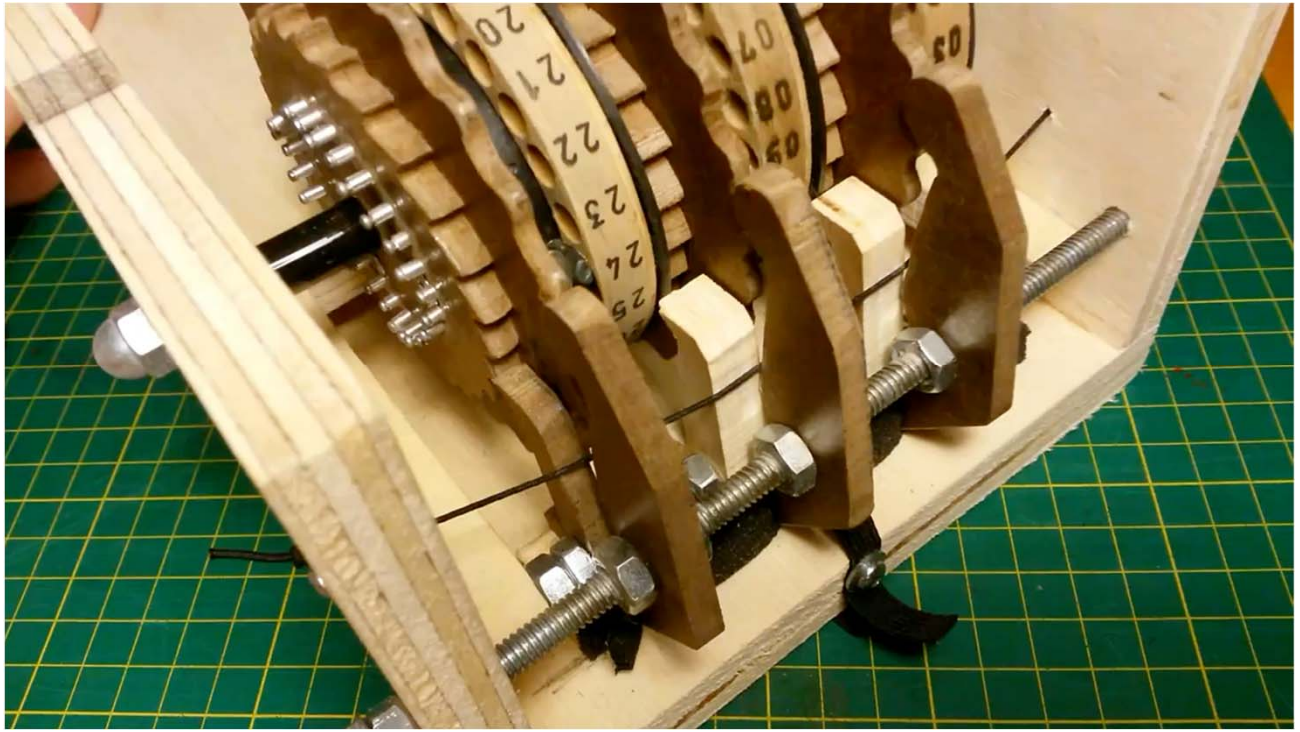
Enigma

Rotor machine

- 在现代密码之前，Rotor machine是实际使用中最常见的复杂替换密码。
- 广泛用于二战。
  - German Enigma, Allied Hagelin, Japanese Purple
- 密码机使用了一系列转子（rotor），每个rotor完成一次替换，在对字母加密后，rotor旋转并改变。

