How to decrypt a Lorenz SZ42 message using Virtual Colossus 3D

A tutorial on decrypting a German Lorenz SZ42 message using Virtual Colossus 3D
https://virtualcolossus.co.uk

This document will take you through an example process used on Colossus to get the start positions of a received message. There wasn't a set program which would run to work out the settings, it was a matter of following a "menu" of different algorithms and sometimes required knowledge of what worked on a particular radio link.

Colossus was not able to run a sequence of algorithms or use the results of previous calculations (like we would expect from a computer today), each run required a decision by the operator or code breaker assigned to that machine on which operation would be the next best to run.

Once the start positions of each of the twelve rotors are calculated, we can use a Lorenz machine to try to decipher the final message.

Wheel breaking

We will assume that we have already broken the wheel patterns for this message. Wheel breaking meant working out the pin settings of all 501 pins on each of the twelve Lorenz cipher wheels. This was generally done using a few methods including from messages in depth (where several messages were received enciphered with the same key) in a process devised by Alan Turing called **Turingery**. Later, a new process called rectangling allowed wheel breaking potentially using a single long message. Colossus could help with rectangling to assist with wheel breaking, and Colossus 6 was used almost exclusively for this, but in general, wheel breaking was done by hand methods.

Breaking the wheel patterns has given us the current pin settings for each of the rotors but not the start positions that the Lorenz operator set for this message. The possible number of different start positions for each message (not including changing the pins on each wheel) is the product of the number of positions for each rotor, which calculates as 43 × 47 × 51 × 53 × 59 × 37 × 61 × 41 × 31 × 29 × 26 × 23 = 1.6034 × 1019. If you could manage to test a million new start positions every second, it would still take you over 500,000 years to check this one message! Fortunately, Bill Tutte and the Bletchley Park code breakers calculated a way to possibly break into the message and find the start positions statistically.

Finding your way

The first thing to do is make yourself aware of where each of the racks and panels we will be referring to are located on Colossus. Looking at the front of Colossus, from left to right, try to find the following racks

- **The C Rack (Counters)**: This includes the Set Totals Panel (the white box with dials on).
- **Rectangling Rack**: The electronic typewriter in front is used as the primary output. (The controls on this panel are not used in this simulation).
- **Relay Rack**: This includes a number of panels: The counter display; The control panel (the two rows of toggle switches) and the setting jacks panel (the twelve rows of jack plugs).
- **The K Rack (Keys)**: This large switch panel on the K Rack is called the Q Panel and is used to set which logic and conditional "program" will be used on a run. Just below this is an angled panel holding several rows of plugs which is the Wheel-breaking panel
- **The J Rack (Jacks)**: This has several panels which are mostly used to route which data is required to be calculated to the processing Q panel.
- **Bedstead**: The large rack sticking out from the end which holds up to two cipher tapes to be processed.

Setting Colossus inputs

The place to start is at the "bedstead", the frame sticking out at the far right, so named as it resembled the metal frame of a bed stood on end. This is where the cipher tape, punched in **ITA2 code** on a 5 hole (sometimes called impulses or bits in modern terms) tape can be mounted. Colossus has two bedsteads (called Near and Far) so one can be running while the other is having a new tape loaded.

Each tape was run across various pulleys and wheels to make it taut then glued start to end in a loop. From default, Virtual Colossus has an example cipher tape loaded and running, but if not, click on the tape on the "Near" to Colossus bedstead and choose Cipher 3 under the Knock Holt section. If you have switched off the motor and the wheels are not spinning, click the switch (which looks like a British 1940s light switch - because it is!) about half way up the side of the bedstead to start it up then wait while the tape motors spins up before returning to this tutorial.

On the J Rack (to the left of the bedstead) is a panel with several large ganged black switches. The first switch to check is one which is marked Near and Far, this tells Colossus which of the bedstead tapes is to be the current input.

Make sure the Tape Select switch is set up to the Near setting. To change a switch, move your mouse over it until you see the hand cursor. Left click and drag up or down while holding down the mouse button then release to set.

Running a few simple counts

A good way to start on Colossus is to simply find out how many characters are in our message. To do this, we need to firstly set the direct Z input (the paper tape impulses) to be read into the Q bus which is where we can set our logic and conditional calculations.

Set the Z input switch up to the direct input Z.

This simply passes all characters from the tape to the Q bus for processing.

Now look at the top section of the Q Panel on the K Rack (the large switching panel). This section has ten rows, on each row there are five 3-way switches (black), one for each impulse, each of which can be set to either dot (up), cross (down) or either (middle). This is the Q bus and the data from the five input impulses are available on each column. At the right are five switches (grey) numbered 1 to 5 which are to set which of Colossus' five counters to store into. To count all the characters passed to the Q bus, we can leave the black selection switches to their middle position and set counter 1 as the result

Set counter 1 switch down

This program simply sets no conditions to check and assigns the result to the first of Colossus' five counters (i.e., match any character input and count it).

