

DOELSTELLINGEN

- Ontdek een boeiende geschiedenis en ga zelf op zoek.
- De oorsprong van de huidige computer architectuur: CPU, ALU, RAM, ROM, getal omzetting tussen binair, hexadecimaal en logische operaties.
- Leer welke soorten programmeertalen er zijn, en dat het maken van programmas niet moeilijk is.
- Ontdek de machinetaal a.d.h.v. de MOS6502 microprocessor. Leer een computer besturen via RAM en ROM adressen.
- Waar liggen de uitdagingen in de toekomst voor de computer industrie. Wat zijn opportuniteiten, wat zijn gevaren.
- Een wegwijzer en basis voor je verdere leven en studie. Je zal ongetwijfeld in aanraking komen met computers, ongeacht je studiekeuze.

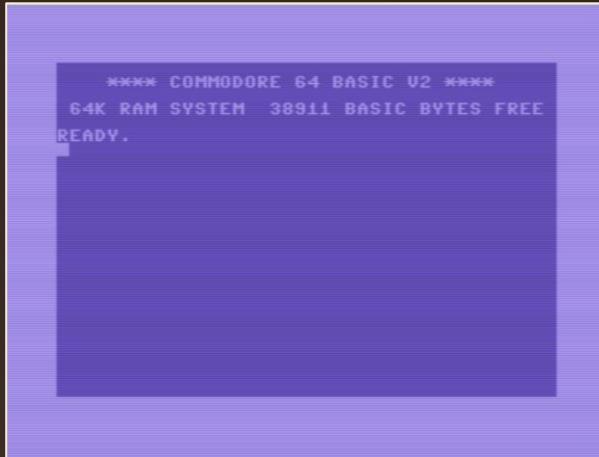
<https://youtu.be/iF8CnG7ee-A>

<https://youtu.be/AvDUDGiL9EQ>



Probeer home
computers van
de 70/80-er
jaren en leer
met emulators!





The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows a workspace structure with files like `pong.c`, `README.md`, and `binair-hexadecimaal.c`.
- Editor:** Displays a C program for the PET 8032 processor. The code includes assembly-like instructions such as `lda #$64*$10000` and `sta.z fpy`. A callout box highlights these instructions.
- Terminal:** Shows the output of running the program in Commodore Basic 4.0, displaying memory usage and a ready prompt.

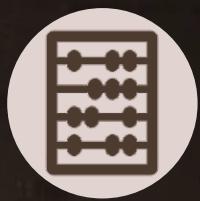
```
1  /*
2   * @file binair-hexadecimaal.c
3   * @author your name (you@domain.com)
4   * @brief Je bent op weg naar je eerste C programma, dat werkt op de PET 8032!
5   *
6   * Je leert wat binaire, octale en hexadecimale talstelsels z
7   * en waarom ze belangrijk zijn in de computerwereld. We zullen
8   * Je leert ook hoe je zelf functies kan maken in je eigen pr
9   * We bekijken samen hoe een C programma in machinetaal is ui
10  * We leren de basis instructieset van de 6502 processor.
11  *
12  * Bekijk OEFENING sectie(s) om dit programma te vervolledigd
13  *
14  * OEFENING 02.1: probeer wat de binaire waarden zijn van de
15  *
16  * OEFENING 02.2: toon de binaire en hexadecimale representat
17  * - Doe dit in eenlus...
18  *     for(char n=0; ...) {
19  *         ...
20  *     }
21  *
22  *
23  * @version 0.1
24  * @date 2022-12-15
25  */

lda #$<$64*$10000
sta.z fpy
lda #>$64*$10000
sta.z fpy+1
lda #<$64*$10000>$10
sta.z fpy+2
lda #>$64*$10000>$10
sta.z fpy+3

*** commodore basic 4.0 ***
31743 bytes free
ready,
```

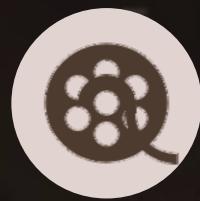
Praktijk via de C-taal en leer wat machinestaal is

