Turing and the Enigma Cipher

Dr Ron Steinfeld

ARC Research Fellow (Cryptography),

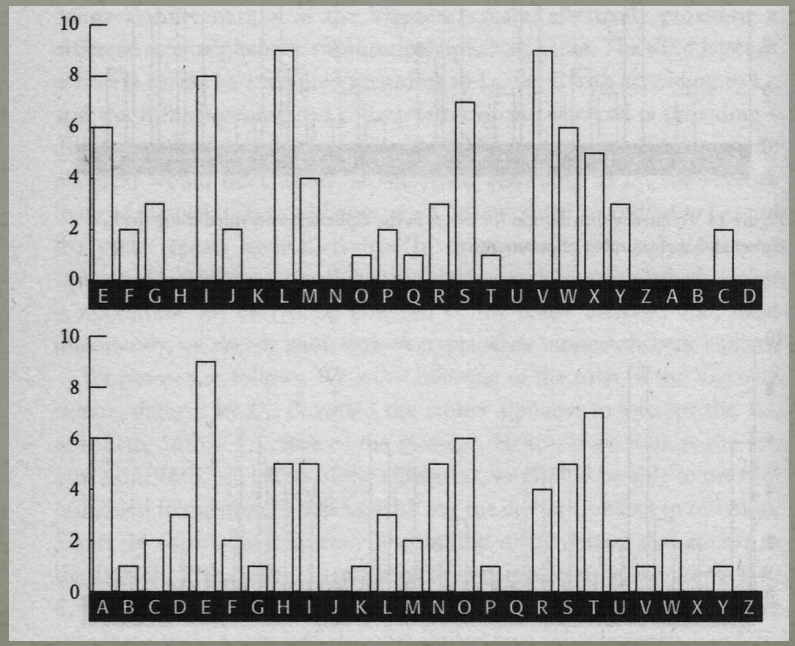
Clayton School of IT,

Monash University

A Simple Encryption Method

Keyword	K I N G K I N G K I N G K I N G K I N G K I N G
Plaintext	t h e s u n a n d t h e m a n i n t h e m o o n
Ciphertext	D P R Y E V N T N B U K W I A O X B U K W W B T

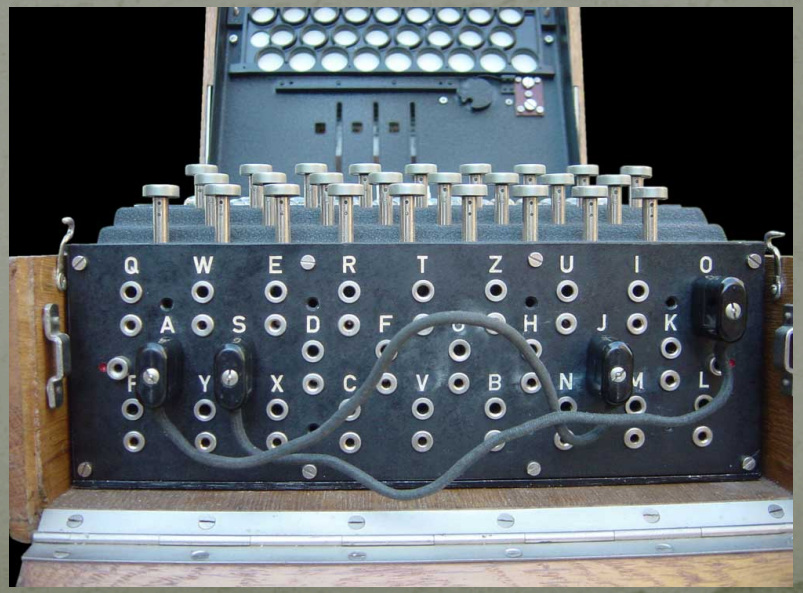
Can be broken by a simple statistical method:



