Specialized Cryptanalytic Machines: Two examples, 60 years apart

Patrick Schaumont

ECE Department

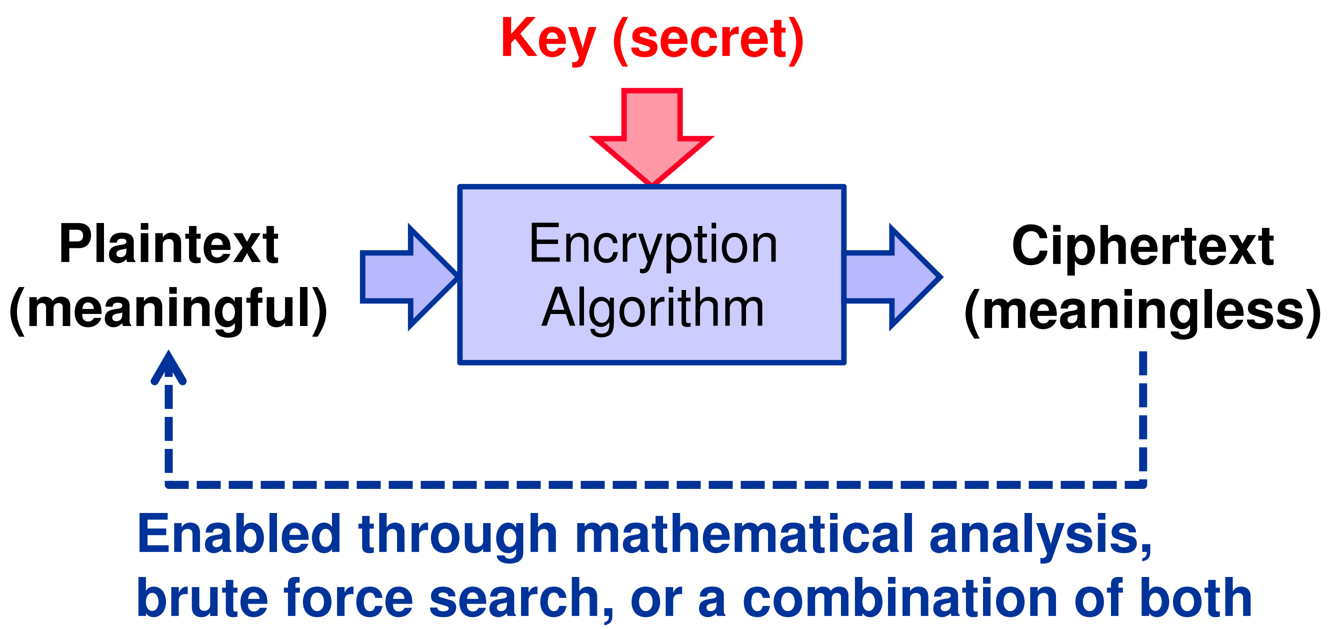
Virginia Tech

What is cryptanalysis?

- **Cryptography aims to defeat cryptanalysis**
- **Cryptanalysis aims to defeat cryptography**
- **Not just for the purpose of making movies ..**

Why do we need it?

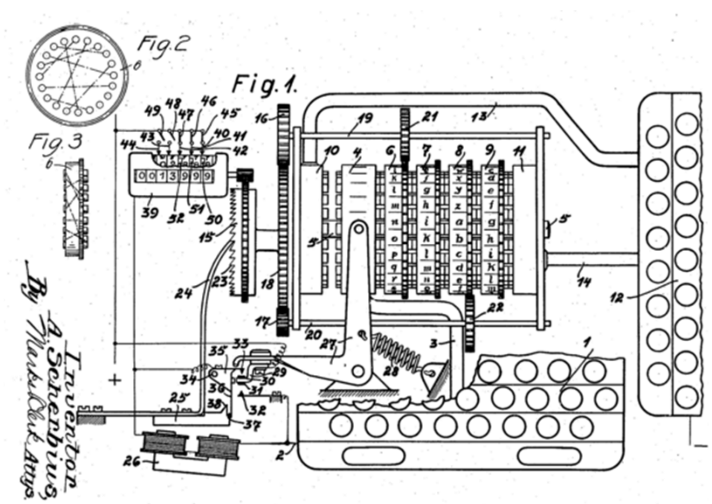
- **Cryptanalysis essential to understand the strength of an encryption algorithm**



What will I talk about?

Cryptanalysis of the

Enigma (1940)



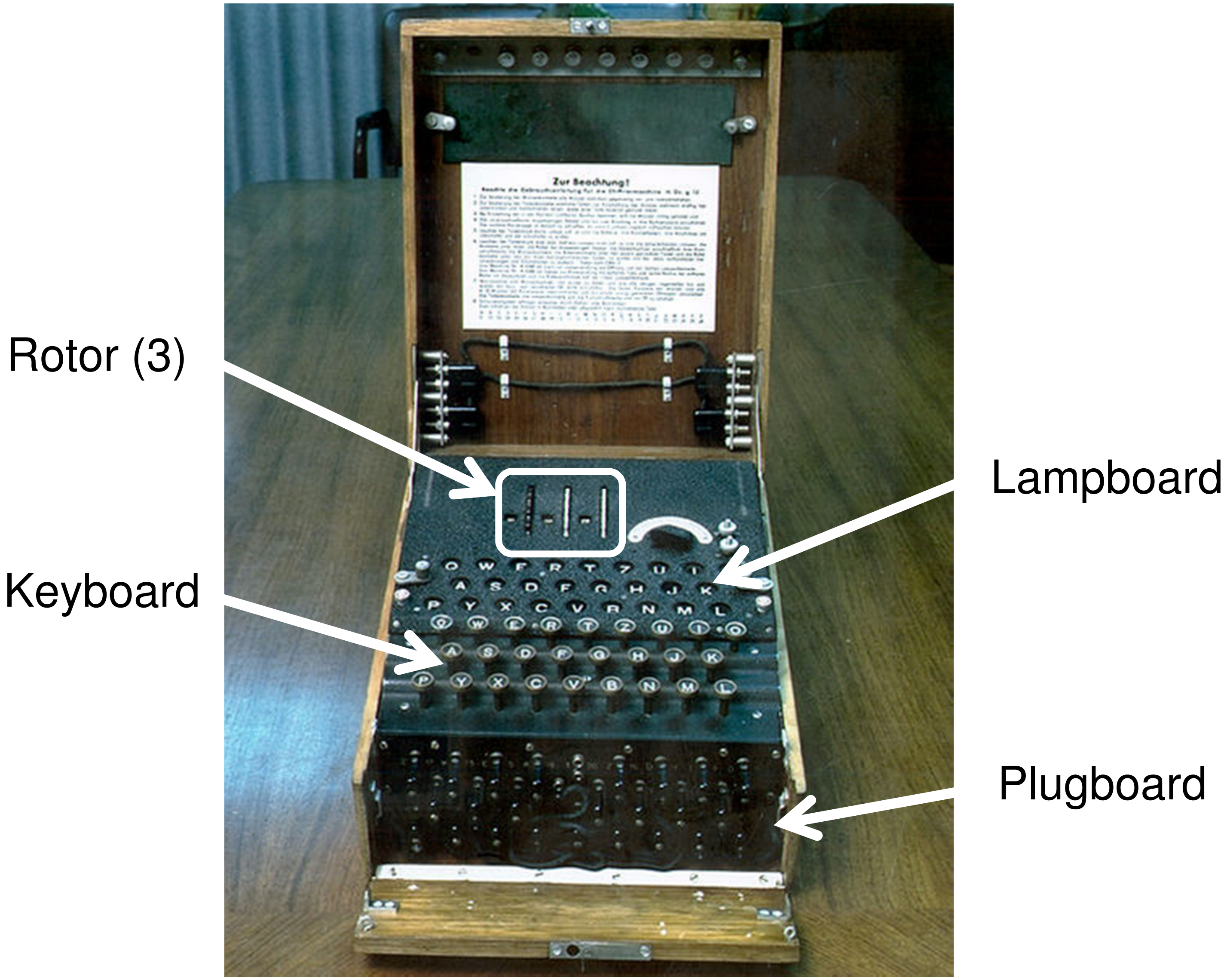
ECC2K-130 (2000)

