Maths from the talk
“Alan Turing and the Enigma Machine”

James Grime

1 The Enigma Machine

By the beginning of the 20th century it had become possible, and necessary, to mechanise encryption. In 1918 a German engineer named Arthur Scherbius patented the Enigma Machine. Originally it was sold to banks, railway companies and other organisations who needed to send secret information. By the mid-1920s the German military started to use the Enigma Machine, with some differences from the commercial version of the machine. Enigma was used by the German military throughout World War II, therefore breaking the enigma cipher became a top priority, first by the Polish, then later by the British and Americans.

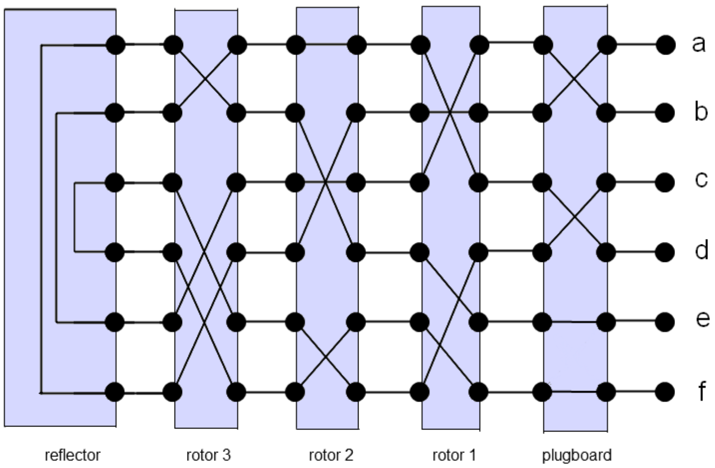
The Enigma Machine was an electro-mechanical machine, about the size of a typewriter, made with steel casing inside a wooden box. On the outside you will see two sets of letters which were the keyboard and the lampboard. You would type your message using the keyboard, the message would then be encrypted letter-by-letter, but instead of printing on paper the encrypted letters would light up on lampboard. The encrypted message would then be written down by the operator and would be transmitted by radio. The machine itself did not transmit.

Essentially, the Enigma Machine is just a large circuit. When you type in a letter on the Enigma Machine it completes a circuit and lights up a letter on the lampboard.

The Enigma Machine has five components;

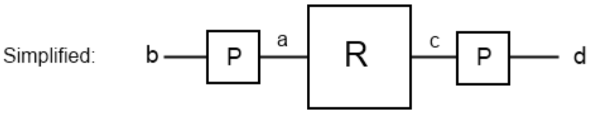
- Rotors 1, 2 and 3: Inside each rotor is a criss-cross of wires, connecting each letter to another in the manner of a general substituion cipher.

- The reflector: The reflector connected 26 letters of the alphabet into 13 pairs.
- The plugboard: The plugboard connected some letters in pairs, and left other letters unpaired. The plugboard is a military addition and is the main difference between this and the commercial Enigma Machine.



The end result is a monoalphabetic substitution cipher.

Let's simplify this picture by calling the plugboard P, and combining the effect of the rotors and reflector into one unit called R. The machine then takes an input, say 'b', goes through the plugboard and produces 'a', goes through the rotors and reflector and produces 'c', then goes through the plugboard a second time and finally outputs 'd'.



The reflector turns the 26 letters of the alphabet into 13 pairs. These pairs are known as a *product of transpositions*. By virtue of being sandwiched in the middle of the machine, the reflector actually determines the behaviour of the whole machine. In other words, the Enigma Machine itself is a product of transpositions, where each letter is encrypted as its paired partner.

In the example above, 'b' becomes 'd' and 'd' becomes 'b'. This means the simplified diagram works in both directions. A product of transpositions makes decryption easy, since encrypting a letter twice will return the original message. We say a product of transpositions is *self inverse*.

In other language, if E is the output of Enigma for a given position, then $E(b) = d$ and $E(d) = b = E(E(b))$.

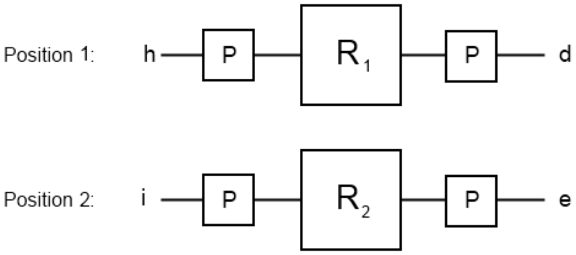
Since the Enigma Machine is the combined result of the plugboard, P, followed by the combined result of the rotors and refelector, R, then followed by the plugboard again, the ouput for a given position may be written as

$$E = PRP.$$

At least one rotor moves after each letter is typed into the machine. This means each letter of a message is encrypted using a different substitution cipher. This is called a *polyalphabetic cipher*. For example, here are the first ten letters of the two monoalphabetic ciphers needed to send the message 'hi'.

input:	a	b	c	d	e	f	g	h	i	j
output 1:	f	i	e	h	c	a	j	d	b	g
output 2:	h	g	j	f	i	d	b	a	e	c

Here, the message 'hi' becomes 'de'. Since each cipher turns the 10 letters into 5 pairs, encrypting a message twice will return the original message. The two outputs can be written using two simplified diagrams.



1.1 Number of ciphers

Each component had a number of possibilities:

- Rotors 1, 2, 3: There are three slots in the machine for rotors. Initially there were only three rotors to use, (labelled I, II, III) but they could be used in any order. There are 6 ways to arrange these rotors. By 1939 this had increased to a choice of five rotors (adding rotors IV, V) for the army and air force, and eight rotors (adding rotors VI, VII, VII) for the navy. There are 60 ways to choose 3 rotors from 5, and 336 ways to choose 3 rotors from 8.
- The Reflector: The reflector was generally not a choice, but fixed and unchanged.
- The Plugboard: Initially the plugboard contected 12 letters into 6 pairs, leaving 14 letters unchanged. The number of ways to choosing 12 letters from 26 is 26!/14!. We then form 6 pairs; the order of the pairs do not matter so we further divide this number by 6!. Finally, the order of the two letters within the pairs do not matter, i.e. AB is the same pair as BA, so we further divide by 2 for each pair. So, the total number of ways to connect 12 letters into 6 pairs is $26!/2^6 6!14! \approx 1.0 \times 10^{11}$. By 1939, the plugboard turned 20 letters into 10 pairs, increasing the number of combinations to 1.5×10^{14}.

Finally, each individual message was encrypted with its own *message key* - this was the three rotor starting positions. Each rotor has 26 starting positions, giving $26 \times 26 \times 26 = 17576$ message keys.

So initially, the total number of input-outputs for the Enigma Machine (individual monoalphabetic ciphers) was

$$\frac{6 \times 26! \times 26^3}{2^6 6! 14!} \approx 1.06 \times 10^{16}$$

By 1939, increases in the number of rotors and plugboard pairs had increased this figure 1.59×10^{20}. As you can imagine, breaking the enigma code was quite a task! However, this number is only a measure of how difficult a cipher is to break by exhaustive key-checking (brute force!). What really made enigma so difficult to break was the way the rotors moved during encryption.