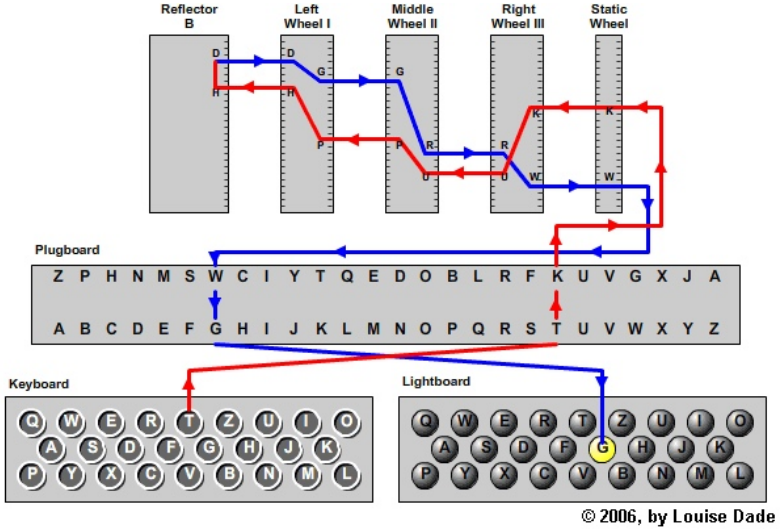
Enigma

- Enigma密码机是一种用于加密与解密文件的Rotor machine，在二战时期广泛使用。



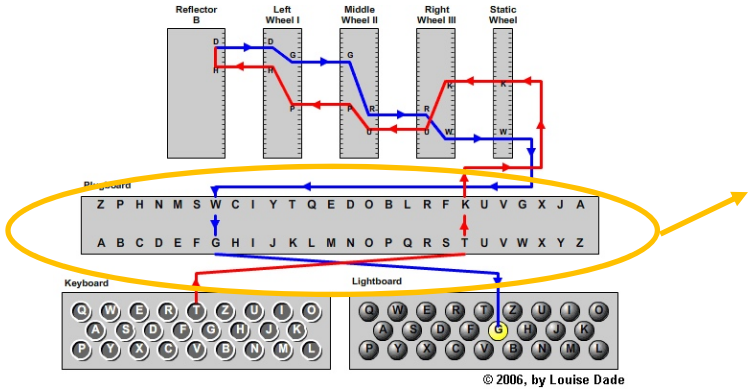
Enigma

- Enigma密码机是一种用于加密与解密文件的Rotor machine，在二战时期广泛使用。



Enigma - 接线板（plugboard）

- 接线板上的每条线都会连接一对字母。这些线的作用就是在电流进入转子前改变它的方向。例如如图所示，将A插口和J插口连接起来。当操作员按下A键时，电流就会先流到J插口（相当于按下J键）再流经转子。