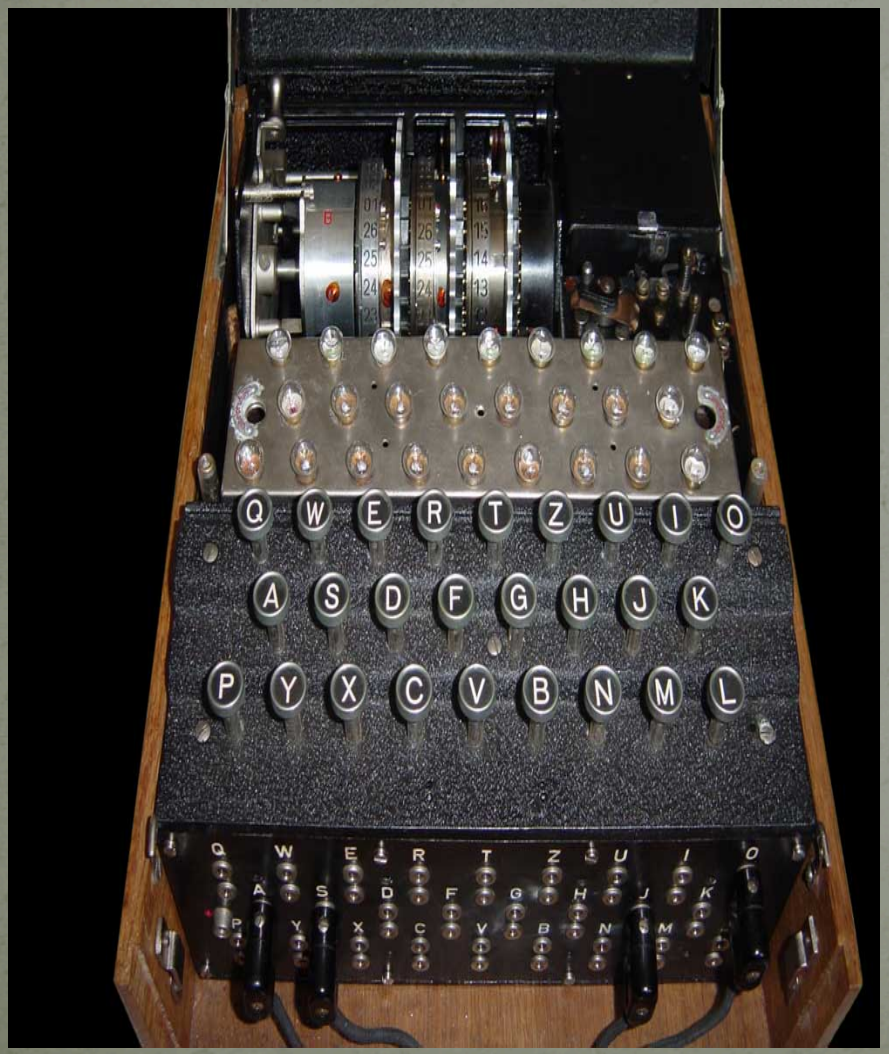
A Simple Encryption Method

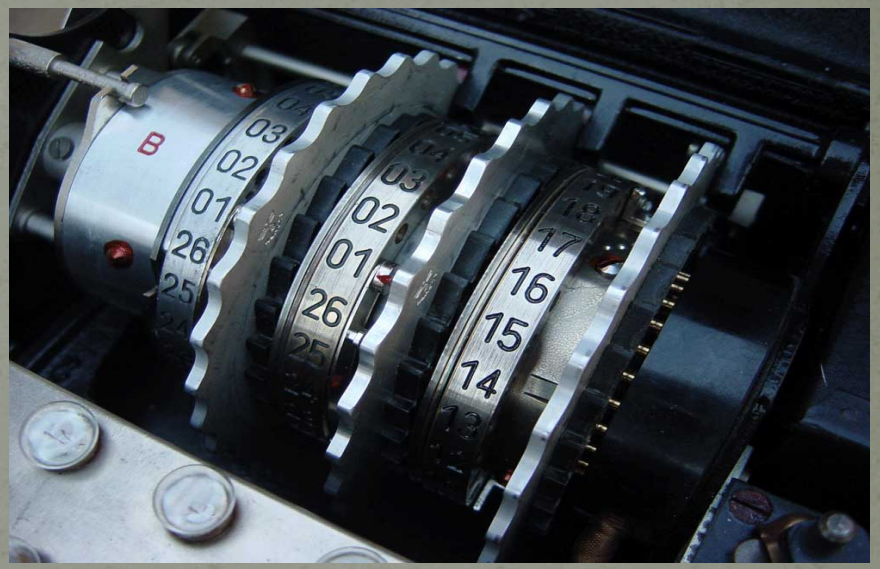
- What is the main weakness?
  - A short repeat period for key allows statistical attack
- How can it be improved?
  - Using a very long key period…
- But how to do it in practice?
  - Enigma provided a powerful solution!