Geschiedenis en evolutie van de computer



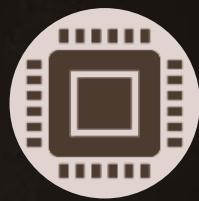
JAREN 40/50

PIONIERS VAN
DE EERSTE
COMPUTERS



JAREN
50/60/70

MAINFRAMES EN
BEDRIJVEN



JAREN 70/80

DE
VERKLEINING
EN DE 8-BIT
HOME COMPUTER



JAREN 80/90

DE DESKTOP PC
EN LAPTOPS



JAREN 90/NU

COMPUTERS IN
ONS DAGELIJKS
GEBRUIK



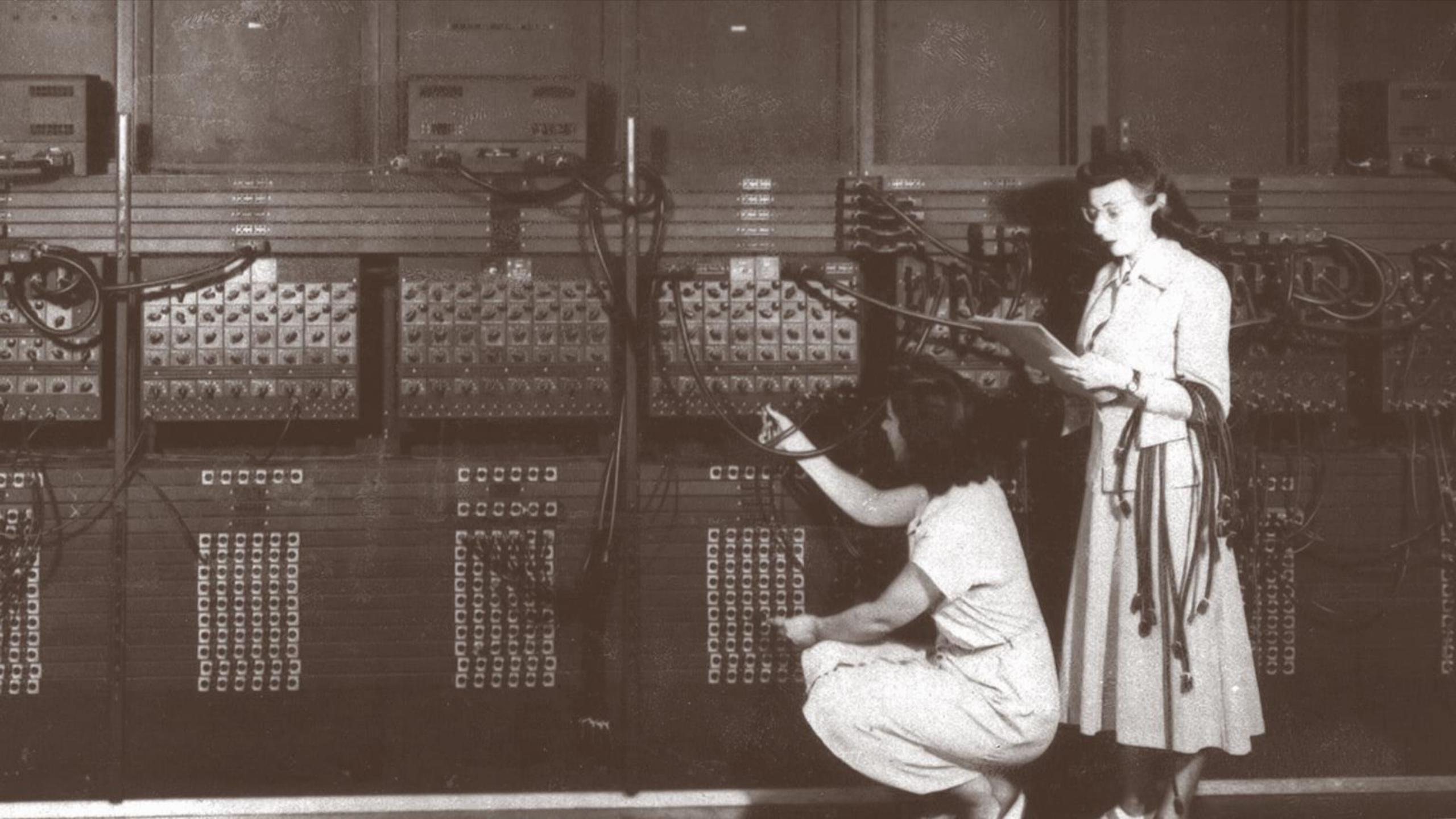