On the Relay Rack is the display panel, the top section marked from a to e are the counters. Each counter display is split into four decades: 1000s; 100s; 10s & 1s, so the value of 5667 characters should be shown on counter "a" as 5 thousands, 6 hundreds, 6 tens and 7 ones.

Another simple count is to find out how many of a specific character are on the cipher tape. To find how many letter B characters are in the data, we firstly need to know the ITA2 code for letter B which is X..XX (see **The Lorenz Machine**).

To set this on the top row of black switches on the Q Panel (K Rack), means putting switches **1**, **4** and **5** down to match a cross setting (a hole on the tape) and switches **2** and **3** up to match a dot. Confirm that you still have the first grey counter switch down and then check the result on counter a on the display panel, if you've done everything correctly, you should see the number of B characters is 163.

Looking back at the Q Panel, the yellow switch to the left of the five counter switches is the "NOT" switch. It inverts the true or false from the comparison switches on the left before the result gets counted. Therefore, if we switch the top row yellow **NOT switch** down, we should get the total number of characters that are not B, a total of 5504 (total of 5667 - 163 = 5504).

Bill Tutte's 1+2=. Break In

Our first real codebreaking task is to find the start positions of the first two Chi wheels. Bill Tutte calculated an equation by using the delta(Δ) of the ciphertext added to the delta of the Chi wheels. The delta is the change in bit pattern between successive character bits, it is zero if there is no change, but one if the bits are not the same. This means that if two successive characters are the same on each of their five bits, the Delta is all zeros which is the "/" or null character. Natural language has lots of double letters plus teleprinter shifts and punctuation which can also contribute significantly to the number of null characters in the delta input. He found that if you took an incorrect start position of the Chi wheels and calculated how many dots resulted, the random nature of the characters meant it would end up being roughly 50% crosses and 50% dots (on our example, 5667/2 = 2834). Due to the nature of German teleprinter text and the stutter of the Psi rotors, he found that if you calculated the same result where the start position was correct, and therefore the final calculated plaintext was in readable German, there was a 55% chance of the result being a dot rather than a cross. While this is not a huge change, statistically, over a whole message of a few thousand characters, this difference was visible.

He also found that it was possible to do this calculation on just two of the Chi rotors at a time and still see a statistical bump which cut down hugely on the number of start positions as the calculations could be done in smaller sections.

We can therefore run his equation on Colossus, calculating the number of dots for each start position of Chi rotor 1 (41 positions) & Chi rotor 2 (31 positions) in turn and find which start position has the highest number of dots. This means we only need to check a relatively small 1271 start positions which should then give us the most likely start position for these two wheels. While this is still a large number of calculations for a human to do (two deltas * the number of characters in the tape * 41 * 31), a computer can calculate and count this with relative ease.

Adding in the Chi wheel patterns

Currently, we are only sending the direct Z (tape input) to the Q Bus, but we want to add the Delta Z (Tape) and the Delta X (Chi) all the way down the tape, this is changed on the J Panel Q Bus Switches. The three top ones on the right are where we assign which of our inputs we wish to assign to the Q bus. We have three inputs to choose from and each can be set to input the direct values or to the delta of the input. The QBus Z input is the data from the tape, the QBus X is the input from the current Chi wheels and QBus Ψ is the input from the current Psi wheels.

Set the **Z input switch** to ΔZ
Set the **X input switch** to ΔX.

This means we are already adding the five separate impulses of ΔZ+ΔX and sending that to our calculation bus.

The next thing we want to do is to set which "pattern" of Chi & Psi wheel lugs are set for this particular message. These will have been worked out in advance by the **Testery**, the section using hand method to decrypt Tunny traffic. (German encrypted teleprinter traffic was codenamed Fish at Bletchley Park and Tunny, meaning tuna fish, was the codename given to the Lorenz traffic and machine).

We are deciphering a message which was encrypted by a pattern which we've called the KH pattern. Find the two switches just below the Near/Far switch which sets either the Bream (a) or KH (b) setting. The one with three ganged switches chooses the Chi/Psi setting while the single switch sets the Mu setting.

Set the Chi/Psi Pattern select switch down to KH (b).
Set the Mu Pattern select switch down to KH (b).

Using the addition section

For this calculation, we are going to be using the bottom half of the Q Panel on the K Rack. There are five rows starting with five red switches on the left, these are the addition rows which add the specified impulses together (Note: Don't forget, whenever we are talking about Colossus adding numbers, it's doing addition modulo 2 or XOR addition). Before we start, we should clear the calculation we were doing earlier to count letter B characters. You only need to clear the grey counter switches to do this so make sure the **counter 1** grey switch (and any others you may have set) on the top section of the Q Panel are set to the middle position.