Rotor 1 moved after every letter. When rotor 1 had done a full revolution it would kick rotor 2 one place. When rotor 2 had done a full revolution it would kick rotor 3 one place. So, there was a fast moving rotor, a middle rotor and a slow moving rotor. Due to a quirk of the middle rotor called 'double stepping', the rotors actually had a period of $26 \times 25 \times 26 = 16900$.

You could also rotate the rotor labeling and kickover point in relation to the internal wiring, this was called the 'Ring Setting'. This made no difference to the number of individual monoalphabetic ciphers so is not usually included in the number of keys. However, by changing the kickover point in relation to the internal wiring, you change the pattern in which the machine moves between individual monoalphabetic ciphers. So, the number of keys is protecting you from brute-force attack, while the system of moving rotors protects you from frequency analysis.

2 Polish Code Breaking

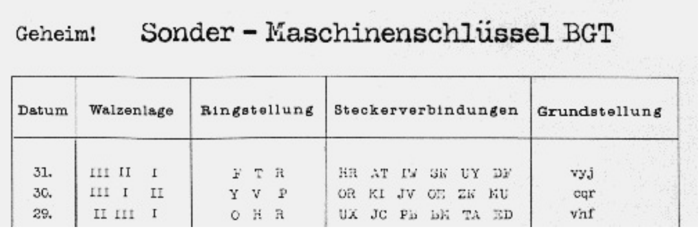
Polish Intelligence were initially unable to break the German Enigma traffic, however driven by the imperative of finding what the Germans were up to, they, uniquely among other nations at that time, decided to try a mathematical approach. In 1932 a team of young mathematicians was set up. It included Jerzy Rozycki, Henryk Zygalski and Marian Rejewski.

Rejewski was given a separate room and told to take a closer look at a pile of the military Enigma encrypts. He was also supplied with an obsolete commercial Enigma machine which had been bought in Germany. Using this, and a series of mathematical deductions, he was able to completely determine the wiring of the rotors now being used in the military Enigma.

They put the new rotors into the machine and... it didn't work. You see on the commercial machine, the keyboard was connect to the first rotor in alphabetical order. So Q became A, W became B and so on. The military machine was different! There are 400 million billion billion ways to connect the keyboard to the first rotor. Then Rejewski wondered whether they had been so foolish as to connect A to A, B to B' and so on. And they had! By 1933 the Poles has a working Enigma replica.

2.1 Characteristic Catalogue

At this point the Enigma Machine was only using three rotors, which could be placed in any of 6 possible arrangements, the plugboard only connected 12 letters into pairs. The rotor order, plugboard arrangement, along with the Ring Setting and Ground Setting, were written down for the Enigma operator on a key sheet.



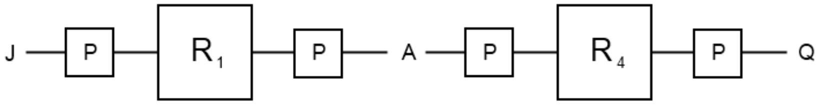
Geheim! Sonder - Maschinenschlüssel BGT

Datum	Walzenlage	Ringstellung	Steckerverbindungen	Grundstellung
31.	III II I	F T R	HR AT IW SK UY DF	vyj
30.	III I II	Y V P	OR KI JV OE ZK MU	cqr
29.	II III I	O H R	UX JC PB BM TA ED	vhf

The only thing not written down on the key sheet were the rotor starting positions. These were three letters that appeared in the windows at the top of the machine when you turn the rotors. Each message could be sent using its own starting position, this position would be sent at the start of each message and encoded using the Enigma Machine itself!

For example, the operator would set his machine to the Ground Setting, say XYZ, then encode his chosen starting position, say ABC, using the machine. He would encode this twice, so ABCABC might become JTEQGL. It was this repetition of the message key that was to be their undoing.

Notice A appears twice in the secret message key. When I pressed A with the rotors in the first position I got J. When I press A with the rotors in the fourth position I got Q. So I can wire up two enigma machine together, like this



Here, A becomes J with the rotors in the first position, and A becomes Q with the rotors in the fourth postion. So altogether J becomes Q.

So the message indicator JTEQGL tells me J becomes Q when I hook up two enigma machines. That's a clue. I can't predict what one enigma machine will do, but I can deduce what two enigma machines will do when hooked together.

By looking at the first and fourth positions of all message indicators, sent using the same Ground Setting, they were able to find cycles. For example, A second message indicator such as QBMTOZ tells me Q becomes T, while a third message indicator such as TPNJQC tells me T becomes J. Altogether, these three message indicators makes a cycle (JQT)

Now you may be receiving dozens of messages everyday. Here are some more message indicators:

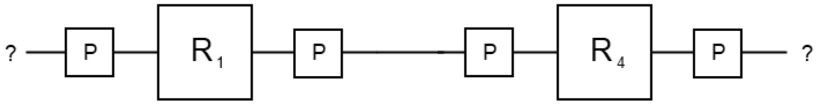
JTEQGL QBMTOZ TPNJQC
AQBPEX BXVBHV CUCFYB
DAXUFM ERZZXN FJLKIK
GZKNDH HFJEZJ IMHIPG
KEGYSF LVFXMD MIDGWA
NDSDAS OYARNT PKPCRY
RCOALE SHIVBR UOUHTW
VWYOKU WNTWVI XLRSUQ
YSWLJO ZGQMCP

Keep going as before and you'll get cycles for the whole alphabet, namely (APCFKYLXSVOR)(DUHEZMGN)(JQT)(B)(I)(W). The lengths of these cycles were called the *characteristic*. So our example has characteristic 12, 8, 3, 1, 1, 1.

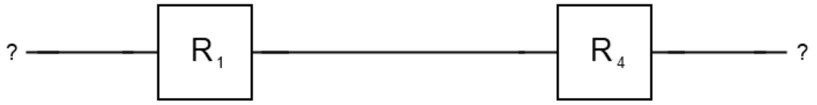
The characteristic was like a fingerprint, although not quite as unique. All you needed to do was to find the characteristic for each setting, then you can use the message indictors to tell you which setting is being used.

We have seen there were ten thousand million million settings. Cataloging the characteristic for each setting would be a huge task, but fortunately you could ignore the plugboard!

We had this picture before:



The two plugboards in the middle aren't necessary because they cancel each other out. Meanwhile, although removing the plugboards at the far ends would affect the output of the machine, it would not affect the characteristic. Mathematicians say the cycle type is the same under *conjugation*. This means the above diagram has the same characteristic as



This means you can work out the characteristic without knowing the plugboard.

A machine called a cyclometer was built, which was essentially two Enigma Machine wired together, with no plugboards and offset by three places - such as the diagram above. With help from the cyclometer, the Polish code breakers were able to catalogued the characteristics for all 17576 rotor positions and all 6 rotor orders, giving $6 \times 17576 = 105456$ characteristics. This worked under the assumption that the second and third rotors did not move.

Naturally the same could be done for letters in the second and fifth positions of the message indicators, as well as letters in the third and sixth positions. These characteristics were also indexed.