DE TOEKOMST

OPPORTUNITEIT
EN GEVAAR

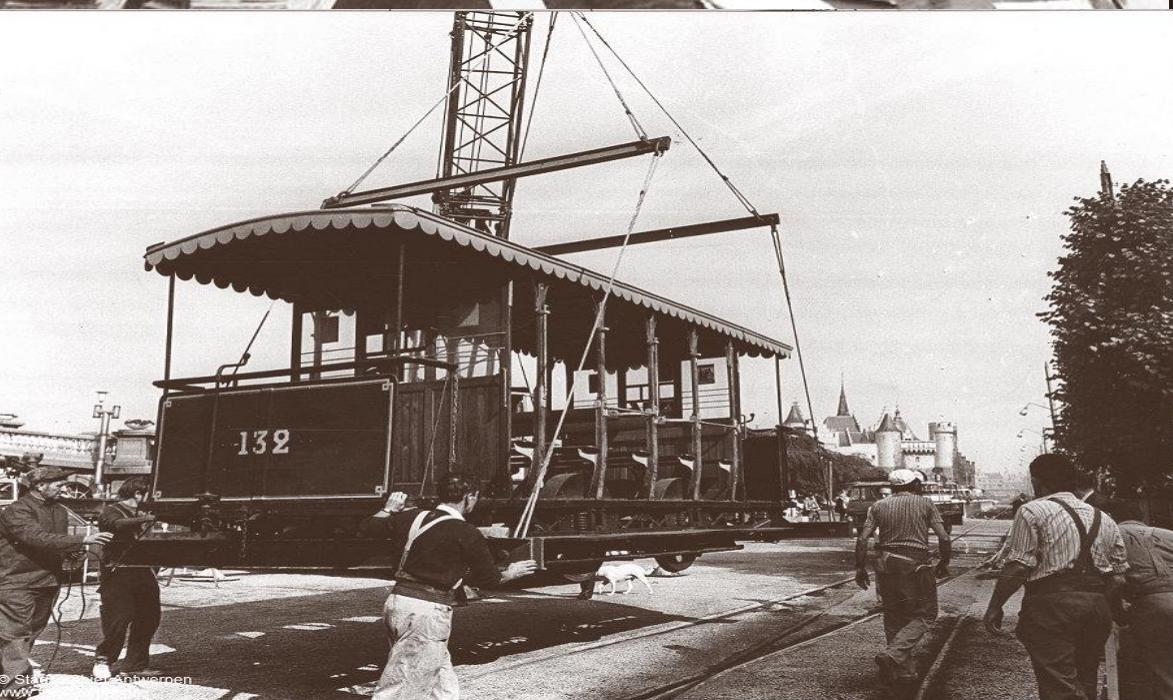


DE PIONIERS

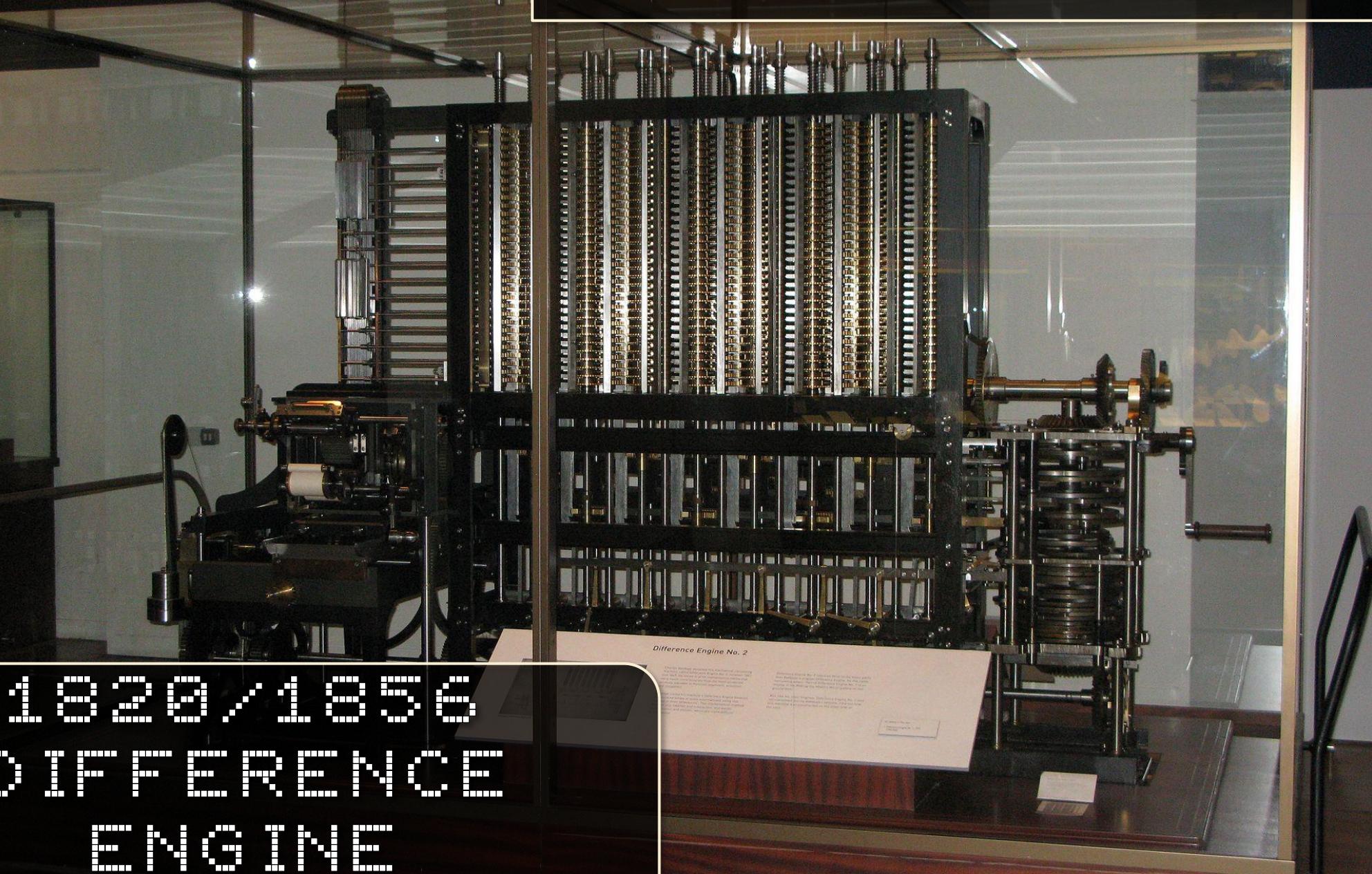
DE JAREN 40 / 50



EEN LEVEN ZONDER COMPUTERS?



CHARLES BABBAGE



1820/1856
DIFFERENCE
ENGINE

Diagram for the computation by the Engine of the Numbers of Bernoulli. See Note G. (page 722 et seq.)