Find the red bank of switches in the lower part of the K Rack key panel and set both the **first** and **second** red keys on the top row to the down position. This tells Colossus to add the first and second impulse.

Next, the yellow switch on each line sets whether to count when the result is either a cross or a dot so set the yellow **equals** key on the same line to up which sets this line to check for the result equals dot.

Finally, the set of five red keys on the right are to select which counter to use. Set the first red key (**counter 1**) to the down position.

This is now counting all tape characters, stepping the Chi wheels at the same time, where ΔZ1+ΔZ2+ΔX1+ΔX2 = dot. This count though is currently only showing the result where the start position of Chi wheel 1 & 2 are both set to 1, we need to tell Colossus to step through each start position in turn and to run the calculation on all start positions.

Assigning a Set Total

If we ran this full set of calculations now, we would get 1,271 separate listed results to sort through to find the highest dot count, but Colossus has a way to filter these results down to a more manageable level. This setting for this is called the Set Total and we can set a fixed value so that Colossus will only print results which are above it. What value should we set? The calculation to work out what the set total is for the 1+2=. is as follows. The expected random score is half n, i.e. half of our total number of characters (5667/2 = 2834). We are using two elements of the five in the ITA2 characters, X1 and X2, so the standard deviation(sigma) of this is ½(sqroot(n)) = 37.6 which, for a 1+2=. run was usually set to 2.5 times sigma up on the random score. This gives a set total value of 2834+94 = 2928.

The Set Total dials are found on the C Rack (the counter rack), which is the rack on the far left of Colossus. There are five columns of four dials, one column for each of the five counters, with a dial to set each decade of the value. Therefore, to set our value of 2928 for counter 1, we need to set the four dials in the first column from the left. To set a dial, click on it and then drag left or right to change the setting.

Set the **1000s dial** to 2 (for 2000), the **100s dial** below it to 9, the **10s dial** to 2 and last of all, the **unit dial** to 8.

Now, we need to specify if Colossus should print if the value is greater or less than the set total, this can be set on the black switches to the right of the dials.

Set the top **counter 1 set total switch** to the > sign on the right.

Starting the run

For this run, Colossus needs to check and count the result for every possible combination of the Chi 1 & Chi 2 wheels. We want to start with the wheels in start positions X1=1, X2=1 and do a count using the whole tape, then X1=2, X2=1 and again do a count etc. Once we get to X1=41, X2=1 (41 is the total number of setting switches on the Chi 1 wheel), we then want to check X1=1, X1=2 and so on right up to the final check X1=41, X2=31.

To set this, we need to look at the main control panel on the relay rack and specifically, the double row of switches to the right of the display panel. We want to set the Chi1 wheel to step fast and the Chi2 wheel step slow (ie, only step once the fast stepping has completed a full rotation). To do this, you will need to set the first two of the blue switches on the bottom row. Set the **Chi 1 stepping switch** down to set as a fast step and set **Chi 2 stepping switch** up to set as a slow step.

We are finally almost ready to begin! The final steps are to press the **SU - SetUp Wheels** switch down (it's the green switch marked SU on the top row - note that it is spring loaded and will return to the centre position once set). This will assign the settings which have been plugged on the jack plug strip below (currently should be all set to start at 1) onto the uniselectors which keep track of the current position of each wheel. The current wheel positions are shown on the lower part of the display panel.

We're all set - so to start the run, press the **MAS - Master switch** down (a black switch, second in on the top right marked "MAS").

Reading the results

All being well, Colossus will now start to step through each combination of both Chi wheel 1 and 2 in turn, adding each character for the whole tape each time our specified logic and counting how many dots are found.

If you look at the Display Panel on the Relay Rack, the lower section shows the current start setting that Colossus is working on. You should see the X1 and X2 counters begin to step upwards, the number on the left are the tens and the 0-9 on the right are the units. After each full tape rotation, the current counter display will be shown and if the value is above the Set Total value, the Electromatic typewriter will output a line showing the X1 and X2 settings as well as the count. The first line to come up on the typewriter should be 01 05 a2937. This line is found where the X2 count has got to 5 (the X1 has done a full check of all 41 settings five times) and the X1 is on 1. The a before the actual count specifies which of the counters are being used (a being counter 1 up to e for counter 5). This will be useful later once we start using the Colossus multiple stepping feature.

Even though Colossus is running the tape at 5000 characters per second, it's a lot of calculations and will take a while to complete. It is doing 5000 characters x 41 Chi1 wheel settings x 31 Chi2 wheel settings so there is a lot of calculating to do so get your favourite hot drink and settle in to watch. If you'd rather not wait, Virtual Colossus has a trick to help pass the time a little quicker!