The job of cataloging took them a year. But by 1937, and with enough messages per day, the code breakers could use the indicators to find the characteristic and look up the rotor order and ground setting that produces it. Other settings are then deduced.

On November 1st 1937, the Germans changed the wiring of the reflector, and the card catalogue was useless. The cataloging process had to be done all over again. It took the Poles less than a year to complete the second card catalogue, but on September 15, 1938 the Germans changed their method of enciphering the keys, and the card catalogue and cyclometre were useless.

2.2 Bomba

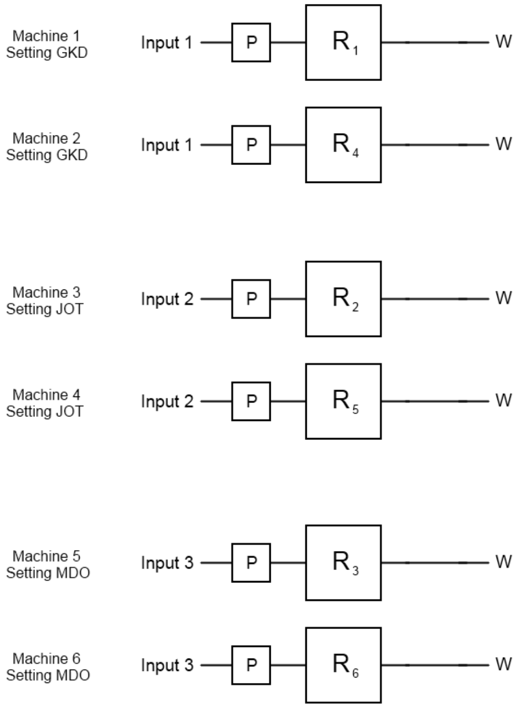
In 1938, the German army and air force stopped using a universal ground setting to encode the message setting. Instead, for each message, the operator picked three letters of his own to be used as a ground setting. The message setting would then be encrypted twice, as before, using the chosen ground setting. The chosen ground setting would be sent in plain at the beginning of the message, followed by the two encrypted message setting. Altogether, these nine letters were called the message indicator.

As before, the code breakers knew that the first and fourth letters were the same before encryption, as were the second and fifth letters, and the third and sixth letters. So the code breakers turned their attention to groupings where the first and fourth pair had been encrypted as the same letter, and similarly for the second and fifth pair, and third and sixth pair. These fixed points were called ‘females’.

A female in the first and fourth positions occurred on average once every 25 messages. The same holds true for the second and fifth pair, and the third and sixth pair. So the chances that any such female occurs is about 1 in 8.

Now they needed to find three messages indicators with the same female in the first position, second position and third position. For example, W in the following indicators; GKD WAVWHA, JOT IWABWN and MDO OTWYZW.

Let’s assume W is not paired on the plugboard. This is a reasonable assumption considering that, at this time, only 12 letters were paired on the plugboard. Now imagine six Enigma Machines with the correct settings. Set the rotors to the positions given in plain at the start of the indicators, and offset the rotors appropriately. Now, even though we don’t know the message setting being encrypted, we know that input 1 from the first indicator, input 2 from the second indicator, and input 3 from the third indicator all result in W.



The Polish now built a machine called Bomba, which simply reversed this idea. There were six Bomby machines, one for each rotor order. Each Bomba was three pairs of Enigma Machines (but without the plugboards) set up like the diagram above. Now, from a single input W, if the first pair gave the same output, the second pair gave the same output, and the third pair gave the same output, then the machine would stop. This would give you the rotor order and ring setting. The plugboard could then be found by hand.

However, the Bomby machines were often unreliable, in which case the code breakers would have to return to manual methods. Using the same idea of females in the message indicators, Zygalski designed large perforated cards. These cards were placed on top of each other, when the correct setting was found, one of the perforations of each sheet would line up, and light would shine from underneath. Like the Bomba, a set of cards were made for each of the six rotor orders and success revealed the rotor order and ring setting.

Using these techniques the Polish cryptographers were, by 1938, reading some 75% of intercepted German Radio transmissions enciphered using the Enigma machine. Then, in December of that year, the Germans added two new rotors, making five available, although only three were used in the machine at any one time. The Polish resources were severely strained, as now 60 sets of Zygalski sheets and 60 bomby would have been required. One month later, the number of plugboard leads were increased from 6 to 10. All this, and knowing from intercepts that their country was about to be invaded, persuaded the Poles to share their information with the French and British. On July 25th 1939, five weeks before Poland was invaded, a secret meeting in the Kabackie Woods near the town of Pyry was held. Here, the Poles handed over their complete solution to the German codes, their Enigma replicas and bomby to the British.

3 Alan Turing

Alan Turing was a talented young Cambridge mathematician. From September 1938, Turing had been working part-time with the Government Code and Cypher School (GCCS), the British code breaking organisation. On 4 September 1939, the day after the UK declared war on Germany, Turing reported for duty at Bletchley Park, the wartime station of GCCS. From then until November 1942, Turing was in charge of Hut 8, a section tasked with solving German naval Enigma messages.

In May 1937 the German Navy introduced a new system encoding the message setting. The Poles could not work out the new indicator system. They suspected that it was a bigram substitution but got no further. Meanwhile, the Polish methods of breaking Army and Air Force message were to be useless if the Germans changed these indicator procedures too, which they did in May 1940.

Turing started to work on a less fragile approach to the problem. During his time working on Enigma, Turing developed a more general approach to breaking the code using known plaintext, called ‘cribs’. He then mechanised this approach with the initial design of the Bombe Machine. Turing also correctly deduced the indicator procedure used by the German navy and developed a statistical procedure for making much more efficient use of the Bombes.

3.1 Deducing Naval Procedures

All branches of the German military used Enigma Machines throughout WWII. The army and air force called their machine Enigma I, while the navy called their machine Enigma M3. These machines were in fact the same, except the naval M3 labelled their rotors with the letters A to Z, rather than the numbers 1 to 26. The M3 also came with three further rotors, making a total of eight. However, what really separated the navy from their counterparts was their procedure for sending message indicators.

This naval procedure was introduced in May 1937. At the time, the Polish code breakers had been unable to work out what this new procedure might be. At the end of 1939 Alan Turing started working on the problem after noticing that no one else was really trying, meaning he could have the problem all to himself!

Turing deduced the following procedure:

- The Plugboard and Ground Setting changed every day, while the Rotor order and Ring Settings changed on odd numbered days.
- From a book, the operator then picks two triplets of letters, for example HLG and KQK. The first triplet was called the key indicator group, and the second was called the message indicator group.