Enigma: History & Construction

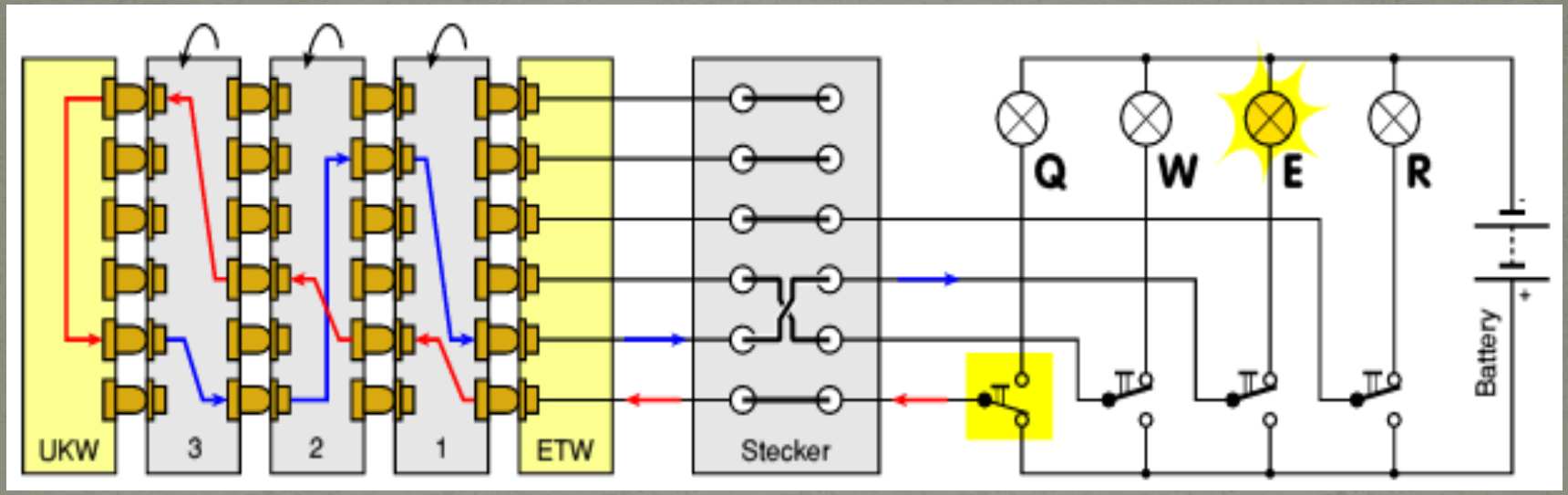


Enigma: Internal Construction





Enigma: Operation



Why was breaking Enigma a challenge?

- Without key settings, cryptanalyst can try **brute force attack**
- But, in Enigma, number of possible keys is enormous e.g. for a typical mid-war army Enigma (ignoring ring settings):
  - 3 Rotor orientations
    - A = 26 x 26 x 26 = 17,576 settings
  - 3 of 5 rotor choices/arrangements:
    - B = 5 x 4 x 3 = 60 settings
  - 10 plugboard cables (swap any 10 pairs of letters)
    - C = 150,738,274,937,250 settings!!!
  - Total # Keys = A x B x C~ 158,962,555,217,826,360,000 ~ 10^{20}

  (Search time : hundreds of years even on **today**'s powerful CPUs!)
- Only hope: find a shortcut attack
  - Avoid checking all possible guessed combinations
  - E.g. guessing and checking only part of key at a time, e.g. guess and check separately for rotor settings only (A X B X C → A x B)?
- But seemed hard due to "entanglement" effect of changing rotors and plugboard settings
- Moreover, even Enigma construction / rotor wiring was a secret!