Number of Operation.	Nature of Operation.	Variables acted upon.	Variables receiving results.	Indication of change in the value on any Variable.	Statement of Results.	Data.		Working Variables.										Result Variables.						
						1V_1	1V_2	1V_3	0V_4	0V_5	0V_6	0V_7	0V_8	0V_9	$^0V_{10}$	$^0V_{11}$	$^0V_{12}$	$^0V_{13}$	$^1V_{21}$	$^1V_{22}$	$^1V_{23}$	$^0V_{24}$		
1	\times	$^1V_2 \times ^1V_3$	$^1V_4, ^1V_5, ^1V_6$	$\{^1V_2 = ^1V_2\}$ $\{^1V_3 = ^1V_3\}$	$= 2n$...	2	n	2n	2n	2n								B_1					
2	-	$^1V_4 - ^1V_1$	2V_4	$\{^1V_4 = 2V_4\}$ $\{^1V_1 = ^1V_1\}$	$= 2n - 1$	1	2n - 1										B_2					
3	+	$^1V_5 + ^1V_1$	2V_5	$\{^1V_5 = 2V_5\}$ $\{^1V_1 = ^1V_1\}$	$= 2n + 1$	1	2n + 1														
4	+	$^2V_6 + ^2V_4$	$^1V_{11}$	$\{^2V_6 = 0V_6\}$ $\{^2V_4 = 0V_4\}$	$= \frac{2n - 1}{2n + 1}$	0	0									
5	\div	$^1V_{11} + ^1V_2$	$^2V_{11}$	$\{^1V_{11} = 2V_{11}\}$ $\{^1V_2 = ^2V_2\}$	$= \frac{1}{2} \cdot \frac{2n - 1}{2n + 1}$...	2									
6	-	$^0V_{13} - ^2V_{11}$	$^1V_{13}$	$\{^2V_{11} = 0V_{11}\}$ $\{^0V_{13} = ^1V_{13}\}$	$= -\frac{1}{2} \cdot \frac{2n - 1}{2n + 1} = A_0$									
7	-	$^1V_8 - ^1V_1$	$^1V_{10}$	$\{^1V_8 = ^1V_3\}$ $\{^1V_1 = ^1V_1\}$	$= n - 1 (= 3)$	1	...	n	n - 1									
8	+	$^1V_2 + ^0V_7$	1V_7	$\{^1V_2 = ^1V_2\}$ $\{^0V_7 = ^1V_7\}$	$= 2 + 0 = 2$...	2	2												
9	\div	$^1V_6 + ^1V_7$	$^3V_{11}$	$\{^1V_6 = ^1V_6\}$ $\{^0V_{11} = 3V_{11}\}$	$= \frac{2n}{2} = A_1$	2n	2		$\frac{2n}{2} = A_1$								
10	\times	$^1V_{21} \times ^3V_{11}$	$^1V_{12}$	$\{^1V_{21} = ^1V_{21}\}$ $\{^3V_{11} = 3V_{11}\}$	$= B_1 \cdot \frac{2n}{2} = B_1 A_1$		$\frac{2n}{2} = A_1$	$B_1 \cdot \frac{2n}{2} = B_1 A_1$...	B_1					
11	+	$^1V_{12} + ^1V_{13}$	$^2V_{13}$	$\{^1V_{12} = 0V_{12}\}$ $\{^1V_{13} = 2V_{13}\}$	$= -\frac{1}{2} \cdot \frac{2n - 1}{2n + 1} + B_1 \cdot \frac{2n}{2}$				0		$\left\{ -\frac{1}{2} \cdot \frac{2n - 1}{2n + 1} + B_1 \cdot \frac{2n}{2} \right\}$				