  To send the indicator groups, they were written in two rows, offset by one place.

```
  H L G
K Q K
```

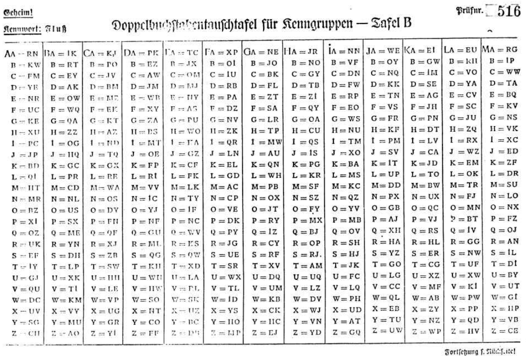
The operator picked two dummy letters to fill the spaces

$$\begin{matrix} A & H & L & G \\ K & Q & K & Z \end{matrix}$$

This made four vertical pairs (bigrams) of letters AK, HQ, LK, GZ.

Each bigram was encoded into a different pair using a bigram table. So if AK → BD, HQ → BJ, LK → EM, and GZ → EJ, the resulting message indicator is BDBJ EMEJ. This would be sent at the beginning and end of each message.

There were 9 bigram tables, to be used on different days. The bigram tables were self inverse, i.e. if AK becomes BD, then BD becomes AK. The receiving operator decoded the eight letters of the message indicator with the help of his bigram table.



Geheim!

Kennwort: Fluß

Doppelbuchſtabentauſchtafel für Kenngruppen — Tafel B

Prüfnr. 516

AA = RN	BA = IK	CA = KJ	DA = PK	EA = TC	FA = XP	GA = NE	HA = JR	IA = NN	JA = WE	KA = EI	LA = EU	MA = RG
B = KW	B = RT	B = PO	B = EZ	B = JX	B = OI	B = JO	B = NO	B = VF	B = OY	B = GW	B = KH	B = IP
C = FM	C = EY	C = JV	C = AW	C = OM	C = IU	C = BK	C = GY	C = DN	C = NQ	C = IM	C = VO	C = WW
D = YE	D = AK	D = BM	D = JM	D = MJ	D = RB	D = FL	D = TB	D = FW	D = KK	D = SE	D = YA	D = TA
E = NR	E = OW	E = MZ	E = WB	E = HY	E = PA	E = ZT	E = ZI	E = RP	E = TN	E = AG	E = CV	E = BQ
F = UC	F = WQ	F = EK	F = XY	F = AS	F = DZ	F = SA	F = QY	F = EO	F = VS	F = JH	F = SC	F = KV
G = KE	G = QA	G = KT	G = ZA	G = PU	G = NV	G = LR	G = OA	G = WS	G = FR	G = PN	G = JU	G = NS
H = XU	H = ZZ	H = AZ	H = BS	H = WO	H = ZK	H = TP	H = CU	H = NU	H = KF	H = DT	H = ZQ	H = VK
I = PC	I = OG	I = ND	I = MT	I = FA	I = QR	I = MW	I = QS	I = TM	I = PM	I = LV	I = RX	I = XC
J = JP	J = HQ	J = TQ	J = OE	J = GZ	J = LN	J = AU	J = IS	J = XO	J = SV	J = CA	J = WZ	J = ED
K = BD	K = GC	K = GX	K = FP	K = CF	K = EL	K = QN	K = PG	K = BA	K = IT	K = JD	K = EM	K = ZF
L = QI	L = PR	L = RE	L = RI	L = FK	L = GD	L = WH	L = KR	L = MS	L = UP	L = TO	L = OK	L = DR
M = HT	M = CD	M = WA	M = VV	M = LK	M = AC	M = PB	M = SF	M = KC	M = DD	M = BW	M = TR	M = SU
N = MR	N = NL	N = OS	N = IC	N = TY	N = CP	N = OX	N = SZ	N = QZ	N = PX	N = UX	N = FJ	N = LO
O = BZ	O = US	O = DY	O = YJ	O = IF	O = VE	O = JT	O = FY	O = YV	O = GB	O = QC	O = MN	O = NX
P = XI	P = SX	P = FH	P = NF	P = NC	P = DK	P = RY	P = MX	P = MB	P = AJ	P = VJ	P = BT	P = FZ
Q = OZ	Q = ME	Q = QF	Q = GU	Q = WV	Q = PY	Q = IZ	Q = BJ	Q = OV	Q = XH	Q = RS	Q = IV	Q = OJ
R = UK	R = YN	R = XJ	R = ML	R = KS	R = JG	R = CY	R = OP	R = SH	R = HA	R = HL	R = GG	R = AN
S = EF	S = DH	S = ZB	S = QG	S = QW	S = UE	S = RF	S = RJ.	S = HJ	S = YZ	S = ER	S = NW	S = IL
T = IY	T = LP	T = SW	T = KH	T = XD	T = SR	T = XV	T = AM	T = JK	T = GO	T = CG	T = UF	T = DI
U = GJ	U = XK	U = HU	U = WH	U = LA	U = WX	U = DQ	U = UQ	U = FC	U = LG	U = XZ	U = XW	U = BY
V = QU	V = TI	V = LE	V = HW	V = PL	V = TL	V = UM	V = LZ	V = LQ	V = CC	V = MF	V = KI	V = UT
W = DC	W = KM	W = VP	W = SO	W = SK	W = ID	W = KB	W = DV	W = PH	W = QL	W = AB	W = PW	W = GI
X = UV	X = VY	X = UG	X = NT	X = HZ	X = YS	X = CK	X = WJ	X = UD	X = EB	X = ZY	X = PP	X = HP
Y = SG	Y = MU	Y = GR	Y = CO	Y = BC	Y = HO	Y = HC	Y = VN	Y = AT	Y = TU	Y = NZ	Y = QD	Y = VB
Z = CH	Z = AO	Z = YI	Z = FF	Z = DB	Z = MP	Z = EJ	Z = YD	Z = GQ	Z = UW	Z = WP	Z = HV	Z = CE

Fortſetzung ſ. Rückſeite!

- The key indicator group, HLG, was associated with the daily setting being used by the sender. This confirmed that both sender and receiver were using the same settings. Finally, the sender and receiver would both set their Enigma Machines to the ground setting and type in the message indicator, for example KQK → IYS. The result, IYS, would be then used as the message key for the rest of the message. Note, the message key was not the message indicator itself.

By studying the indicators of naval messages that had already been deciphered by other methods, Turing, and the other code breakers of Hut 8 at Bletchley Park, were able to partially piece together the bigram tables.

A breakthrough was achieved after the disguised armed trawler Polares was seized by HMS Griffin in the North Sea on 26 April 1940. The Germans did not have time to destroy all their cryptographic documents, and the captured material revealed the precise form of the indicating system, confirming what Turing had deduced. The bigram tables themselves were not part of the capture, but the settings lists found allowed more messages to be broken, helping with the further reconstruction of the bigram tables.

3.2 Banburismus

By the start of the war, the naval Enigma Machine was using three rotors from a choice of eight, giving a total of 336 rotor orders. Turing developed a statistical method using a measure called that he called the *'Ban'* to help reduce this number for the code breakers.

Using the bigram tables, the code breakers could now recover the message indicator from the beginning of naval messages. Naval Enigma operators were told to set their machine to the day's ground setting, type in the message indicator using the Enigma Machine itself, and the result would be used as the message setting. For example, the Enigma Machine might turn the message indicator KQK into the message setting IYS.