- 接线板只在从右到左进入最右侧那个齿轮以及从左到右从最右侧齿轮出来亮灯的时候才起作用。

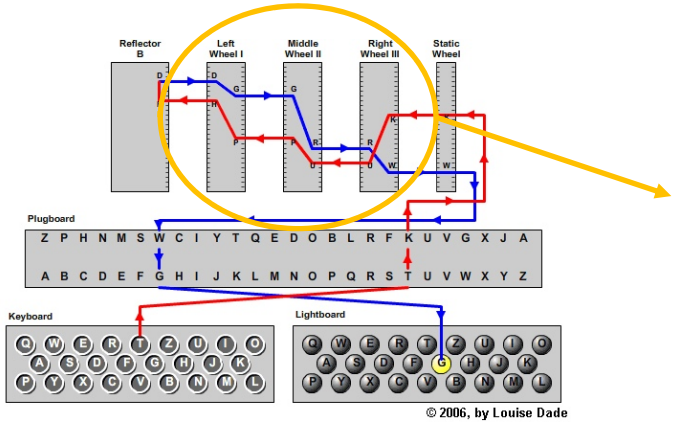
Enigma - 接线板（plugboard）

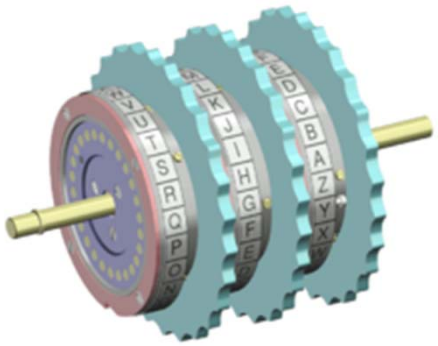
- 编程举例：
  - 假定接线板设置为:A-B, C-D，其他不变
- char plug[27] ="BADCEFGHIJKLMNOPQRSTUVWXYZ";
  /*ABCDEFGHIJKLMNOPQRSTUVWXYZ*/
- char c='A', e;
- e = plug[c-'A']; // e='B';
- A → B

- 注意接线板查表既可以正向查,也可以反向查,结果是一样的。

Enigma - 转子（Rotor）

- 密码机的核心部分。
- 转子的关键部件：
  - Ring setting
  - Message key
  - Rotor本身





Enigma - Ring Setting

- 操作前所需的初始设置的一部分，即现代术语中初始化向量（Initialization vector）的一部分。

- Ring Setting是齿轮（转子）内部的初始状态，它们在齿轮转动时不会发生变化。

Enigma - Message Key

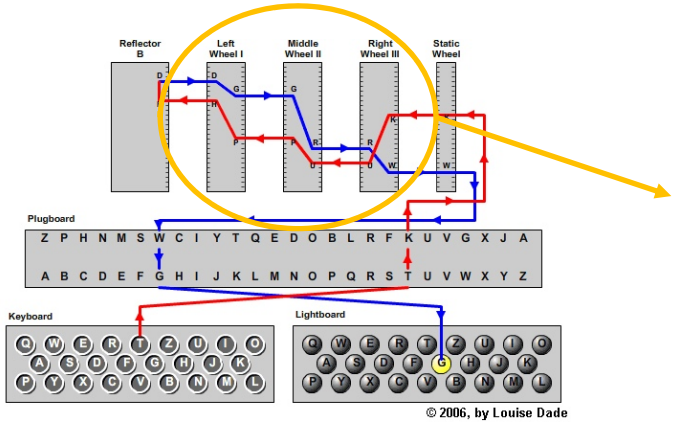
- Message Key是齿轮外部的状态。
- 会随每一次按键而发生变化。
- 如何把Message Key传递给对方?
  - 发送方随机想出3个齿轮的外部状态(Message Key)为: ABC
  - 以明文的形式把ABC发送给对方;
  - 再想出今天要用到密钥即真正用来加密的齿轮初始状态为: ZJU
  - 在当前齿轮初始状态为ABC的情况下，连续按下ZJU得到ZJU的密文设为Z'J'U'发送给对方。
  - 对方在齿轮初始状态为ABC的情况下，输入Z'J'U'一定可以解密出ZJU。
  - 双方都把齿轮外部的初始状态设为ZJU，然后就可以开始正式通讯。

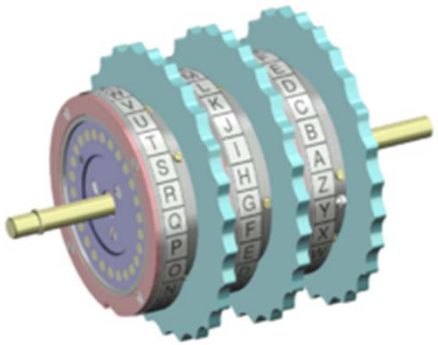
Enigma - 齿轮加密预处理

- 假定I齿轮的RingSetting=B(内部), MessageKey=A(外部)
- 现在按键盘A的时候, A进入I齿轮后, 要做以下运算:
- char c = 'A';
- int delta = MessageKey - RingSetting;
- c = ((c-'A')+delta+26)%26 + 'A';
- 此时c='Z';
- 然后拿c去查齿轮I的表进行加密，出齿轮时还需减去delta。
- 所有齿轮都是一样的步骤:
  - 从左到右进入要加delta，右边出来要减delta。