12	-	$^1V_{10} - ^1V_1$	$^2V_{10}$	$\{^1V_{10} = 2V_{10}\}$ $\{^1V_1 = ^1V_1\}$	$= n - 2 (= 2)$	1	n - 2									
13	-	$^1V_6 - ^1V_1$	2V_6	$\{^1V_6 = 2V_6\}$ $\{^1V_1 = ^1V_1\}$	$= 2n - 1$	1	2n - 1														
14	+	$^1V_1 + ^1V_7$	2V_7	$\{^1V_1 = ^1V_1\}$ $\{^1V_7 = 2V_7\}$	$= 2 + 1 = 3$	1	3													
15	\div	$^2V_6 + ^2V_7$	1V_8	$\{^2V_6 = 2V_6\}$ $\{^2V_7 = 2V_7\}$	$= \frac{2n - 1}{3}$	2n - 1	3	$\frac{2n - 1}{3}$												
16	\times	$^1V_8 \times ^3V_{11}$	$^4V_{11}$	$\{^1V_8 = 0V_8\}$ $\{^3V_{11} = 3V_{11}\}$	$= \frac{2n}{2} \cdot \frac{2n - 1}{3}$	0		$\frac{2n}{2} \cdot \frac{2n - 1}{3}$									
17	-	$^2V_6 - ^1V_1$	3V_6	$\{^2V_6 = 3V_6\}$ $\{^1V_1 = ^1V_1\}$	$= 2n - 2$	1	2n - 2														
18	+	$^1V_1 + ^2V_7$	3V_7	$\{^1V_1 = ^1V_1\}$ $\{^2V_7 = 3V_7\}$	$= 3 + 1 = 4$	1	4													
19	\div	$^3V_6 + ^3V_7$	1V_9	$\{^3V_6 = 3V_6\}$ $\{^3V_7 = 3V_7\}$	$= \frac{2n - 2}{4}$	2n - 2	4	$\frac{2n - 2}{4}$...				$\left\{ \frac{2n}{2} \cdot \frac{2n - 1}{3} \cdot \frac{2n - 2}{3} \right\} = A_3$								
20	\times	$^1V_9 \times ^4V_{11}$	$^5V_{11}$	$\{^1V_9 = 0V_9\}$ $\{^4V_{11} = 5V_{11}\}$	$= \frac{2n}{2} \cdot \frac{2n - 1}{3} \cdot \frac{2n - 2}{4} = A_3$	0													
21	\times	$^1V_{22} \times ^5V_{11}$	$^0V_{12}$	$\{^1V_{22} = 1V_{22}\}$ $\{^0V_{12} = 2V_{12}\}$	$= B_3 \cdot \frac{2n}{2} \cdot \frac{2n - 1}{3} \cdot \frac{2n - 2}{3} = B_3 A_3$	0		$B_3 A_3$...		B_3				
22	+	$^2V_{12} + ^2V_{13}$	$^3V_{13}$	$\{^2V_{12} = 3V_{12}\}$ $\{^2V_{13} = 3V_{13}\}$	$= A_0 + B_1 A_1 + B_3 A_3$										
23	-	$^2V_{10} - ^1V_1$	$^3V_{10}$	$\{^2V_{10} = 3V_{10}\}$ $\{^1V_1 = ^1V_1\}$	$= n - 3 (= 1)$	1	n - 3									
Here follows a repetition of Operations thirteen to twenty-three.																				ADA LOVELACE NOTE G				
24	+	$^4V_{13} + ^0V_{24}$	$^1V_{24}$	$\{^4V_{13} = 0V_{13}\}$ $\{^0V_{24} = 1V_{24}\}$	$= B_7$							
25	+	$^1V_1 + ^1V_3$	1V_3	$\{^1V_3 = 1V_3\}$ $\{^0V_6 = 0V_6\}$ $\{^0V_7 = 0V_7\}$	$= n + 1 = 4 + 1 = 5$	1	...	n + 1	0	0												