Breaking Enigma: Historical background

- 1931: Enigma design details sold to French intelligence

- Mid 1930's: Polish cryptanalysis
  - Built an Enigma machine replicate
  - Brilliant attack relied on a flawed enciphering procedure used by German military in the 1930's (changed in May 1940)
  - Implemented electromechanical computers ("bombes") to automate attack procedure

Turing and Enigma: Historical Timeline

- Sep. 1938: Alan Turing begins working part-time for GCCS on Enigma
- Jul-Aug. 1939: Polish cryptanalysis results smuggled to London
- Sep. 1939: War breaks, Turing enlisted to Bletchley Park
- Early 1940: Turing finalizes his design for a bombe to implement his known plaintext ("crib")-based attack
- May 1940: German Military fixes flaw in enciphering procedure: Polish attack fails and British decryptions drop dramatically
- Aug. 1940 - 1945: New Bombes arrives, improving decryption rate dramatically

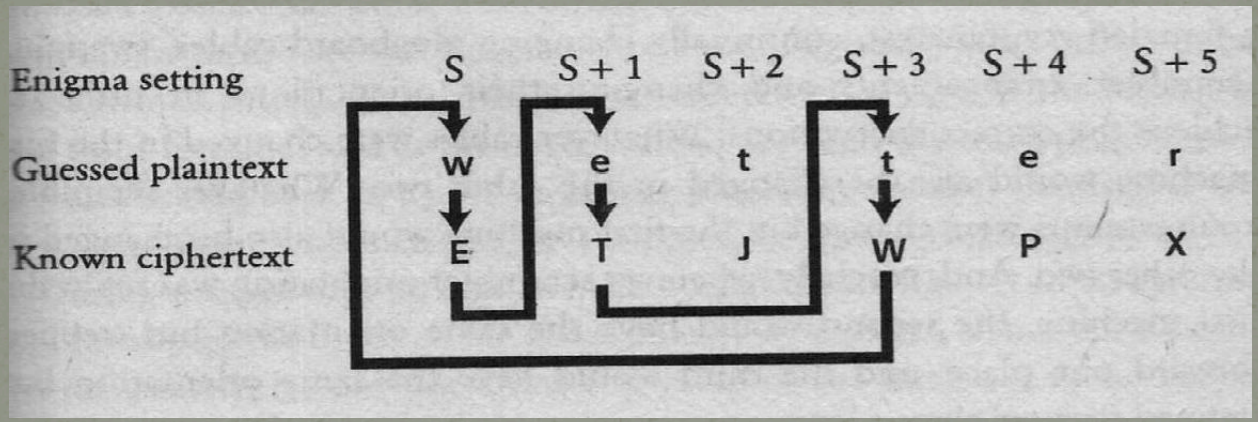
Bletchley Park and Turing

Turing's crib-based attack

- Crib: a known (guessed) plaintext and corresponding ciphertext pair
  - Obtained by knowledge of German procedures (e.g. from previously decrypted ciphertexts)
  - Messages at certain times of the day would follow a predictable pattern (e.g. weather reports)
    - E.g. Guessed plaintext: . . w e t t e r . . .
      - Known ciphertext: ...A G K E T J W P X R P Q ....
    - Bletchley Park workers accumulated a collection of such "cribs"
- Turing focused on exploiting such cribs to deduce the key