Enigma - 转子（Rotor）

- 每次按键，转子都会发生转动，保证了每次按相同的键得到的结果不一样。
- 注意, 当按下某个键时, 对该键进行加密的密钥并非当前齿轮的状态，而是齿轮转了一下以后的状态。例如: 设3个齿轮从左到右分别为III, II, I, 并且齿轮的外部状态（message key）从左到右为AAZ, 则输入字母A时，齿轮转到AAA的位置，此时AAA就是密钥。





Enigma - 单个齿轮加密示例

- 设I号齿轮的RingSetting（内部）=B, MessageKey（外部）=D, 则
- Δ=MessageKey-RingSetting='D'-'B'=2
- 现在假定输入字母A，则A进入I号齿轮时，需要先加上Δ即变成A+2=‘C’
- 设rotor I:
  - char rotor[27]="EKMFLGDQVZNTOWYHXUSPAIBRCJ";
  - // ABCDEFGHIJKLMNOPQRSTUVWXYZ
- char c='C', e;
- e = rotor[c-'A']; // e='M';
- 查表:'C'→'M', 而从I号齿轮出去时要减去Δ，即'M'-2='K'。

Enigma - 多个齿轮加密示例

- RingSetting: III=A, II=A, I=B
- MessageKey: III=A, II=A, I=C
- Enigma是先转动齿轮再查表的，敲键A后MessageKey变成: III=A, II=A, I=D
- Δ=MessageKey-RingSetting='D'-'B'=2
- 从II号齿轮返回到I号齿轮时，进入I号齿轮的字母为G
- 此时G+Δ=G+2=I
- rotor I:
  - char rotor[27]="EKMFLGDQVZNTOWYHXUSPAIBRCJ";
  - // ABCDEFGHIJKLMNOPQRSTUVWXYZ
- I反查表得V, V-Δ=V-2=T, 最终T就是密文。

Enigma - 转子（Rotor）转动

- 卡口：当前齿轮处于卡口时，会带动更高位的齿轮旋转。
- 5个齿轮使下一个齿轮发生跳转的字母:
- QEVJZ; 齿轮的当前位置,从左到右对应齿轮I II III IV V
- RFWKA; 齿轮的下一步位置
  - Royal Flags Wave Kings Above
- 假定3个齿轮为III、II、I, 齿轮I的当前位置=Q且II的当前位置=A, 现在敲任何一个键，都会使齿轮I转到R这个位置, 此时I会带动II旋转, 于是II会从A转到B。

Enigma - 转子（Rotor）- double stepping

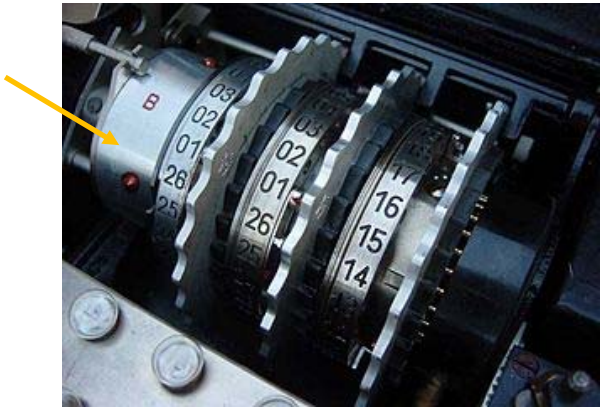
- 假定III=1=A, II=4=D, I=17=Q
- 现在I旋转，从Q变成R，一定会带动II旋转:
- III=1=A, II=5=E, I=18=R
- 此时再旋转I的话，I本来是不应该带动II转的(因为当前I不在Q这个位置), 但是II还会再转(double stepping)，
- 同时II带动III旋转:
- III=2=B, II=6=F, I=19=S