======== ECC2K-130 ========
m = 131
f = x131 + x13 +x2 + x + 1
seedE = NO
a = 00 00000000 00000000 00000000 00000000
b = 00 00000000 00000000 00000000 00000001
seedP = 092FE1A8 9014D696 E6768756 1517586A A17BF123
U_x = 02 B8CB4816 38A7BB32 A5214816 621C9B9E
U_y = 07 CC4AAFC3 5046760A 6EF92D38 BFB9F5E1
P_x = 05 1C99BFA6 F18DE467 C80C23B9 8C7994AA
P_y = 04 2EA2D112 ECEC71FC F7E000D7 EFC978BD
h = 04
n = 2 00000000 00000000 4D4FDD57 03A3F269
seedQ= 328D0AE9 E6124D69 6E676875 61517565 06A34A25
V_x = 07 04AA2F3B 92953C63 B8CBB577 A6F83F07
V_y = 03 94249E7F 29B33ADE 47ABEE95 27EEE974
Q_x = 06 C997F3E7 F2C66A4A 5D2FDA13 756A37B1
Q_y = 04 A38D1182 9D32D347 BD0C0F58 4D546E9A

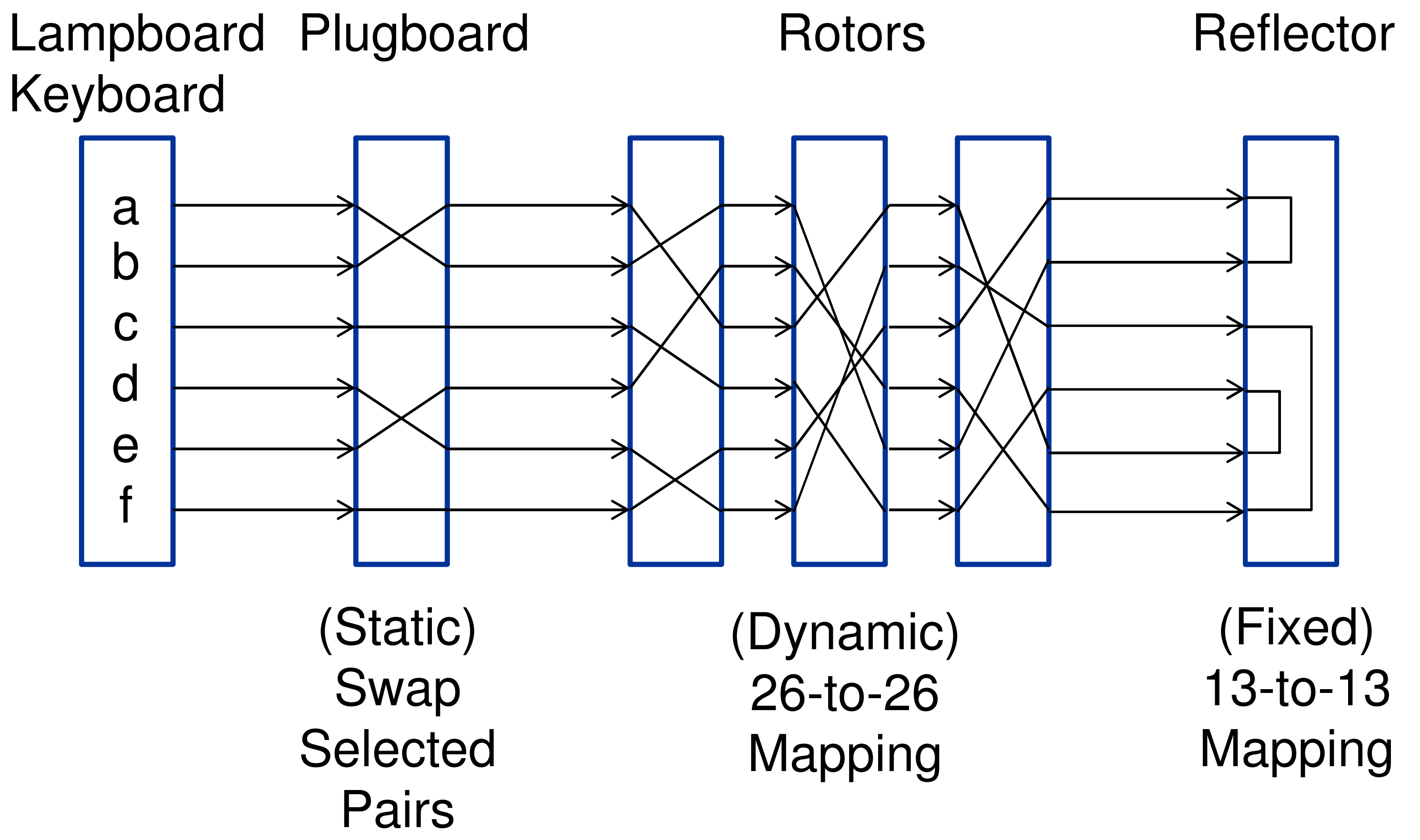
Example 1: Enigma

- **Used in Nazi Germany before/during World War II**
- **Initially broken by Polish Cipher Bureau (1932)**
  - Cryptanalysis refined by British/French Military Intelligence
  - Enigma Cryptanalysis had a major influence on the outcome of WW-II