Turing's crib-based attack

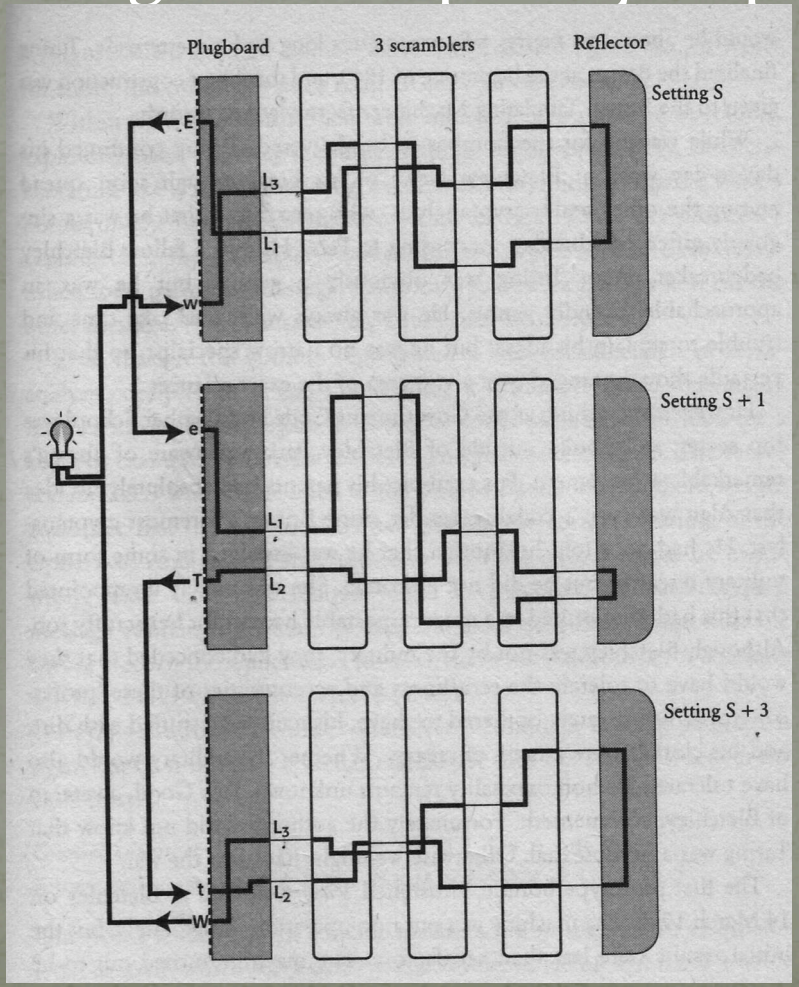
- Essence of Turing's idea:
  - Reduce key search space by looking for loops in cribs



  - Loops imply conditions only on rotor settings → Discard rotor settings that do not satisfy loop conditions (deduction by contradiction).
    - "Search space reduced from ~ 10^{20} to ~ 17,576 x 60 ~ 10^{6}!
      (from hundreds of years to milliseconds on a modern CPU!)
  - From rotor setting conditions, use crib to quickly determine plugboard cable settings one by one

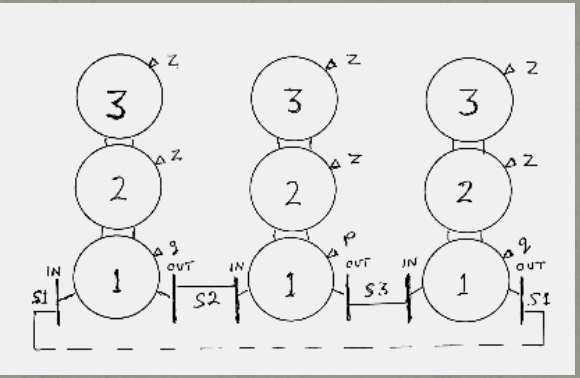
Turing's crib-based attack

Loops provide information dependent only on rotor settings, removing the complexity of plugboard



Automating the attack: The Bombe

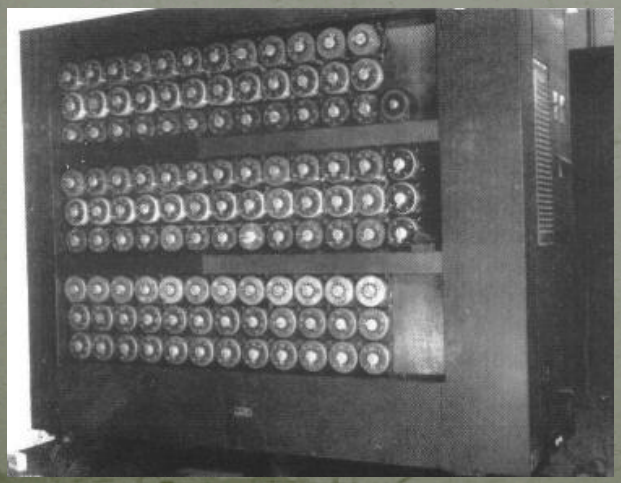
- Turing's Bombes consisted essentially of "batteries" of Enigma 3-rotor sections connected in series and fed back in a loop (with relative rotor orientations adjusted for particular loop)
- To allow for series connection, 3-rotor sections were built "double-ended" with separate 26 inputs and 26 outputs (forward and back wires in concentric rings)