The code breakers looked for two messages that had the first two letters of their message indicators in common, for example KQK and KQB. Since both indicators are transformed using the same ground setting, it is likely the resulting message settings will also have the first two letters in common, for example IYS and IYV. In which case, shifting the second message three places to the right will put these messages in synch. In other words the letters of each message are being encrypted using the same rotor position.

```
indicator KQK: U X C Y B G D S L V W B D J L K W I P E H V Y Q D T R Q X K E E S Q S S P Z
indicator KQB:       F N S D S C C W V I P E M W F L E S C Y S P V R X M C F Q S Q D V U L B T
```

The parts of the two messages that overlapped in this way were said to be *'in depth'*.

However, the code breaker does not yet know that a shift of 3 will put these messages in depth. There will be 50 alignments to check (25 places to the left and 25 places to the right). To determine whether two messages are in depth we look for matching letters.

Given two letters, the probability that the two letters match is

$$P(\text{two letters match}) = P(\text{A})^2 + P(\text{B})^2 + \cdots + P(\text{Z})^2,$$

where $P(x)$ is the probability (or frequency) of the letter x.

If all letters are equally likely, then the probability of a match is $26 \times (1/26)^2 = 1/26$. Using letter frequencies for English the probability is about 1/15. In German, the probability is about 1/13. In German naval messages however, this probability was determined to be about 1/17.

If two messages are in depth, two letters that matched in the plaintexts will be encrypted as the same letter in the ciphertext. So matches will occur with the same frequency as they do in the plaintexts, with a rate of about 1 in 17. However, if the messages are not in depth, then the two ciphertexts will compare as if they were random, giving a rate of about 1 in 26.

In the example above, shifting the second message 3 places to the right results in nine repeats, including two bigrams.

Counting the number of matching pairs involved strips of paper several metres wide, onto which were up to 50 alphabets printed vertically. The letters of each message were punched with a hole in successive columns. The two sheets were then slid one against the other above a lit background to reproduce all 50 alignments. At each position, matching letters showed up as visibly matching holes, to be counted and recorded for that alignment. As the sheets were printed in Banbury they were called Banburies, and the whole process was called Banburismus.

Turing developed a scoring system to determine whether two messages were in depth. Each match increased the score by about +4, while runs of successive matches increased the score even more. A score over +34 meant the two messages were more likely to be in depth than not in depth. For example, the two messages above, offset 3 places, results in a score of +52.

The theory behind the scoring was based on the principle of Bayes' Theorem for conditional probabilities, which states:

$$P(A|B) = \frac{P(B|A)}{P(B)} \cdot P(A),$$

where $P(A|B)$ may be read as 'the probability of A given B'.

We achieve Bayes' Rule by dividing two instances of Bayes' Theorem to get:

$$\frac{P(A_1|B)}{P(A_2|B)} = \frac{P(B|A_1)}{P(B|A_2)} \cdot \frac{P(A_1)}{P(A_2)}.$$

In the special case where $A_2 = \bar{A}_1$, the complement of A_1, we get the odds form of Bayes' Theorem,

$$O(A|B) = \frac{P(B|A)}{P(B|\bar{A})} \cdot O(A),$$

where $O(A) = P(A)/P(\bar{A})$ are the odds of A.

In our case, assuming Enigma encrypts letters as equally likely, the odds two messages are in depth given a match is,

$$O(\text{in depth}|\text{match}) = \frac{P(\text{match}|\text{in depth})}{P(\text{match}|\text{not in depth})} \cdot O(\text{in depth}) = \frac{(1/17)}{(1/26)} \cdot O(\text{in depth}) = \frac{26}{17} \cdot O(\text{in depth}).$$

This means each match increases the odds that the message are in depth by a factor of 26/17.

On the other hand, the odds two messages are in depth given a non-match is,

$$O(\text{in depth}|\text{non-match}) = \frac{P(\text{non-match}|\text{in depth})}{P(\text{non-match}|\text{not in depth})} \cdot O(\text{in depth}) = \frac{1-(1/17)}{1-(1/26)} \cdot O(\text{in depth}) = \frac{416}{425} \cdot O(\text{in depth}).$$

This means each match decreases the odds that the message are in depth by a factor of 416/425.

Moreover, if we assume successive events along the overlap of the two messages are independent, their factors can be multiplied together to give a composite factor for the alignment as a whole.

For example, imagine testing two messages with an overlap of 32 letters where 7 pairs matched and 25 that did not. The composite Bayes factor for this is $(26/17)^7 \times (416/425)^{25} = 11.5$. This information increases the odds from 1 to 49 (because there were 50 equally likely alignments before the event) to 1 to 4.3.

To simplify the handling of the many Bayes' factors, logarithms are used to turn the multiplicative factors into additive scores called '*Bans*'. Each match therefore increases the score by $\log_{10}(26/17) = 0.18$ Bans, while each non-match decreases the score by $\log_{10}(416/425) = 0.0093$ Bans. A Ban is equal to 10 deciBans (dB), or 20 halfdeciBans (hdB). The idea of deciBans is analogous to decibels. Each match therefore increases the score by 3.7 hdB, while each non-match decreases the score by 0.19 hdB.

The above assumed each match was an independent event, allowing us to multiply their effect. However, this is not quite the case. For example, a run of successive matches might mean you have found a short

German word and by coincidence that word is in synch in the two messages. In this case, the probability of a repeating bigram is greater than the product of the probabilities of the two repeating letters taken separately. A recently-released paper which Alan Turing wrote at the time shows him analysing the impact of repeating bigrams, trigrams and up to hepta-grams. Hut 8 used this elaboration, but this paper does not pursue it farther.

Banburists would search for shifts with a score greater than +34 hdB, this corresponded to odds of being in depth better than 1 in 1, i.e. the messages were more likely to be in depth than not in depth. In the example above we had two messages sent with the message indicators KQK and KQB. A shift of 3 places to the right resulted in 9 matches, two of which were bigrams (the probability of a repeating bigram is greater than the product of the probabilities of the two repeating letters taken separately), giving a score of +52 hdB. This corresponds to a Bayes' factor of about 400, and increases the odds of being in depth from 1 to 49, to about 8 to 1. In other words, the probability of being in depth rises from 1/50 to 8/9. This success would be written as K+3=B.

In the same way, Banburists use other messages to find other relations such as Q-2=X, X-4=H, H-2=B and K+8=X. These could then be put together to form a chain:

K - - B - H - - - X - Q

The Banburist would then try to determine the ground setting alphabet of the third rotor, i.e. the effect of enciphering each of the 26 letters of the alphabet. For example, we believe B and K+3 have the same output, so if the output of K is A, then the output of B is D. Shift the chain through all 26 positions and reject any positions that result in an invalid Enigma cipher, i.e. does not form 13 pairs.

output:	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
	K	-	-	B	-	H	-	-	-	X	-	Q															possible
		K	-	-	B	-	H	-	-	-	X	-	Q														impossible, forms pairs (KB) and (BE)
			K	-	-	B	-	H	-	-	-	X	-	Q													impossible, since H becomes H
				K	-	-	B	-	H	-	-	-	X	-	Q												possible
																											etc

The code breakers would now try fitting other letter chains, such that the chains no letters in common, and that letters from each chain were in different positions. Eventually they will hope to be left with just one candidate, maybe looking like this:

output:	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
	-	-	X	-	Q															K	-	-	B	-	H	-
	F	-	-	-	-	A	-	-	D	-	-	-	O													
										N	U	P														

Such an alphabet forces the conclusion that the third rotor is in fact Rotor I. This is because Rotor II would have caused a mid-wheel turnover as it stepped from E to F, yet that's in the middle of the span of the letter-chain F - - - - A - - D - - - O. Likewise, all the other possible mid-wheel turnovers are precluded. Rotor I does its turnover between Q and R, and that's the only part of the alphabet not spanned by a chain.