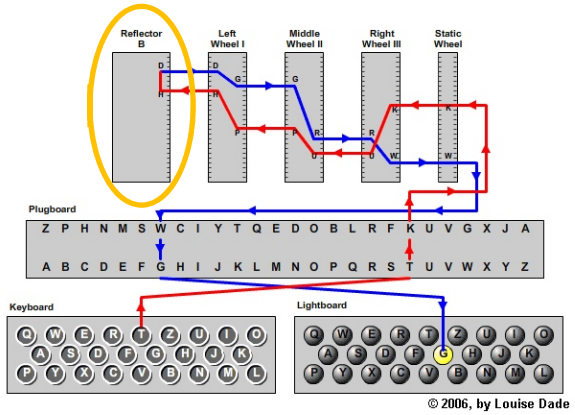
Enigma - 转子（Rotor）- double stepping

- 归纳起来讲, II有两种情况会转动:
- (1) I从Q转到R
- (2) II当前在E位置, I不管在什么位置
- double stepping是由Enigma的机械结构决定的，该现象只会出现在中间那个齿轮上。

Enigma - 反射器（reflector）

- Enigma密码机的最后一个转子之后有一个反射器，将最后一个转子的其中两个触点连接起来，并将电流沿一个不同的路线导回。
- 反射板上的映射是两两交换的，这使得加密过程与解密过程变得一致。
- 反射器使Enigma具有了如下性质：加密后得到的字母与输入的字母永远不会相同。

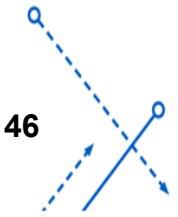




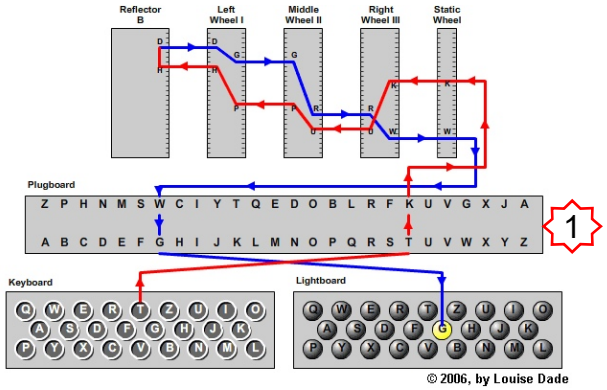
Enigma - 反射器（reflector）

- 编码举例：
- char reflector[27]=" YRUHQSLDPXNGOKMIEBFZCWVJAT";
- // ABCDEFGHIJKLMNOPQRSTUVWXYZ

- V→W
- W→V
- Y→A
- A→Y



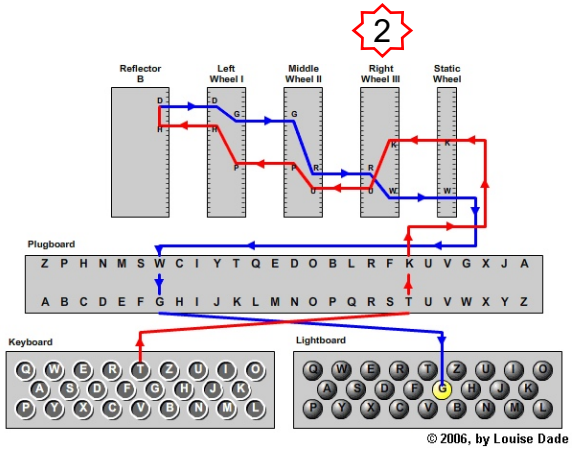
Enigma – 加密示例



- 假设明文为 A ，经历三个齿轮，delta = 0
- 第一步：接线板plugboard
- 假定接线板设置为:A-B, C-D
- char plug[27] ="BADCEFGHIJKLMNOPQRSTUVWXYZ";
  /*ABCDEFGHIJKLMNOPQRSTUVWXYZ*/
- char c='A', e;
- e = plug[c-'A']; // e='B';
- A → B