To the left on the wall is a clock, if you click on the clock face once, you'll get a time speed up (depending on your modern computer speed) which will help pass the time a little faster. Click once more to switch back to normal time. I bet the WRNs operating the original Colossus wished they had this option!

Once Colossus has done the full run using all settings, it will start working from the beginning again, it won't stop automatically. To show that it has done a full run, there is a yellow bulb at the top of the Rectangling Rack (above the typewriter) that will light up once the initial start positions are run for a second time. Watch for that bulb lighting up and you can return the MAS switch back to it's normal position to stop Colossus.

Take a closer look at the result printout (I don't suggest you do it now as you'll lose your place on this tutorial, but you can click on the paper to get an easier to read version of the full print on this clipboard). The first few results should be as follows

X1 X2 count
01 05 a2937
24 05 a2929
31 05 a3108
38 05 a2937
09 06 a2940

The result we are looking for is the one with the largest count, one which is well above the average value of 2834. By far the largest count is where **X1=31 and X2=5** at a count of **3108**. this is about 7 sigma above what would be expected of a random string of letters - an almost certain result!

This therefore is likely to be the Chi1 & Chi2 start settings the German operator set the Lorenz SZ42 machine wheels to when enciphering this message! We have our initial break-in.

If this initial break-in didn't give a valid or obvious count, then further calculations would have been done on either other wheels together or a different calculation completely. The operators and codebreakers got to know which set of runs was more likely to work for each Lorenz link as time went on and more messages deciphered.

Multiple stepping using the Remembering circuits

The methods used in this tutorial, for simplicity, are using single stepping, checking one full set of calculations for each rotation of the tape, but from Colossus v2 onwards, "remembering" circuits were added which meant Colossus could store the four previous tape impulses and run calculations all five in parallel. This gave a five times increase in speed giving an effective 25,000 character per second processing speed. This next section will show you how to run the same calculation we just completed, but this time, with the remembering circuits.

Using the Colossus v2 Remembering Circuits

When Tommy Flowers built the second Colossus computer, he made quite a few improvements over the first one and this version of Virtual Colossus is based on that second version.

One of the main improvements was to add in "Remembering" circuits which could hold the previous four bits from any of the Chi, Mu or Psi streams. This allowed five comparisons to be made for each cipher character read from the tape which meant that we could step forward in jumps of five wheel positions at a time rather than one. This makes quite a speed difference as we'll now test.

The remembered bits appear as a set of green switches under "R" on the main K rack Q switch panel.

On the main switching Q Panel (K rack), we should currently have the first two red switches still both down. Set the **first red switch** to its midpoint and then switch down the **green remembered bit** switch on the same row. Make sure we have the yellow switch up (to check for dot) and then clear the **red counter switch 1** and set down **counter switch 5**.

Next, continue down the next four rows settings each in a similar pattern. Set the second red switch down, the green switch down, the yellow switch up and finally, counter switch 4 for the second row down, counter 3 for the third etc. until we have all five counters being used.

The reason we are putting the first line into counter 5 and going backwards is so the results would be printed in the correct order. The first green switch gives us the actual X1 bit, the second switch gives us the R bit one character back, the third, two characters back etc.

Next, we need to move over to the Main Control Panel on the Relay Rack where we need to tell Colossus which wheel to remember. This is done using the red switches – find the one marked **χ1** and set it up (6th in on top row).

We also need to set each of the Set Total rotary dials to the same value (2928) for each of the five counters (ie, set 2 on all columns of the 1st row and 9 on all the 2nd row etc) plus, switch all five of the set total switches to the right to the > sign.

Before we start the run, take a quick look at the Display Panel, you should see that all five of the counters are working and showing varying counts due to each checking one back on the incoming data.

Clear any previous run and get a fresh roll of paper by clicking on either end of the typewriter platen on the **red paper feed knobs** to remove the existing paper.

Now on the main control panel, press down once on the green SU switch then down on the black MAS switch to begin the run.

Watch the Display Panel again, you should notice that the fast stepping Chi 1(X1) wheel is now jumping in steps of five at a time calculating five steps in parellel per tape rotation. We're now running at an effective processing speed of 25,000 characters per second rather than the actual 5,000 allowed by the tape speed. This allows us to complete what have taken 24 minutes on Colossus v1 in just under 5 minutes on Colossus v2.