Having the rotors turn over in different places made it possible to distinguish the wheels by Banburismus and reduce the number of rotor orders to be tried. The additional naval rotors VI, VII and VIII had the same turnover points and so were indistinguishable from one another, a great nuisance to the Banburist.

Once the end rotor is identified, these same principles can be extended to handle the middle rotor. This was more complex, since this meant comparing message indicators with only the first letter in common, and checking 1350 overlaps. This large task was aided with help of machines. Banburismus reduced the number of wheel orders to be checked from 336 to somewhere between 3 and 90, and was seen as a great intellectual game to the code breakers at Bletchley Park.

The first day to be broken was 8 May 1940 by Hugh Foss, thereafter celebrated as “Foss’s Day”. The task took until November that year, by which time the intelligence was very out of date, but it did show that Banburismus could work. It also allowed much more of the bigram tables to be reconstructed. By the end of 1940, much of the theory of the Banburismus scoring system had been worked out. It was eventually killed in 1943 by the rapidly increasing number of Bombes which made it unnecessary to spend much time and labour in reducing the number of wheel orders to be run - it was simpler and quicker to run all wheel orders.

3.3 The Bombe Machine

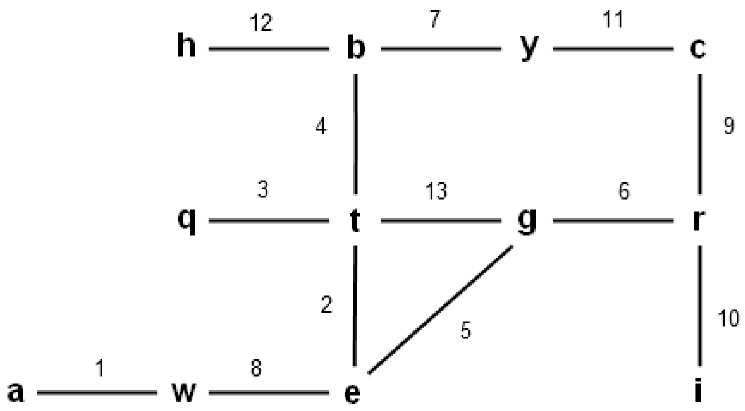
Up until now, the Polish and British code breakers had been exploiting a flaw in the procedures of the army and air force when sending message indicators. The repetition of the encryption of the message setting was a vital clue in breaking the code. Then, in May 1940, the procedure changed. Message settings were no longer to be repeated in this manner.

A more general approach was needed to break the code, so a way was found to exploit a more inherent flaw of the machine. Since each setting of the Enigma Machine turned the 26 letters of the alphabet into 13 pairs, as a consequence no letter can be paired with itself. The cryptanalysts could then take some common phrase, slide it underneath the ciphertext and find where that phrase might fit in the ciphertext; two letters were *not* allowed to match. These phrases were known as *cribs*.

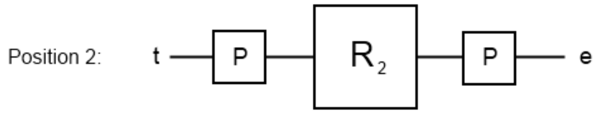
For example, there is only one place the phrase ‘wetterbericht’ (‘weather report’) may fit in the ciphertext below. Moving the crib to the left or right will result in two letters coinciding.

	1	2	3	4	5	6	7	8	9	10	11	12	13	
··· j x	a	t	q	b	g	g	y	w	c	r	y	b	g	d t ···
	w	e	t	t	e	r	b	e	r	i	c	h	t	

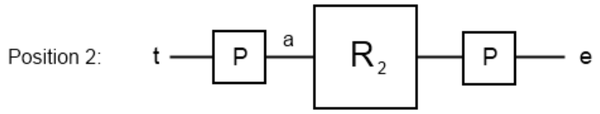
We can draw a diagram describing the relations between letters, where two letters are joined by a labelled edge if those two letters form a pair at that position. This diagram is called a *menu*. The menu for the crib above is:



Let's assume this portion of the ciphertext does correspond to our crib phrase. Then we can see that, in position 2, 't' becomes 'e'. This is the combined effect of the plugboard, the rotors in position 2, followed by the plugboard again:



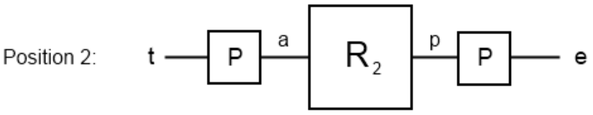
Remember, because the Enigma Machine acts as a product of transpositions, this diagram also works in the opposite direction.Alan Turing realised that, for a given rotor setting, it is possible to deduce certain plugboard settings. For example, let's assume (ta) is a pair on the plugboard. So after the first use of the plugboard, input 't' becomes 'a'.



This is then followed by the rotors in the second position. Starting from some initial position, below are the complete outputs of 13 successive monoalphabetic ciphers from an enigma machine without its plugboard, i.e. R_i, $1 \leqslant i \leqslant 13$. Notice, these ciphers are still products of transpositions.