Enigma – 加密示例

- 第二步： 齿轮I rotor I
- char rotor[27]="EKMFLGDQVZNTOWYHXUSPAIBRCJ";
- // ABCDEFGHIJKLMNOPQRSTUVWXYZ
- char c='B', e;
- e = rotor[c-'A']; // e='K';
- B→K

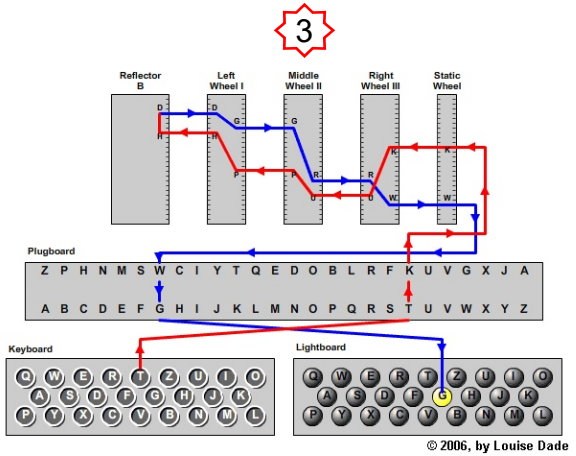


Enigma – 加密示例

- 第三步： 齿轮II rotor II

- AJDKSIRUXBLHWTMCQGZNPYFVOE
- ABCDEFGHIJKLMNOPQRSTUVWXYZ

- K→L

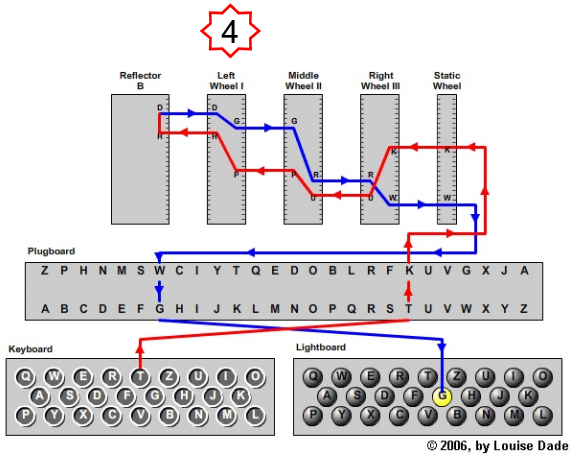


Enigma – 加密示例

- 第四步： 齿轮III rotor III

- BDFHJLCPRTXVZNYEIWGAKMUSQO
- ABCDEFGHIJKLMNOPQRSTUVWXYZ

- L→V

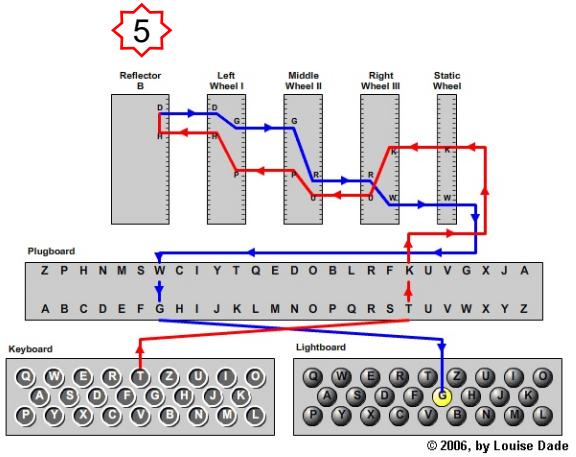