Reading the results

Your results should be as follows:

X1	***X2***	***count***
02	***05***	***d2937***
27	***05***	***b2929***
32	***05***	***d3108***
11	***06***	***c2940***
04	***08***	***d2932***
19	***08***	***d2932***
32	***10***	***d2931***
26	***11***	***a2938***
31	***11***	***c2941***
20	***22***	***d2931***
12	***25***	***d2944***
17	***25***	***c2933***
31	***26***	***e2951***
13	***29***	***e3006***
28	***29***	***d2930***
31	***31***	***e2940***

The values you get will need to be adjusted depending on which letter counter is being printed as each is actually running one or more back on the actual value. For the first reading found 02 05 d2937 or X1=02 X2=05 on counter d, to find the actual wheel setting, you would count back one on the X1 position. For counter d, we are actually 1 remembered character back from the actual setting. a is 4 back, b is 3 back, c would be 2 back etc. Therefore, the reading found for counter d in this case is 01 05 = 2937.

You may recall from our initial run that our result was X1=13 and X2=5 with a result of 3108. We can see this on our third row of results in counter d marked as 32 05 d3108 so taking X1 one back as it is in counter d, we get our real result of X1=31, X2=5.

Calculate Chi wheel 4 and 5 starting positions

Now we have the start positions for X1 and X2, we can use those to assist in the calculation for the rotors starts for X4 and X5.

The calculation we will use in this case is called 4=5=/1=2 where the items to the right of the / are known. This means count results where impulse 4 = impulse 5 where the known values for impulse 1 = impulse 2.

The algorithm assumes that we know X1 and X2 so we need to set the start positions for those rotors to the values we found in the first run.

On the Relay Rack, below the main control panel are twelve rows of jack sockets, each corresponding to the number of possible start positions on the Chi, Mu and Psi wheels.

A single jack plug is inserted into the socket where we want Colossus to start, the default should currently be position 1 on all wheels.

Find the top row of plugs and remove the jack plug from socket 1. This is done by clicking on the plug and while holding down the mouse button, drag downwards to remove. Releasing the mouse button with the plug out will remove it fully.

In the last run, we found that the likely start position of Chi wheel 1 (X1) was X1=31 so count along the top row until you find socket number 31. There are white marks on the board every five sockets to help you count, the first board has twenty sockets, so we need to look at the right hand section and eleven sockets in. Once you have found it, simply click on the socket to insert a plug.

Now do the same for the second row which is the X2 start position. We found the start position of X2=5 so remove the X2 plug in position 1 and insert one in **X2 socket 5**.

We can confirm we have the right settings by pressing the **SU - SetUp Wheels** switch on the control panel above. The new start positions of X1=31 and X2=5 should be shown on the lower part of the Display Panel.

We are still using the Q Bus values of ΔZ and ΔX so leave the switches on the J Rack as before but we need to set some new logic on the Q Panel switches.

Using a little logic

Before we start, we should first clear our last program so make sure to set each of the red and yellow switches on the Q Panel we used on the first run back to their normal centred position. For this run, we need to use the conditional switches on the top half of the Q Panel.

Colossus has several conditional rows which can each work simultaneously on the five impulses of data coming in on the Q bus. By default, each of the rows are logically AND together. There are five black switches for each row and each switch can be changed to match either a dot, a cross or both. Remember, we can use these to match specific characters, for example, the letter B in ITA2 code is x..xx so if we set the switches Q1 to Q5 to x..xx respectively, then this row will match any input on the Qbus where the result is a letter B.

The next switch you need to be acquainted with is the yellow switch on each row, this is the negate (or NOT) switch. Setting this down on any row will count where the result is not the conditional value set (ie, count all the characters input where it's not the letter B).

Back to our initial algorithm, the 4=5=/1=2. For this, we want to find all character where the impulse in Q1,Q2,Q4 and Q5 are all equal. To do this, we firstly use the top row to find where each of these four impulses are a dot.

Set Q1, Q2, Q4 and Q5 on the top row to Up (dot) and set Q3 to centre which means anything will match. For the moment, leave the yellow Negate switch at the centre and finally, make sure to set the first grey counter switch down (count the result into counter 1).

Now, we're going to set the same thing on row 2 but checking where all four impulses are a cross.

Set Q1, Q2, Q4 and Q5 on row 2 to Down (cross) then set Q3 to centre, again leave the yellow Negate switch centred for the moment and finally, make sure to set row 2 counter 1 switch to down to also count into counter 1.

The logic for this is now currently count into counter 1 where the QBus matches ..?.. AND xx?xx which is obviously wrong and can never be set! We require ..?.. OR xx?xx for which we can do a clever bit of logic.

An OR logic can be achieved by the following
A OR B = NOT(NOTA AND NOTB)
which we can do on Colossus as follows.

Set the yellow switch **Row 1 Negate** and **Row 2 Negate** both to the down position which gives NOT A and NOT B.

Under the ten rows of grey counter switches on the right are a row of five yellow switches. These set a Negate All to the counter which does a not on all the results above. Set the first of these yellow switches, the **Negate all counter 1** to down.

This means we have now achieved our calculation to find where row 1 OR row 2 impulses are equal and to count them into counter 1.