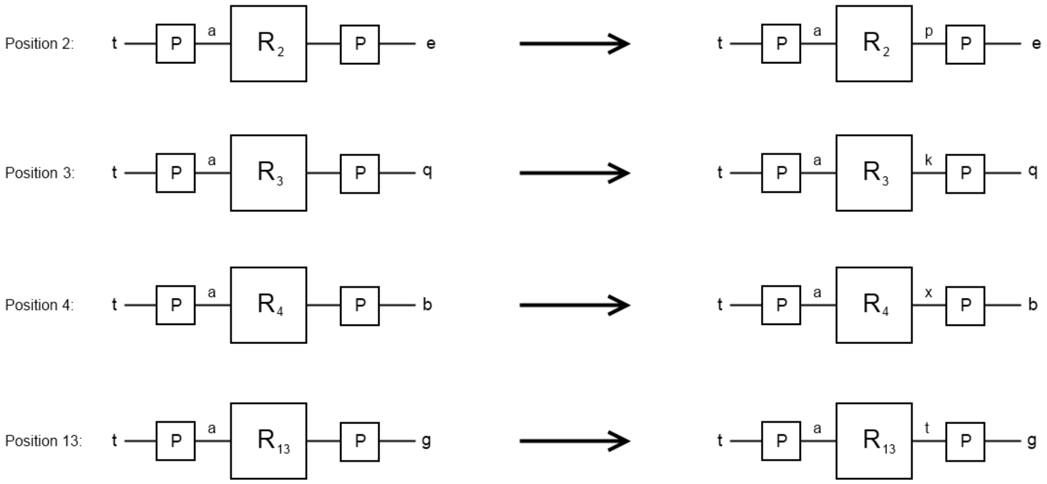
input:	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
output 1:	j	f	q	x	h	b	s	e	k	a	i	y	z	t	v	u	c	w	g	n	p	o	r	d	l	m
output 2:	p	n	s	k	u	z	o	w	v	l	d	j	r	b	g	a	t	m	c	q	e	i	h	y	x	f
output 3:	k	d	p	b	i	q	t	m	e	o	a	n	h	l	j	c	f	t	g	r	x	z	y	u	w	v
output 4:	x	o	d	c	h	z	l	e	p	y	u	g	q	w	b	i	m	v	t	s	k	r	n	a	j	f
output 5:	o	f	y	e	d	b	z	x	l	w	q	i	n	m	a	s	k	v	p	u	t	r	j	h	c	g
output 6:	v	w	p	t	m	x	k	u	o	l	g	j	e	z	i	c	r	q	y	d	k	a	b	f	s	n
output 7:	d	r	t	a	g	u	e	m	z	k	j	x	i	v	y	w	s	b	q	c	f	n	p	l	o	i
output 8:	v	z	e	j	c	q	u	n	l	d	y	i	r	h	w	x	f	n	t	s	g	a	o	p	k	b
output 9:	h	x	i	z	g	p	e	a	c	y	o	v	s	t	k	f	w	u	m	n	r	l	q	b	j	d
output 10:	h	v	x	m	z	i	k	a	f	s	g	w	d	q	u	r	n	p	j	y	o	b	l	c	t	e
output 11:	o	w	m	p	y	l	t	z	k	x	i	g	c	u	a	d	r	q	v	f	n	s	b	j	e	h
output 12:	r	u	z	l	j	y	i	t	f	e	m	d	k	x	q	r	o	a	p	h	b	w	v	n	f	c
output 13:	t	l	s	p	o	h	x	f	q	k	j	b	w	r	e	d	i	n	c	a	y	z	m	g	u	v

This works under the assumption that the second and third rotors do not move. The probability that the second rotor will move for a crib of length n is $n/26$, so cribs were kept to 10-14 letters. From this table we see the input ‘a’ in the second position becomes ‘p’.



This leaves the second use of the plugboard. However, since we know the final output is ‘e’, to make this diagram work we can deduce that (pe) is the other plugboard setting.

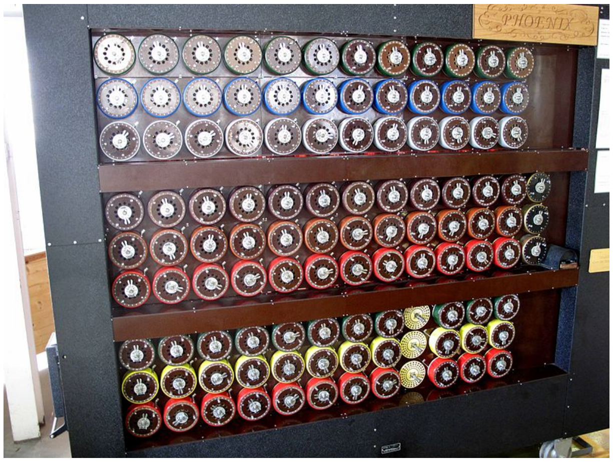
In the same way, by considering what happens in the third, fourth and thirteenth positions we get three further plugboard settings, namely (kq), (xb) and (tg).



But here we have a contradiction, (ta) and (tg) cannot both be pairs on the plugboard. So our original assumption that (ta) is paired on the plugboard is wrong. If we can show all possible pairs for a given letter like ‘t’ lead to a contradiction, then our rotor setting must be incorrect, and we should try another.

Turing’s brainwave was to realise that, due to the way the enigma works as a self-inverse product of transpositions, if we had assumed (tg) was a plugboard setting it would equally imply (ta), and so this assumption is equally false. The settings (pe), (kq) and (xb) may also be eliminated for the same reason. So, once we find a contradiction, we can eliminate all other plugboard settings derived from our assumption as ‘fruit of a poison tree’.

Turing used this principle at Bletchley Park in his ‘Bombe Machine’. The Bombe acted as simultaneous Enigma Machines connected in series, logical implications from one ‘enigma machine’ could then feed into the others, which were set at different positions with their input and output determined by the menu.

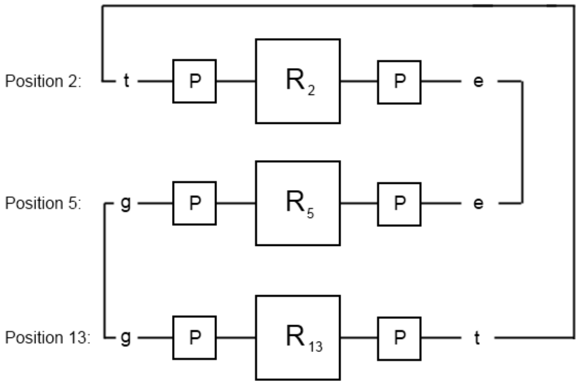


In the example above, the rotors of four of these ‘enigma machines’ would be set to position 2, 3, 4 and 13. An electrical current applied to the initial plugboard assumption of (ta) would result in a current passing through (pe), (kq), (xb) and (tg). This feedback was detected allowing these plugboard settings to be rejected. The chain of logical conclusions were deduced almost instantaneously by the electrical circuits.

After making an initial plugboard assumption, either;

- the current flows through all plugs for a given letter. The rotor position must be incorrect and we should try another;
- the current flows through one plug for a given letter, or all but one plug, and the machine stops;
- the current flows through some other number of plugs, meaning further checking is required;
- we get a false stop due to coincidence.

To reduce the problems of false stops and results that required further checking, the cryptographers used menus that contained cycles, this made inconclusive situations less likely and reduced the number of false stops. In our example above, using the menu we see a cycle connecting the letters t, e, and g. The correct rotor set-up will then make the following loop possible:



Since the Enigma Machine is the combined result of the plugboard, followed by the rotors, then followed by the plugboard again, the ouput for a given rotor position i may be written as

$$E_i = PR_iP.$$

Applying successive Enigma functions we get the relation

$$t = E_{13}E_5E_2(t).$$

Because the plugboard is a product of transpositions it is self-inverse. This means if we write $E_i = PR_iP$ consecutive plugboards will cancel each other out, and this relation can be rewritten as

$$P(t) = R_{13}R_5R_2(P(t)).$$

This calculation still requires an assumption about the plugboard, but it is a much tougher condition to satisfy and may be used to find contradictions as before. We may produce similar conditions from other cycles from the menu. The probability of satisfying k such cycles at random, causing a false stop, is $(1/26)^k$.