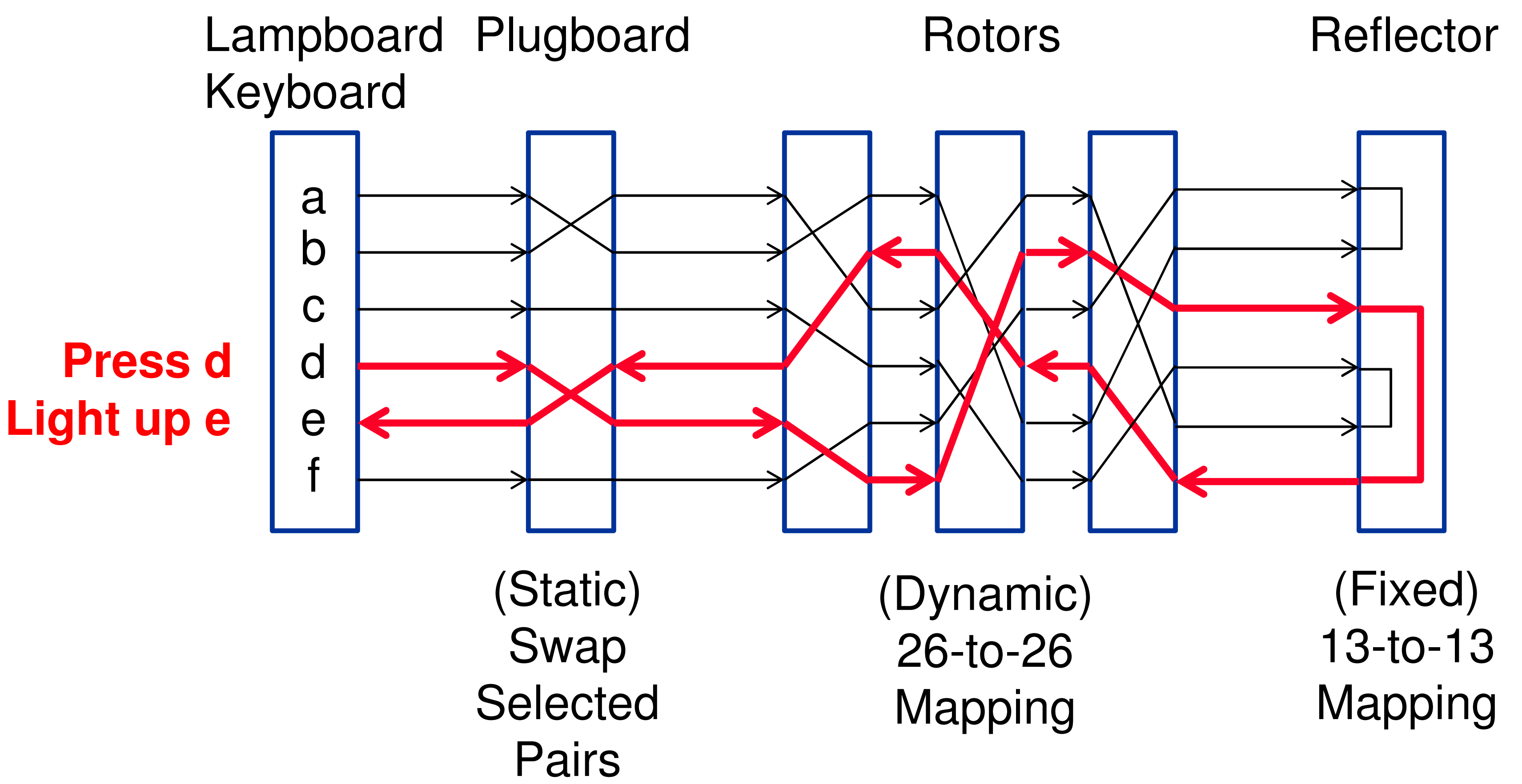
Enigma Cipher



Enigma Cipher

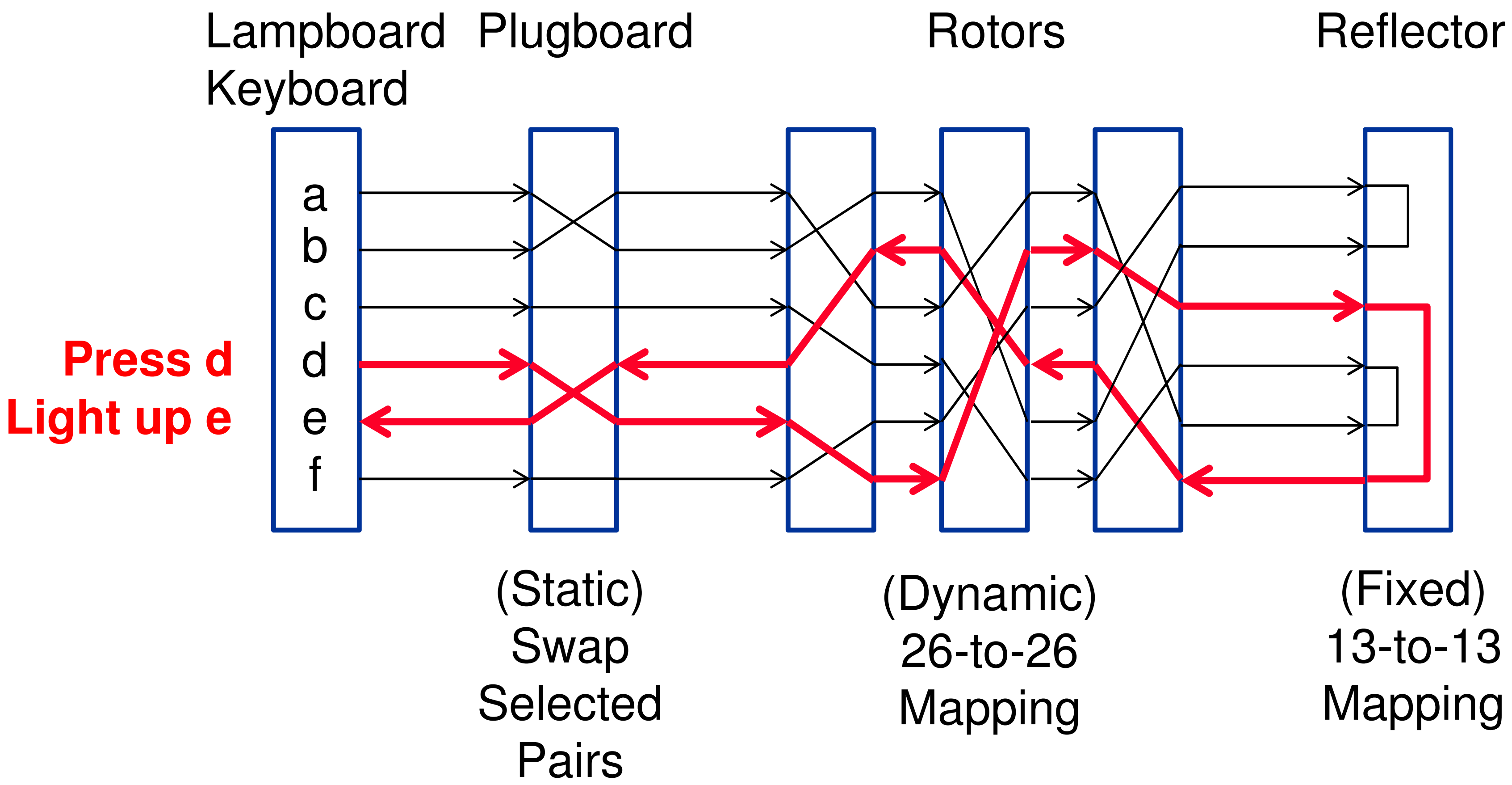


Enigma Cipher



Enigma Cipher

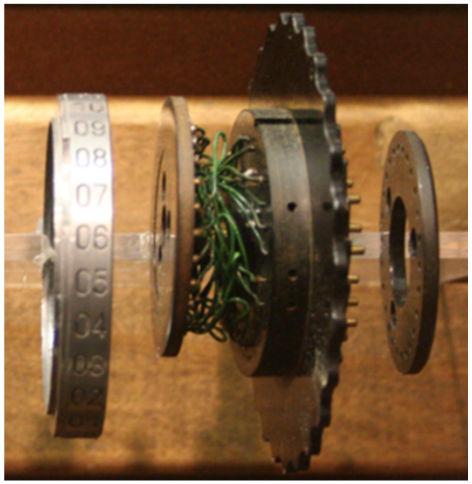
Each setting of the machine results in a reciprocal mapping of the plaintext alphabet into the ciphertext alphabet



- **After each letter of ciphertext, the rotors step in an odometer-like fashion**

Right-side View

Exploded View



Left-side View

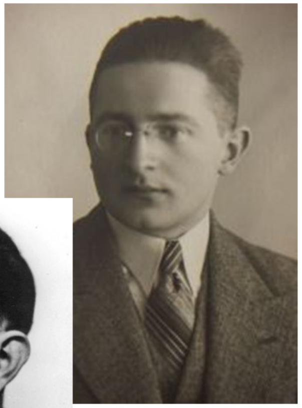
Enigma Strength

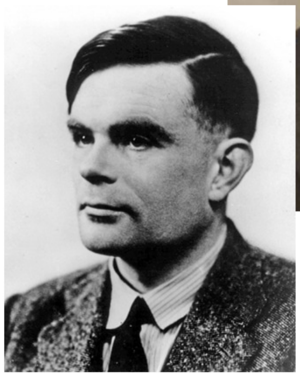
- **The Enigma machine itself was not secret**
  - **Secrecy is in the initial setting**
- **Number of initial positions: $1.074\ 10^{23}$**
  - **Rotor positions: 26 x 26 x 26 — 17576**
  - **Rotor selection (3 out of 5): 5 x 4 x 3 — 60**
  - **Ringstellung (notch): 26 x 26 — 676**
  - **PlugBoard (10 plugs): $150\ 10^{12}$**