Now, back to the main control panel on the Relay Rack. We need to change the two rotors that we are stepping so find the two blue switches on the bottom row (X1 & X2) and set them both back to the centre normal position.

Now set the **Chi 4 stepping switch** to down which will make this step 'fast' and **Chi 5 stepping switch** to up which will step 'slow' every time X4 has done a full rotation. As X4 and X5 have 26 and 23 positions respectively, this gives us a total of 598 loops through the cipher text for this run.

Lastly, before we begin this run, we should calculate the Set Total value. The expected random score for four elements out of five is n/8 which gives 709. The sigma is 1/4 sqrt(3n) which equals 32.6. This algorithm usually sets at 5 to 6 sigma so the Set Total should be random + 5 sigma = 872.

On column one of the Set Total switches on the Counter Rack, set the value to 0872.

Before we begin our second run, we should remove the results from the first run on the typewriter. Click on either end of the typewriter platen on the **red paper feed knobs** to remove the existing paper.

To start the run, as before, firstly set the green SU switch on the main control panel to down to reset and confirm we are starting on our existing X1 & X2 values and then press the MAS switch down to begin.

Check the Display Panel is counting up on row X4 and X5 and wait while the full 598 runs are completed until you get the yellow bulb lighting up. (Don't forget you can speed up time a little if you don't want to wait quite as long).

The results

Again, we seem to have a clear highest count for our X4 & X5 wheels at start position **X4=15 and X5=08**! The results should be as follows

```
X4 X5 count
15 03 a0880
15 04 a0900
15 06 a0902
15 08 a0969 <-- highest count
26 08 a0874
15 10 a0907
15 12 a0932
15 14 a0919
15 16 a0889
15 18 a0900
15 22 a0886
```

Finding the last Chi wheel start

We are closing in on the settings for all of the Chi rotors, just one more to find. This is a simple run to get a count of a few set characters for which the count should be higher in the delta plaintext. We will count the number of /(null), 5(figure shift) and U characters for $\Delta Z+\Delta X$ given the start positions of X1,X2,X4 and X5.

For this run, we are again going to use the top conditional rows of the Q Panel to count into three separate counters to find our highest counts for each.

We need to set the following matches
Match / char, ITA = ... in to counter 1
Match 5 char, ITA = xx.xx in to counter 2
Match U char, ITA = xxx.. in to counter 3

So set the top row of black switches to all up, clear the yellow Negate switch and make sure the grey counter 1 switch is down. Row 2: set down, down, up, down down, clear the yellow Negate and this time, make sure just grey counter 2 switch is down. Row 3: set down, down, down, up, up and set into counter 3. Don't forget to also clear the yellow negate all switch below the conditional rows.

For this run, we are just checking one wheel with only 29 positions so let's tell Colossus to output all results.

On the Set Totals panel, change the first < Off > switch to the off position which means ignore the set total value and print everything.

As before, we must now set the start positions of the X4 and X5 wheels on the Relay Rack Jack Plugs start control panel. Remove the plugs on rows 4 and 5 and set them to X4=15 and X5=08.

On the main control panel, set both the blue stepping switches X4 & X5 to the centre and set **Chi 3 stepping switch** to down 'fast' only.

Now finally, get a nice fresh piece of paper in the typewriter, press the SU setup button down and confirm we have our start settings for X1, X2, X4 & X5 then press the MAS switch down to start.

The results

Each of the results for X3 lists the number of characters for /, 5 and U in counter a,b and c respectively. Colossus should give a large maximum count for the / character in counter a of 360 and also for the U character in counter c of 231. We can conclude therefore that the start position of X3 may well be **X3=10**. The results should be as follows

```
X3 count
01 a0273 b0217 c0174
02 a0256 b0218 c0177
03 a0299 b0209 c0189
04 a0225 b0214 c0177
05 a0297 b0219 c0193
06 a0242 b0223 c0166
07 a0301 b0217 c0190
08 a0286 b0208 c0197
09 a0219 b0212 c0170
10 a0360 b0204 c0231 <--
11 a0205 b0211 c0160
12 a0294 b0210 c0203
13 a0270 b0217 c0186
14 a0255 b0217 c0176
15 a0296 b0213 c0204
16 a0234 b0229 c0165
17 a0270 b0224 c0192
18 a0278 b0217 c0167
19 a0267 b0216 c0195
20 a0283 b0207 c0194
21 a0222 b0211 c0180
22 a0327 b0208 c0210
23 a0214 b0220 c0164
24 a0292 b0219 c0206
25 a0280 b0214 c0190
26 a0214 b0220 c0163
27 a0329 b0211 c0211
28 a0223 b0221 c0168
29 a0284 b0205 c0195
```

Congratulations! We have our Chi wheel settings for this message.