When the machine stops the rotor and plugboard settings are recorded. The remaining settings can be deduced by hand. Gordon Welchman later added a 'diagonal board' to the Bombe, increasing the machine's connectivity and speed. Now the Bombe Machine could check the logical implications of all 17576 rotor positions in 20 minutes.

Hut 8 was served by a large staff ranging from the brilliant individual cryptanalysts to the clerical staff and the WRNS who kept the bombes going throughout the 24 hours. The work called for organisation and management of the highest quality. Neither was up Alan Turing's street. Even while Turing was nominally Head of the Hut, it was Hugh Alexander who supplied them: he was both a brilliant cryptanalyst and a brilliant manager of cryptanalysts and staff at every level. When Turing moved to other work in November 1942, Alexander became Head of the Hut in name as well as in reality.

With all this in place, the bombe machines became a highly successful attack on the Enigma throughout the Second World War.

4 Other Methods

Finally, let's finish with some other methods, clues and operator mistakes that helped the break the Enigma code.

- ANX crib: Like the code breakers of Bletchley Park, the Polish Cipher Bureau had also used cribs. For example, messages starting with 'ANX' (German for 'to' followed by 'X' as a spacer).
- FORTYWEEPYWEEPY cribs: If a message was a continuation (a FORT) of another, it would start with FORT followed by the time of origin of the first message. Its name came from one such message which was a continuation of message 2330. Numerals at this time were read off the top row of the key board and inserted between Ys, so 2=W, and 3=E, and 0=P, and the continuation of message 2330 was written as FORTYWEEPYWEEPY.
- EINSing: Later, a captured German revealed under interrogation that Enigma operators had been instructed to encode numbers by spelling them out. Alan Turing reviewed decrypted messages and determined that the number 'eins' (German for 'one') appeared in 90% of messages. A catalogue was produced of the word 'eins' encrypted by every possible rotor order and starting position. All groups of four letters in a given message were checked using the Eins Catalogue. Once a match had been found, the rest of the message could be checked using this Enigma setting.
- Gardening: When cribs were lacking, Bletchley Park would sometimes ask the Royal Air Force to 'seed' a particular area in the North Sea with mines. The Enigma messages that were soon sent out would most likely contain the name of the area or the harbour threatened by the mines. This process came to be known as gardening.
- Hervil Tip: After rotating the alphabet labels to the prescribed ring setting, the lazy army or air force operator might not turn the rotors by more than a few places when selecting the first part of the indicator (which was sent in plain). So, especially in the mornings when the machine had just been set up for the day, these indicators turned out to be a clue to that day's ring setting. This sloppy practice was anticipated by John Herivel soon after his arrival at Bletchley Park in January 1940.
- Six letter words: At the beginning of each message, it was German army and air force procedure to allow the operator to pick two triplet of letters, the first as a ground setting and the second as a message setting. The ground setting would be sent in plain, followed by the encrypted message setting. The operator should pick six random letters, but an unwise operator may pick a six letter word. So if the first three letters, sent in plain, were BER, then the encrypted three letters that followed may have been LIN (BERLIN). Similarly, LON would be followed by DON, and HIT by LER. Naval procedures did not allow such free choice.
- Restrictions on the rotor orders: A rotor order was not allowed to be repeated on a monthly setting sheet. This meant that when the Enigma settings were being found on a regular basis, certain rotor orders could be excluded if they had already appeared that month. Some networks stipulated that no rotor should be in the same slot as it had been the previous day, this again reduced the number of rotor orders that had to be tried.

- Restrictions on the plugboard: The Air Force stipulated that no letter should be connected on the plugboard to its neighbour in the alphabet. This reduced the problem of identifying the plugboard connections and was automated in some Bombes with a Consecutive Stecker Knock-Out (CKSO) device.

- Parkerismus: Some of the columns of rotor orders, ring settings or plugboard connections were reused from previous months. The resulting analytical short-cut was christened at Bletchley Park Parkerismus after Reg Parker, who had, through his meticulous record-keeping, spotted this phenomenon.

After the war, a considerable number of German cryptographic personnel were detained by the Americans. Among the things they learned was that German cryptographers, at least, understood very well that Enigma messages might be read, and that Enigma was not unbreakable. They just found it impossible to imagine anyone going to the immense effort required. When Abwehr personnel who had worked on cryptography were interned at Rosenheim around May 1945, they were not at all surprised that Enigma had been broken, only that someone had mustered all the resources in time to actually do it.

5 Other rotor cipher machines

The first rotor cipher machine was developed for the Dutch navy in 1915 by R. P. C. Spengler and Th. van Hengel. The navy decided not to adopt the idea, but prevented the it was being patented by the original inventors.

In the United States Edward Hugh Hebern built a rotor machine using a single rotor in 1917. He sold a small number of machines to the US Navy in 1931. Unknown to Hebern, William F. Friedman of the US Army's SIS promptly demonstrated a flaw in the system that allowed it to be cracked with enough work.

Other inventors of similar machines were Dutchman Hugo Koch and Swede Arvid Gerhard Damm who both patented ideas in 1919. However it was Arthur Scherbius who went on to have the most success. Scherbius patented the Enigma Machine in 1918, with the first commercial version of the machine, known as Enigma A, going on sale in 1923. The military version of Enigma, known as Enigma I, was created in 1930. The military Enigma used different rotors than the commerical version, it also had the plugboard at the front of the machine. Also, instead of printing on paper, Enigma I used a lampboard. Not only did this make the machine cheaper, but also made it eight times lighter than the commercial machine.

I am a paragraph that descibed JN25 and Red/Purple japanese cipher machines.

From 1937 the British had their own cipher machine known as 'Typex'. This machine, borrowing heavily from the Enigma design, would use 5 rotors from a choice of 8, with the addition that each rotor could be also be placed in reverse. The two rightmost rotors did not move and were the equivalent to Enigma's plugboard - except this componant is not self-inverse. The machine as a whole was self-inverse due to containing an Enigma style reflector. The three rotors on the left moved in a similar way to Enigma, but moved more frequently, with each rotor having between 5 to 9 kickover points per cycle. Typex was still vulnerable to Turing's crib attack, but the frequency of the kickover points would require far more ciphertext. The German cryptographers concluded that Typex was more secure than Enigma, which they already believed to be unbreakable and did not make any serious attempts to break the British cipher.

From 1937 the US navy started using their own Enigma style rotor cipher machine called 'Electric Code Machine Mark II'. By 1941 this machine had also been adopted by the US army and called SIGABA. This machine had 15 rotors, five of which were cipher rotors like Enigma. The remaining rotors were responsible for creating a complex pseudorandom stepping of the cipher rotors. Even with the original plaintext it would be difficult to work out the settings. The SIGABA cipher was never broken, however it was a large, heavy and fragile machine, as well as being difficult to operate. Unlike the Enigma, SIGABA was not practical for field work, and other communication methods needed to be used such as the famous Navajo code talkers.

From 1943, both Typex and SIGABA cipher machines could be modified to produce the 'Combined Cipher Machine' allowing encrypted communication between Britain and the US.

I am a paragraph that describes Lorenz and Geheischreiber machines, and in brief Bill Tutte, Tommy Flowers and Colossus.