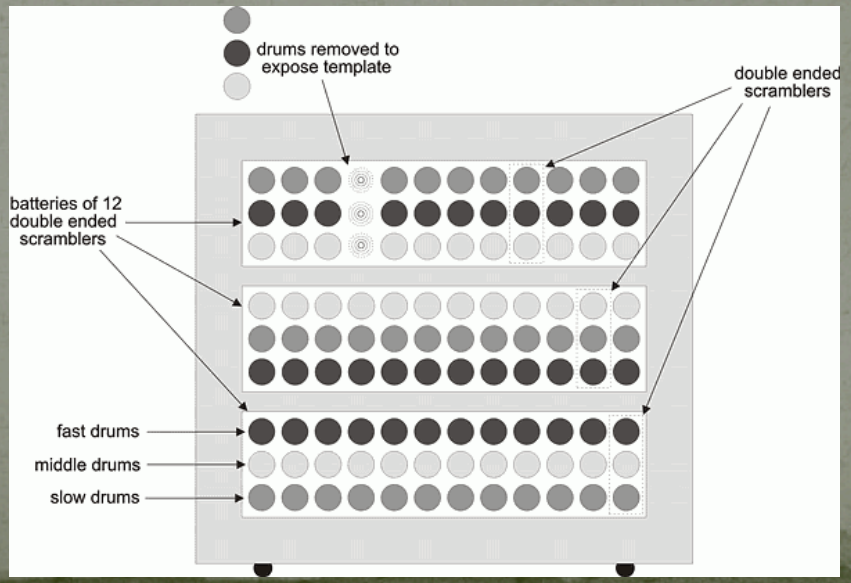


- Loop condition satisfied if a voltage applied at some input L3 on left comes out at L3 on right → no voltage on other 25 wires.

Automating the attack: The Bombe

- Electric motors were used to drive the fast rotors at 120 rpm (~ 2.5 hours for full search), with other two rotors incrementing every 26 (resp. 676) revolutions of the fast rotor
- Voltages on wires monitored for loop stopping conditions by relays and motors stopped if condition was satisfied.





Automating the attack: The Bombe

- More than 200 bombes were in operation by the end of the war, and are believed to have contributed greatly to the Allied war effort
- Turing made various improvements to bombe design during the war, and later faster bombes were built in the USA with Turing's consultation– Turing's improved techniques allowed the use of fewer bombes, leading to quicker implementation and deployment

Turing and The Naval (M4) Enigma

- Naval Enigma (M4) used on U-boats

was more difficult than other variants:

  - Used a more complex key establishment ("indication") procedure, initially unknown to British
  - From Feb. 1942, a fourth rotor was added
  - Very little cribs were known
- In Dec. 1939, Turing used intelligence-supplied clues to deduce the indication procedure.
- He also devised advanced statistical techniques (known as "Banburismus") to help further reduce the bombe search effort
- In 1940-41, this allowed some U-boat traffic to be deciphered (with the help of cribs deduced from captured German boats)
- In 1942, there was a period of little progress, until cribs were found from captured U-boat documents
- Also 4-rotor bombes were designed and led to US-navy Bombe

Turing's Legacy for Cryptography

- Huge key space does not necessarily imply strong security
  - Apparent security ("entangled" rotors/plugboard) based on intuitive arguments can be misleading
  - Careful mathematical analysis can reveal structural weaknesses → shortcut attacks
  - Motivation for modern approach of "provable security"
    - Mathematical proof relating hardness of breaking cipher to hardness of a well studied mathematical problem.

- Ciphers should be designed to remain secure even in the presence of extra known information
  - Known algorithm / plaintexts /cribs
  - Nowadays, also:
    - **chosen** plaintexts, and even
    - **chosen ciphertexts,**
    - **"side-channel" physical attacks**

References

- Singh S. “The Code Book”, 1999.
- Sebag-Montefiore H., “Enigma: The Battle for the Code”, 2000.
- http://www.codesandciphers.org.uk
- http://www.ellsbury.com/enigmabombe.htm
- http://users.telenet.be/d.rijmenants/en/enigma.htm
- http://www.bletchleypark.org/
- http://www.cryptomuseum.com/
- http://ed-thelen.org/comp-hist/NSA-Enigma.html