- **Equivalent strength: 76 bit key**

Breaking the Enigma

- **An 80-bit key is hard to identify by brute-force search, especially in a time without electronic computers**
- **Cryptanalysis by Rejewski (Polish Cipher Bureau), and Turing (GCCS) reduced complexity to a 30-bit search !**
- **They also build a machine to perform this 30-bit search: the <u>Bombe</u>**





Known-Plaintext Attack

Received ciphertext from a weather ship:

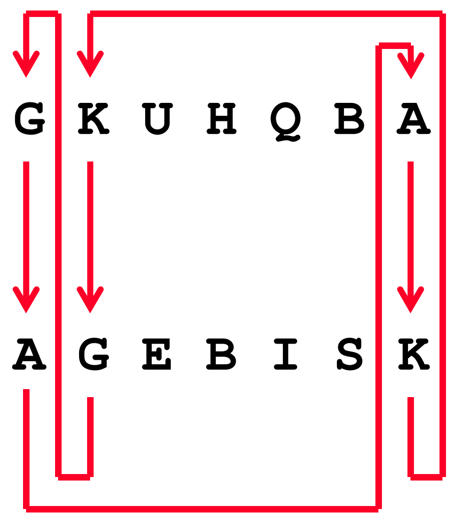
`R W I V T Y R E S X B F O G K U H Q B A I S E`

Crib (= guess at its meaning)

`W E T T E R V O R H E R S A G E B I S K A Y A`

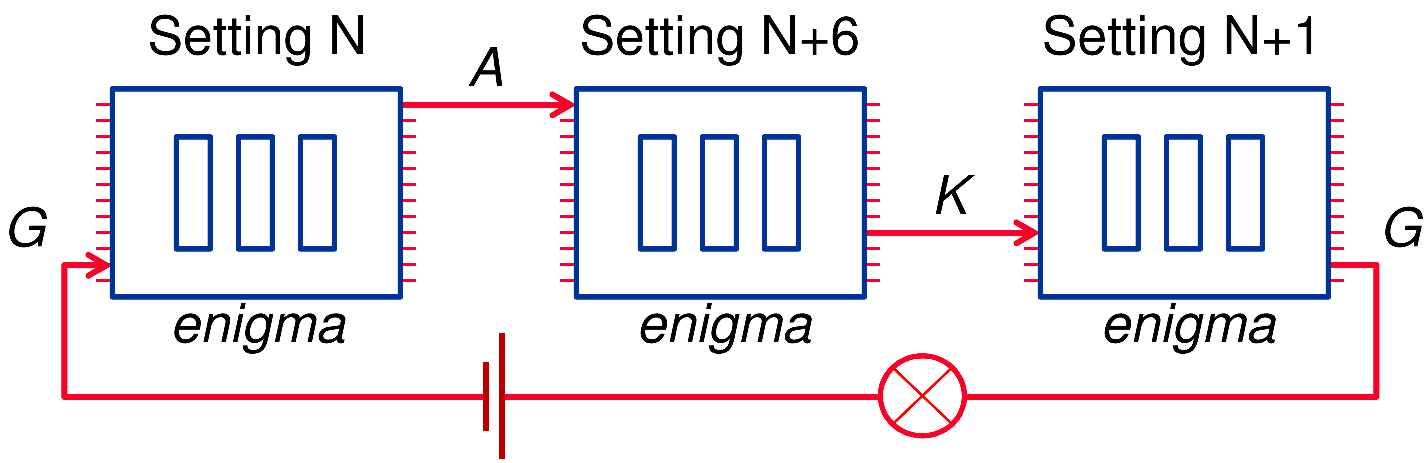
Find Loops in Ciphertext and Crib

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
R	W	I	V	T	Y	R	E	S	X	B	F	O	G	K	U	H	Q	B	A	I	S	E
W	E	T	T	E	R	V	O	R	H	E	R	S	A	G	E	B	I	S	K	A	Y	A