X1=31, X2=5, X3=10, X4=15 and X5=8

At this point, where we can calculate the deChi of the message (the result of the ciphertext minus the key added by the Chi wheels), the job was generally passed onto the Testery where code breakers like Jerry Roberts would manually work the final Psi and Motor settings out. This meant the limited number of Colossus machines could get on with the next break in for another message.

Colossus was quite capable of working through and calculating the settings for the whole message so if you're up for the challenge, you can continue on to see if you can get all twelve settings for this message.

Calculating the Motor Settings

This can be achieved by counting the number of / characters in the delta deChi in all places where the Total Motor = x (Total Motor is the basic motor, caused by an x on M37, plus any limitation such as in this case, X2 one back).

Check that the QBus Z is set to ΔZ and QBusX is set to ΔX (as all the start settings for the Chi wheels are now set correctly, this should give us the deChi).

Find the Motor & Limitation Control switches which are found on the J Rack just under the larger ganged Q Bus selection switches. Set the **X2 Limitation switch** down (this means the Total Motor value that the Lorenz used to decide when to step the Psi wheels also uses the value of the Chi wheel number 2, one place back to make the stepping pattern more complex. This was used on the Lorenz SZ42a model.)

Set the last X3 start position jack to our found position of X3=10. Make sure all of the top five rows of start position jack plugs on the Relay Rack are set to our found Chi wheel starts (31, 5, 10, 15 & 8). All of the other start positions should still be set to 1.

Finding the Set Total Motor value

Before we start setting the algorithm, we can use Colossus to find a value for our Set Total for this motor run. First, we can set Colossus to count all the characters into Counter 1.

Set the conditional Row 1 Q1, Q2, Q3, Q4, Q5 and Negate to centre and just switch down this rows Counter 1 switch. Make sure the switches on Row 2 and Row 3 are all centred.

Checking the value on the Display Panel for counter 1, you should see the total characters on the tape **5667**

We would like to count all places where the Total Motor is set to x which requires the use of a special switch. The Total Motor switch is a single yellow switch which can be found right at the bottom of the main Q Panel switching board. Set the **Total Motor** switch to down. This causes the Total Motor signal to only allow counting when Total Motor is an x.

Check the output again and you should see that Counter 1 is now showing a value of 1815, the number of places down the de-Chi cipher text where the Total Motor = x. The random score is 1815/32 approximately 57. The sigma is 1/8(sqrt(7 x 1815)) = 14. The motor wheels on this pattern set at about 10 sigma, so change the counter 1 Set Total to a value of **197** (random + 10 sigma) and switch the Set Total mode switch for counter 1 to the right to >.

Back to our motor setting calculation, to set the motor wheels, first we want to put a / character (all dots) onto the first row of the Conditional Section on the Q Panel and set them to count into Counter 1.

Set each of the top row of grey switches up to dot and set row 1 counters to just count into counter 1.

For the Step settings, we want to set Fast Step to M61 and the Slow Step to M37. This means the 61 pin motor wheel will step after every run but the 37 pin motor wheel will only step once the M61 has completed a full rotation.

To do this on the main control panel, firstly clear the X3 blue switch on the bottom row. The Motor (Mu) step switches are the two blue switches, between the red ones, the 7th & 8th in from the left. Set the first **M61 step switch** down (fast) and the second **M37 step switch** up (slow).

Get a nice fresh roll of paper in the typewriter, press down on the SU switch to set our start positions then start Colossus running with the MAS switch.

Note that this is a big run (67 x 37 = 2,479 tape revolutions at 5000 chars/sec is about 47 minutes in real time!) so you might want to speed it up a little.

You should confirm that the highest score for all possible starts on our two motor wheels is **247 at M61=59, M37=26.**

We now have the settings for M61 and M37 so set these in the start positions on the jack plug strips on the main relay rack. The Motor start jack strips are the middle two separated ones on the board, M61 the upper and M37 the lower one.

Calculating the Psi wheels

Now we are starting to get to work on the final plaintext German in the final message, so setting the Psi wheels requires the direct Z, X, & Ψ signals, not the deltas. Any limitations must be set (in this case the X2 one-back limitation).

On the J Rack, set the Q Bus selection ganged switches Z, X and Ψ all up to their non delta inputs. Confirm that the X2 limitation switch below is set into the down position.

The most common character in the plain text is usually "space" which is ..x.. in ITA2 code, thus checking for maximum count of impulses 1+2=., 3=x and 4+5=. should set the Psis.

Setting Psi 1 & 2

One way to set Psi1 and Psi2 is to do 1+2=. on the Addition section, so we should first clear any counters on the Conditional area.

Set both the first and second red switches on the top row of the lower addition section to down (add impulses Q1 & Q2). Set the yellow equals switch to Up (dot) and then set the Counter 1 red switch on the right down.