Enigma – 加密示例

- 第五步： 反射器 reflector

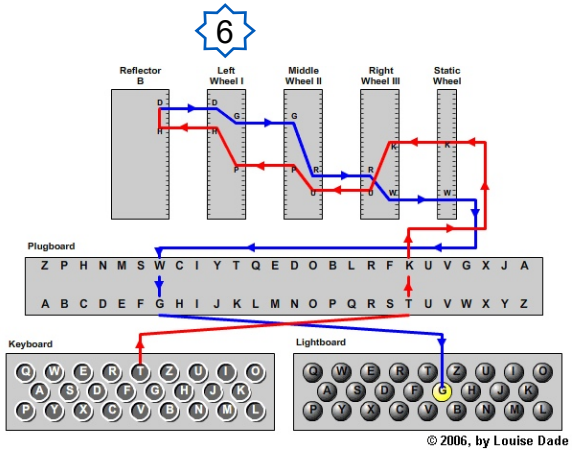
- char reflector[27]=" YRUHQSLDPXNGOKMIEBFZCWVJAT";
- // ABCDEFGHIJKLMNOPQRSTUVWXYZ

- V→W



Enigma – 加密示例

- 经过上述5步, A转成W, 但W还不是密文, 还要走一条逆向路径:4-3-2-1, 把W进一步转化成密文。
- 第六步： 齿轮III rotor III
- char rotor[27]="BDFHJLCPRTXVZNYEIWGAKMUSQO";
- // ABCDEFGHIJKLMNOPQRSTUVWXYZ



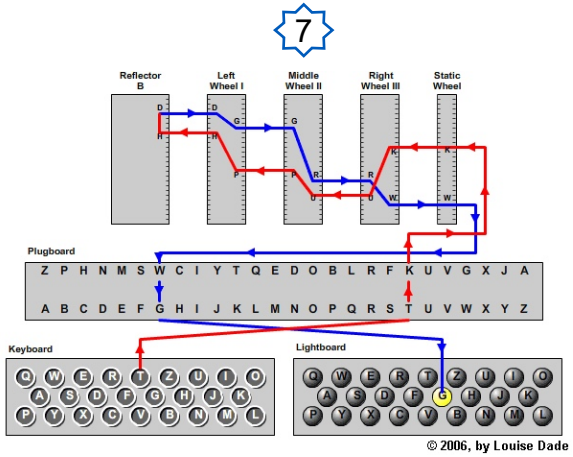
- W→R
- 注意按逆向路径经过3个齿轮时要反查表, 即在数组rotor中寻找元素'W', 得到该元素的下标设为i, 再把i+'A',就得到'R'。