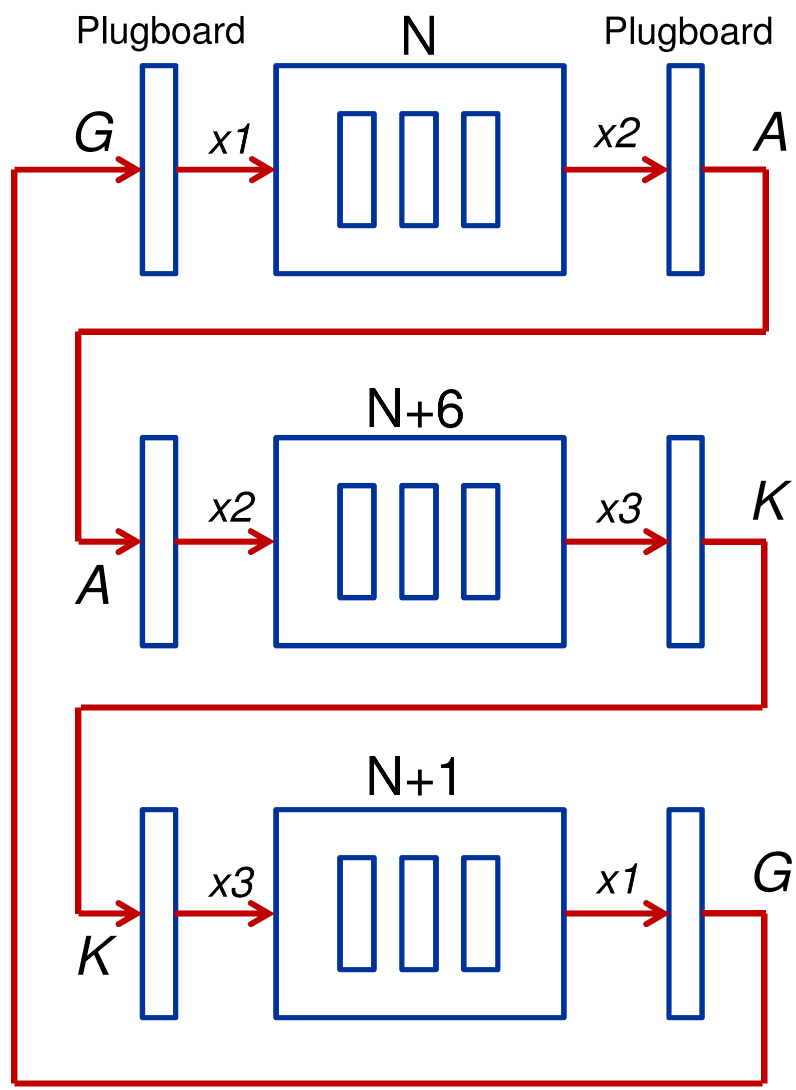
Parallel Search with Enigma Machines

Take 3 Enigma's and wire them up as follows

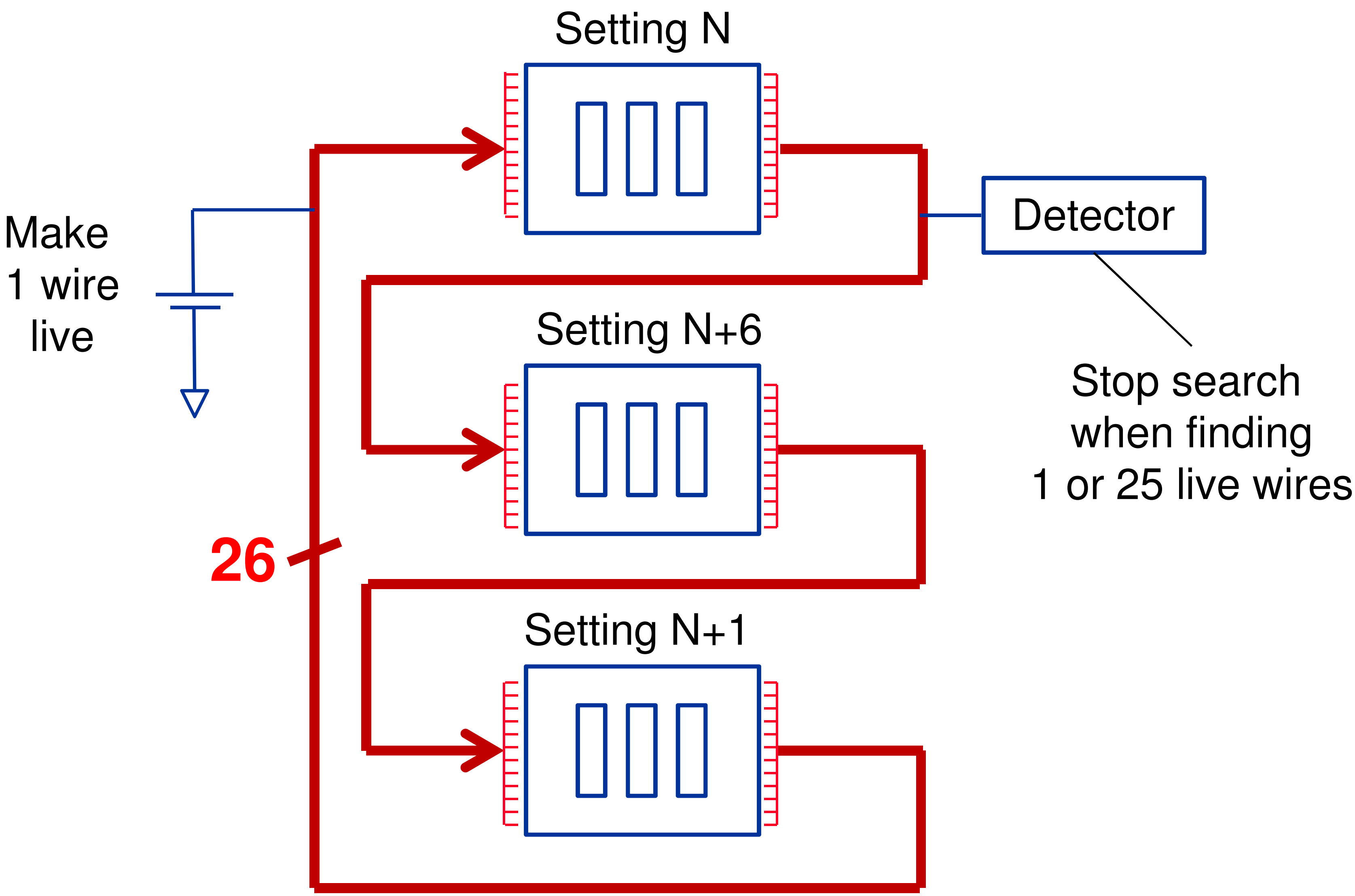


Next, try all rotor positions until a closed loop is found. A closed loop indicates a possible match

Dealing with the Plugboard



Dealing with the Plugboard



The Bombe

Bombe Efficiency

- **Number of initial positions: $1.074\ 10^{23}$**
  - **Rotor positions: 26 x 26 x 26 — 17576**
  - **Rotor selection (3 out of 5): 5 x 4 x 3 — 60**
  - **Ringstellung (carry): 26 x 26 — 676**
  - **PlugBoard (10 plugs): ~~$150\ 10^{12}$~~**

- **Need to test only $712\ 10^{6}$ positions**
  - **Easy to run in parallel on up to 60 Bombes, each with a different Rotor selection**

Example 2: ECC Challenge

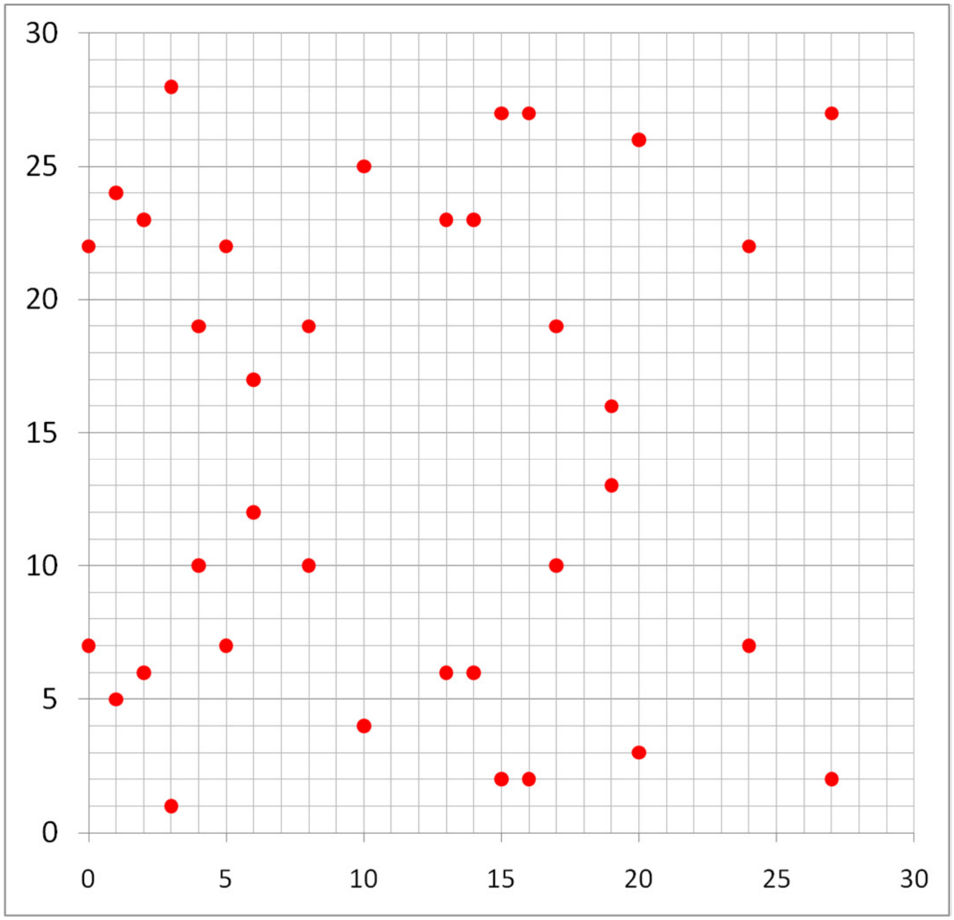
- **Elliptic Curve Cryptography uses Elliptic Curves over Finite Fields**
  $y^2 = x^3 + ax + b$ over GF(p)

- **Prime Field GF(p)**
  - **integers 0 up to p-1**
  - **addition mod p, multiplication mod p**

- **The EC Curve contains all points (X,Y) in GF(p) for which the equation holds**

Example Curve over GF(p)