We also need to centre the Set Total Motor switch (the yellow switch on it's own at the bottom of this panel).

For our Set Total this time, since we are working with two elements, the same as our very first calculation on the X1 & X2 wheels, we can use the same value of 2928.

For stepping, we need to clear the two blue Motor stepping switches on the main control panel and set the first blue switch in the Ψ group **Ψ1 Stepping switch** down (fast) and the **Ψ2 Stepping switch** up (slow).

You should hopefully know how to start a Colossus run by now so clear our typewriter, SU then MAS to begin and wait for Colossus to calculate our results.

You should see a clear maximum count of 3634 at **Ψ1=32 and Ψ2=36**.

Again, as before, we should set these known values on the start positions jack strips ready for our next count. The Psi wheel starts are the five grouped rows of jack sockets on the lower part of the jack strips. Set Start Ψ1 to 32, Start Ψ2 to 36.

Setting Psi 4 & 5

Now, set S4 and S5 by using 4+5=. which is going to be almost identical to our first run again. On the Q Panel, centre the Q1 and Q2 red switches and set down the Q4 and Q5 on the same row to add the 4th and 5th impulses this time.

Change the Fast Step to Ψ4 and the Slow Step to Ψ5 and use the same Set Total value. Start your run and again find the highest count. The printout should show another clear maximum count of 3705 at Ψ4=9 and Ψ5=43 so set those on the last two jack strips.

The final Ψ5 count

Nearly there! Our final count is to find Ψ3 using 3=x. Set the Psi 4 & Psi 5 red addition switches back to centre and the middle Psi 3 to down.

Change the Equals yellow switch to be in the down position (count where the result is equal to cross).

Clear the Ψ4 & Ψ5 stepping switches and put **Ψ3 Stepping switch** down.

For the final time, clear our typewriter, press SU then MAS and then sit back and wait for your final setting. Maximum count of 3055 shows where **Ψ3=11**.

Our Lorenz cipher message results

Finally, we have our Psi wheel start positions and our full set of start positions for all twelve wheels!

Ψ1 = 32, Ψ2 = 36, Ψ3 = 11, Ψ4 = 09, Ψ5 = 43
M37 = 26, M61 = 59
X1 = 31, X2 = 05, X3 = 10, X4 = 15, X5 = 08

Good work codebreaker - very well done if you made it this far!

How to read the final message

Colossus did not have a setting to directly output the plaintext message, although it was possible to check it one character at a time to confirm a good setting using the span dials. More generally, the operator would run a quick count matching several different characters and confirming that the most common ones were the highest count using all of the correct start positions (for example space, the figure and letter control characters).

The next step in deciphering the message was that the start positions would have been sent on to the Tunny machines which were electronic, rack-based machines which were built to emulate the Lorenz machine. The found wheel positions would be set on the Tunny machine and the ciphertext typed into the teleprinter attached. If all was well, plain German would come out of the printer!

There is not (as yet) currently a VirtualTunny machine simulation available on VirtualColossus, but there is a **Lorenz simulation** so we can use that, but there is a slight problem! If you dialled in the start positions you now have into the Lorenz machine to decipher the message, you would find the output is still a random string of characters .. what is the problem?

The reason is that the Lorenz machines rotors run backwards with respect to the setting numbers on them! If you start Chi wheel 1 on setting 10 for example, when it steps after enciphering a character, it will move to the position marked 9, then 8,7,6 and so on rather than the expected counting up. Bletchley Park didn't know this of course, so when they numbered the start positions, they assumed it was counting up. It doesn't matter for deciphering the message if they were using the same numbering on both Colossus and the Tunny machine to decipher it, but it does mean that the start position and pin numbering between the Lorenz machine as used in Germany and Colossus are reversed!

This is the reversed wheel settings for Chi 5 (with 23 positions).

Lorenz	23	22	21	20	19	18	17	16	15	14	13	12
Colossus	01	02	03	04	05	06	07	08	09	10	11	12

Lorenz	11	10	09	08	07	06	05	04	03	02	01
Colossus	13	14	15	16	17	18	19	20	21	22	23

You can hopefully see that our found setting for X5 on Colossus is 08 which means, the start position on the actual Lorenz machine would be 16.

Therefore, if you want to decipher this message on the Lorenz SZ42, you must use these start positions.

Ψ1 = 12, Ψ2 = 12, Ψ3 = 41, Ψ4 = 45, Ψ5 = 17
M37 = 12, M61 = 3
X1 = 11, X2 = 27, X3 = 20, X4 = 23, X5 = 16

Load Virtual Lorenz ready to decipher our final message

Twitter: **@VirtualColossus**
Facebook: **VirtualColossus**
Web: **https://virtualcolossus.co.uk**