Enigma – 加密示例

- 第七步： 齿轮II rotor II

- AJDKSIRUXBLHWTMCQGZNPYFVOE
- ABCDEFGHIJKLMNOPQRSTUVWXYZ

- R→G

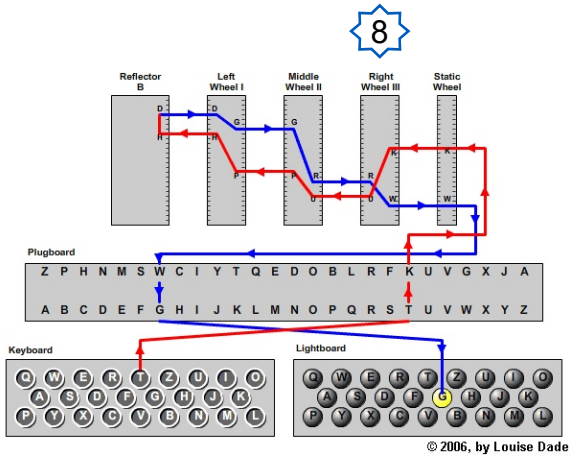


Enigma – 加密示例

- 第八步： 齿轮I rotor I

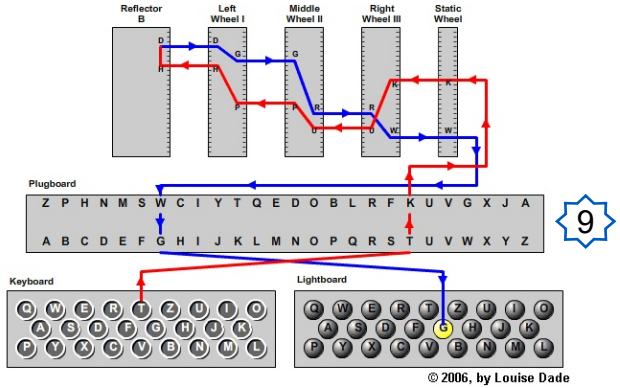
- EKMFLGDQVZNTOWYHXUSPAIBRCJ
- ABCDEFGHIJKLMNOPQRSTUVWXYZ

- G→F



Enigma – 加密示例

- 第九步：接线板plugboard
- char plug[27] ="BADCEFGHIJKLMNOPQRSTUVWXYZ";
  /*ABCDEFGHIJKLMNOPQRSTUVWXYZ*/



- F→F
- W转成F, 其中F就是密文。
- 解密时，尝试把F当作明文重新把前面的9步走一遍,最后出来的是A。

Enigma

- 齿轮IV、齿轮V在示例中没有用到，转换表如下。
- 齿轮IV:
- ESOVPZJAYQUIRHXLNFTGKDCMWB
- ABCDEFGHIJKLMNOPQRSTUVWXYZ

- 齿轮V:
- VZBRGITYUPSDNHLXAWMJQOFECK
- ABCDEFGHIJKLMNOPQRSTUVWXYZ

Enigma安全性

- ABCD中抽取2对字母的组合如下:
- AB CD; AC BD; AD BC

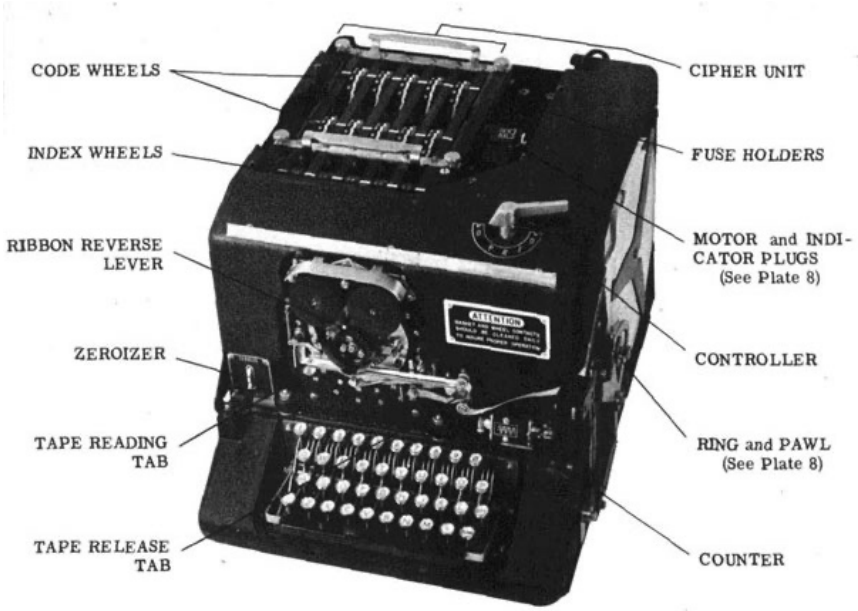
- 10对字母在接线板上的组合 = $C_{26}^{20} * (A_{20}^{10} / 2^{10})$

- 密码机的设置方法:
- $A_5^3 * 26^3 * (C_{26}^{20} * A_{20}^{10} / 2^{10} + C_{26}^{18} * A_{18}^{9} / 2^{9} + \ldots C_{26}^{2} * A_2^1 / 2^1 + 1)$

- 设置方法高达 $1.5*10^{19}$ 种

其他Rotor Machine

Typex



Sigaba

Thank you!