- **Points of $y^2 = x^3 + 4x + 20$ over GF(29)**

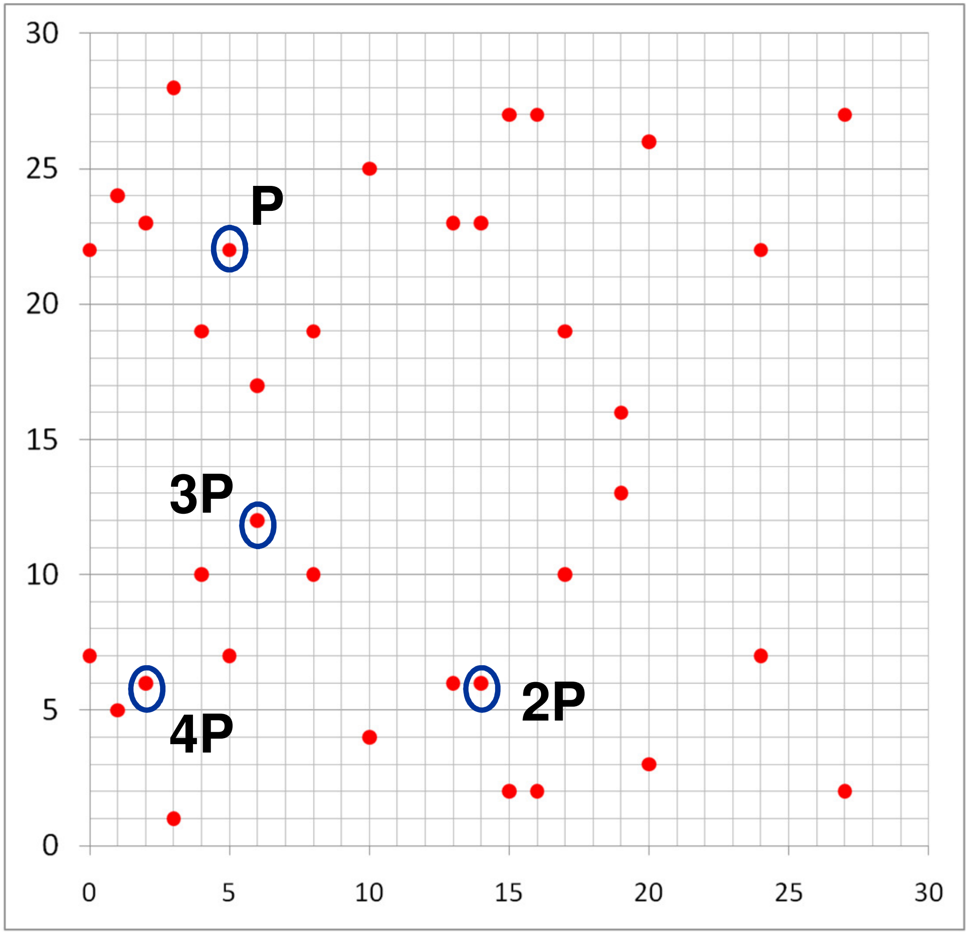


Point Operations

- **EC points related through Point operations**
  - **Point addition: Q = P1 + P2**
- **With proper choice of curve parameters, all points from a *group***
  - **{∞, P, 2P=P+P, 3P=P+P+P, 4P, ..., (#E-1).P}**

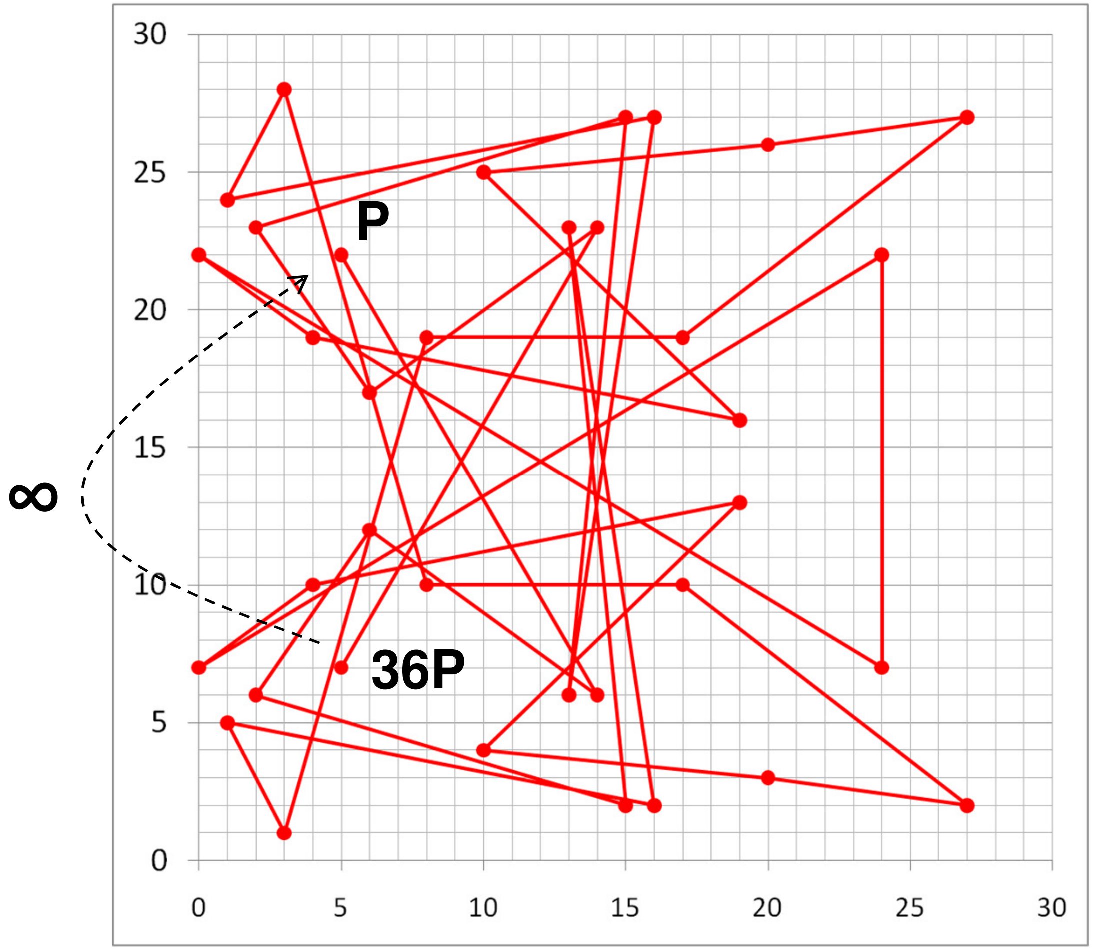
Example Curve over GF(p)

- **Points of $y^2 = x^3 + 4x + 20$ over GF(29)**



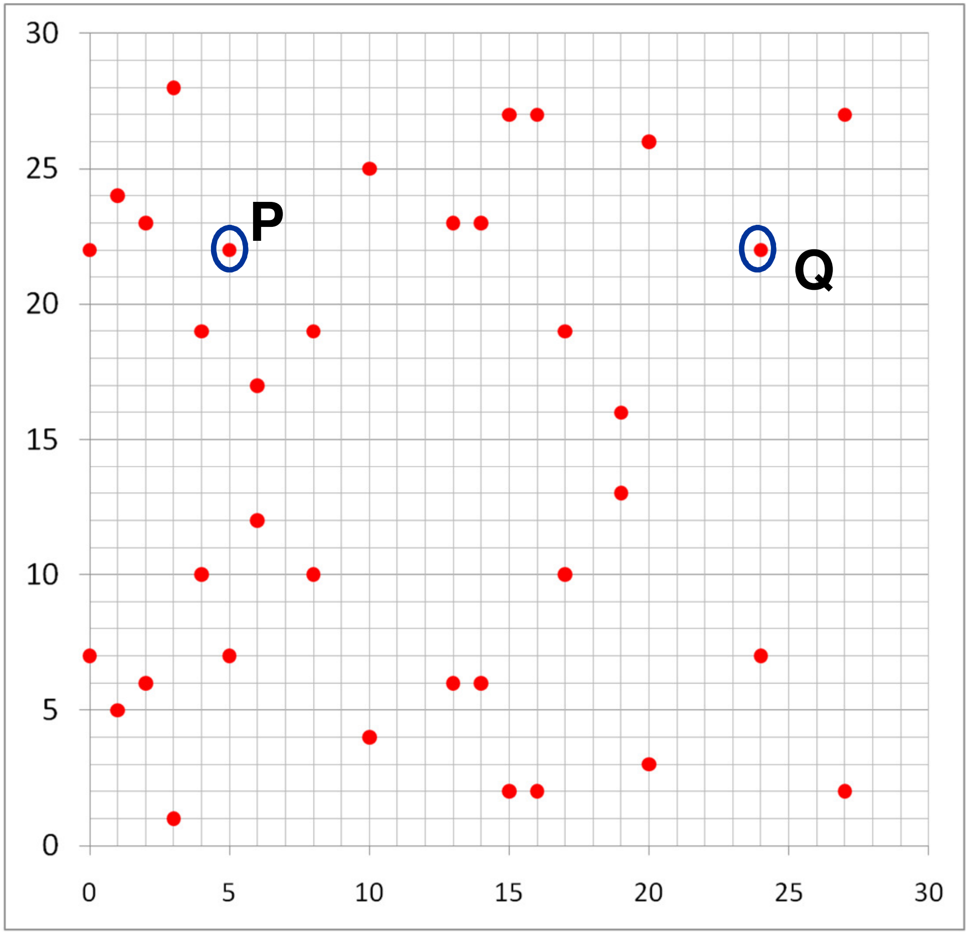
Example Curve over GF(p)

- **Points of $y^2 = x^3 + 4x + 20$ over GF(29)**



Cryptography using EC Points

- **Given P and Q = n.P, what is n?**



Certicom Challenge

- **Certicom has defined (1997) a "challenge": Given Q, P and curve. Find n?**

Challenge	Field size (in bits)	Estimated number of machine days	Prize (US$)
ECC2K-108	109	1.3×10^6	$10,000
ECC2-109	109	2.1×10^7	$10,000
ECC2K-130	131	2.7×10^9	$20,000
ECC2-131	131	6.6×10^{10}	$20,000

Broken: ECC2K-108, ECC2-109

Current Target: ECC2K-130

(Additional Challenges up to 358 bit field size (and $100K reward) exist)

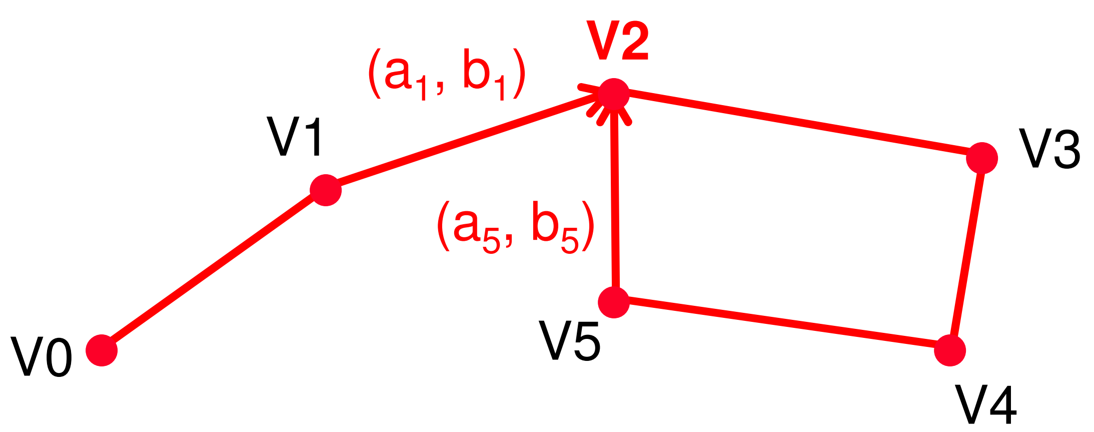
Solving ECDLP

- **Best known mechanism to solve Q = n.P is an efficient randomized search (!)**
  - **Generate random points V_i:**
    $$V_i = a_i .P + b_i .Q$$
  - **Until a collision occurs:**
    $$V_i = V_j \quad \text{but} \quad (a_i,b_i) \neq (a_j,b_j)$$
  - **Then solve for n:**
    $$n = (a_i – a_j) . (b_j – b_i)^{-1}$$

    $\sqrt{2^{130}} \rightarrow 2^{65}$!!
  - **Picking random (a, b), a collision is expected after considering sqrt(p) points**

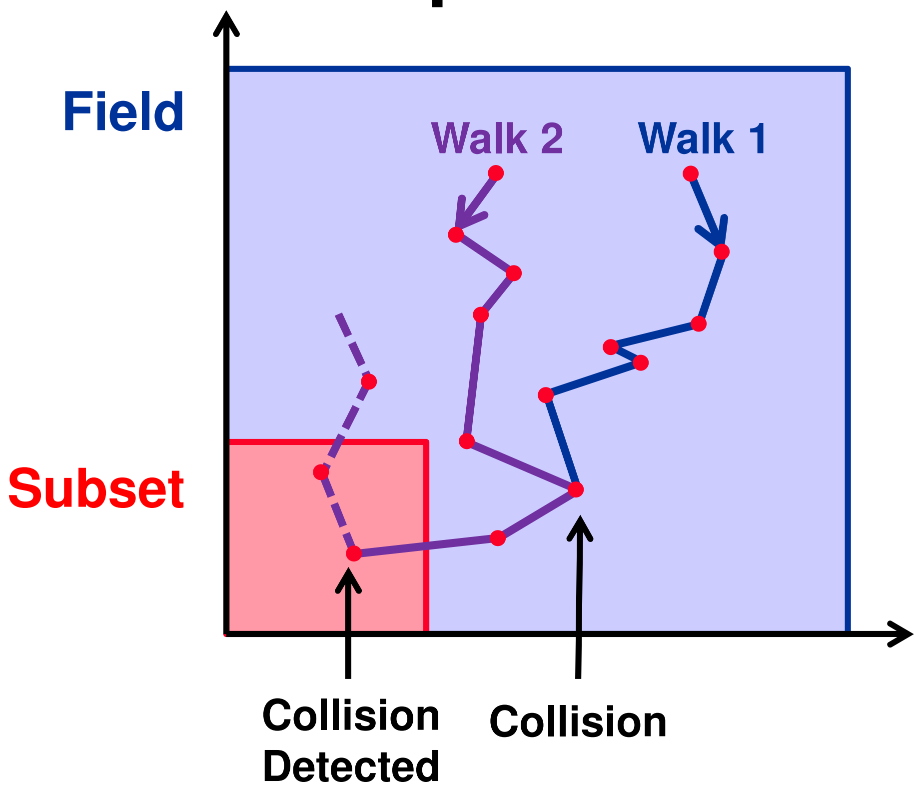
Pollard rho: Efficient Search

- **[Pollard 1976] To avoid excessive storage requirements, generate random points using a random walk**
- **Finite number of EC points, so random walk will be a cycle**



Parallelized Random Search

- **[Van Oorschot 94] Execute multiple random walks at a time**
- **Collect subset of points on a server**



Estimated Efficiency

- **How fast can we walk?**

http://www.ecc-challenge.info

Platform	Steps per Second	# Machines to break ECC130K in one year
Opteron 875 (2 core, 2.2GHz)	4.17 million	16,360
Core 2 Q6850 (4 core, 3 GHz)	22.45 million	4054
Playstation 3 (CELL with 6 SPE)	27.67 million	2466
GTX 295 GPU (60 core, 1.24GHz)	54.03 million	1263

Breaking ECC2k-130

http://eprint.iacr.org/2009/541.pdf

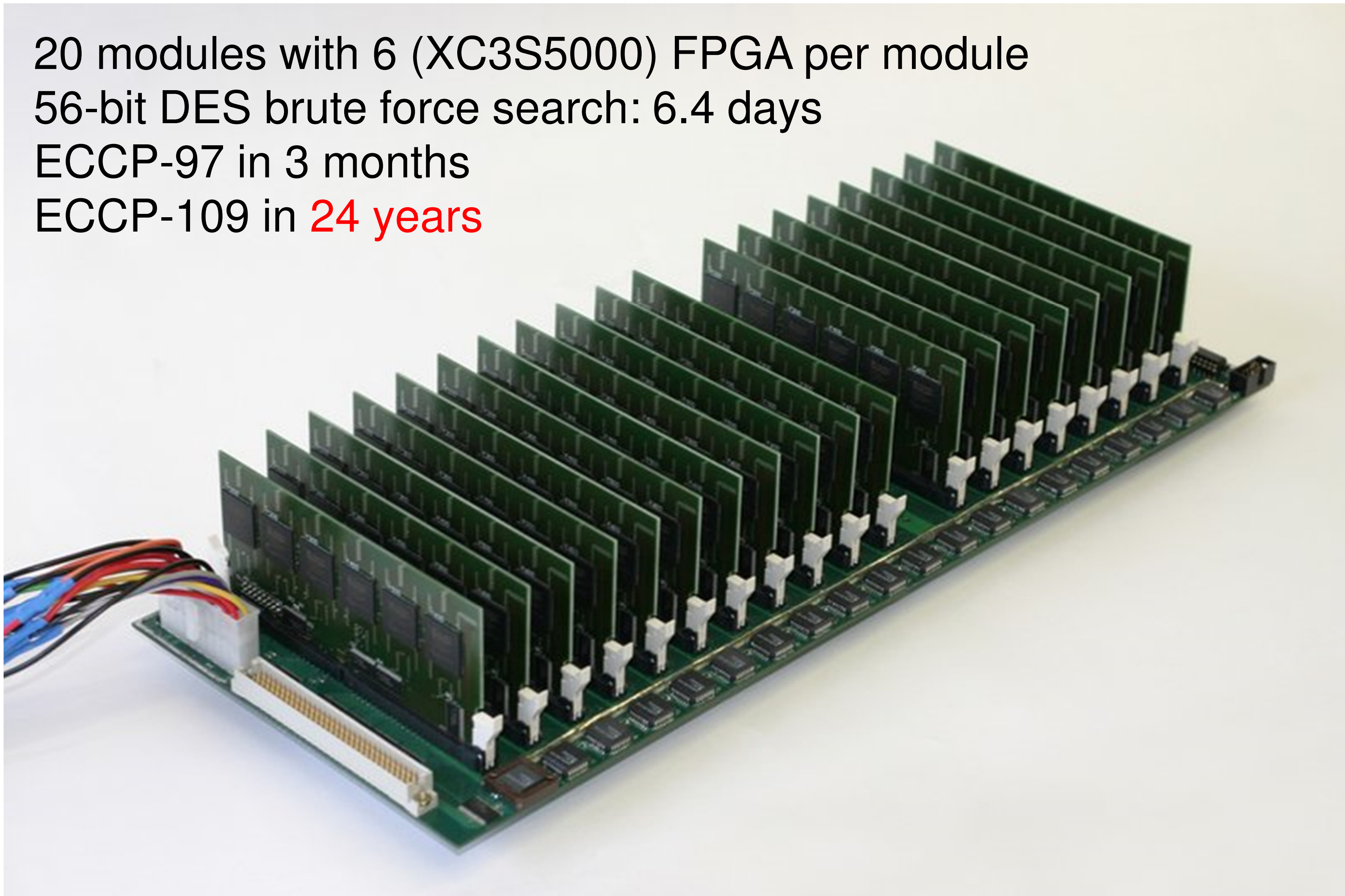
Daniel V. Bailey[1,10], Lejla Batina[2], Daniel J. Bernstein[3], Peter Birkner[4], Joppe W. Bos[5], Hsieh-Chung Chen[6], Chen-Mou Cheng[7], Gauthier Van Damme[2], Giacomo de Meulenaer[8], Luis Julian Dominguez Perez[9], Junfeng Fan[2], Tim Güneysu[10], Frank Gürkaynak[11], Thorsten Kleinjung[5], Tanja Lange[4], Nele Mentens[2], Ruben Niederhagen[12], Christof Paar[10], Francesco Regazzoni[8], Peter Schwabe[4], Leif Uhsadel[2], Anthony Van Herrewege[2], and Bo-Yin Yang[6]*

[1] RSA, the Security Division of EMC, USA
dbailey@rsa.com
[2] ESAT/SCD-COSIC, Katholieke Universiteit Leuven and IBBT
Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium
[3] Department of Computer Science
University of Illinois at Chicago, Chicago, IL 60607–7045, USA
djb@cr.yp.to
[4] Department of Mathematics and Computer Science
Technische Universiteit Eindhoven, P.O. Box 513, 5600 MB Eindhoven, Netherlands
p.birkner@tue.nl, tanja@hyperelliptic.org, peter@cryptojedi.org
[5] EPFL IC IIF LACAL, Station 14, CH-1015 Lausanne, Switzerland
{joppe.bos, thorsten.kleinjung}@epfl.ch
[6] Academia Sinica, Taiwan
{kc,by}@crypto.tw
[7] National Taiwan University, Taiwan
doug@crypto.tw
[8] UCL Crypto Group, Place du Levant, 3, B-1348 Louvain-la-Neuve, Belgium
{giacomo.demeulenaer, francesco.regazzoni}@uclouvain.be
[9] Claude Shannon Institute for Discrete Mathematics, Coding and Cryptography.
School of Computing, Dublin City University, Ireland
ldominguez@computing.dcu.ie
[10] Horst Görtz Institute for IT Security, Ruhr University Bochum, Germany
{gueneysu, cpaar}@crypto.rub.de
[11] Microelectronics Design Center, ETH Zürich, Switzerland
kgf@ee.ethz.ch
[12] Lehrstuhl für Betriebssysteme, RWTH Aachen, Germany
ruben@polycephaly.org

Other Efforts: COPACOBANA

http://www.copacobana.org

20 modules with 6 (XC3S5000) FPGA per module
56-bit DES brute force search: 6.4 days
ECCP-97 in 3 months
ECCP-109 in 24 years



Conclusions

1940 - Enigma

- Analysis Target: 80 bit key
  - Search complexity 30 bit
- Weight:
  - Bombe: 1000 Kg
  - Enigma: 5 Kg
- Electromechanical Analysis
  - 120 keys per minute
- Time to success
  - One day

2000 - ECC2K-130

- Analysis Target: 130 bit key
  - Search complexity 65 bit
- Weight:
  - Distributed CPU: 1000 Kg
  - ECC: 100 g (98 g battery)
- Electronical Analysis (2010)
  - 3 Gkeys per minute (on GPU)
- Time to success
  - One year

Conclusions

1940 - Enigma

- Analysis Target: 80 bit key
  - Search complexity 30 bit
- Weight:
  - Bombe: 1000 Kg
  - Enigma: 5 Kg
- Electromechanical Analysis
  - 120 keys per minute
- Time to success
  - One day

2000 - ECC2K-130

- Analysis Target: 130 bit key
  - Search complexity 65 bit
- Weight:
  - Distributed CPU: 1000 Kg
  - ECC: 100 g (98 g battery)
- Electronical Analysis
  - 3 Gkeys per minute (on GPU)
- Time to success
  - One year

Despite the wonders of Moore, Advanced VLSI design, Cryptanalytic machines did not hold up to the improvements in Cryptography

this is good news :)

Learning more

- Enigma
  - D. Rijmenants: http://users.telenet.be/d.rijmenants
  - T Sale: http://codesandciphers.org.uk
  - G. Ellsbury: http://ellsbury.com
  - F. Weierud: http://cryptocellar.web.cern.ch
- ECC2K-130
  - Certicom Challenge: http://www.certicom.com
  - Search: http://ecc-challenge.org
  - Search: http://eprint.iacr.org/2009/541