by a Variable-card.
by a Variable-card.

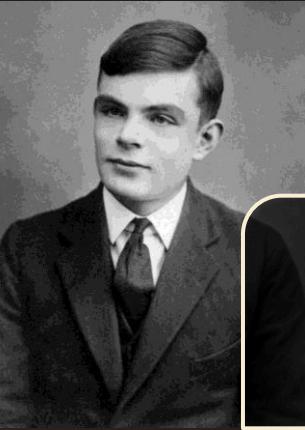
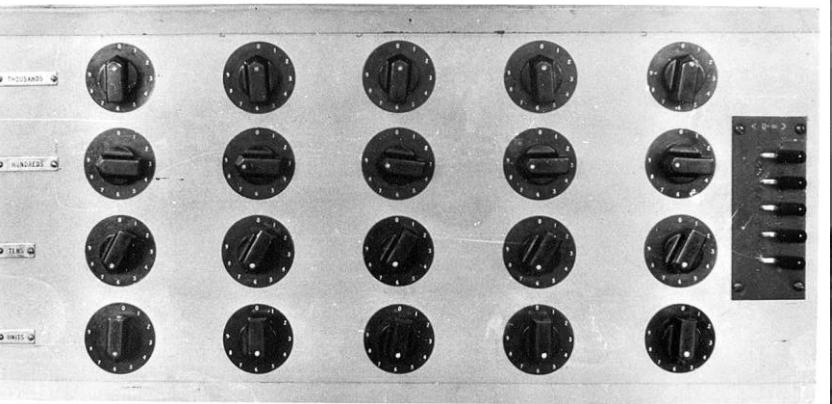
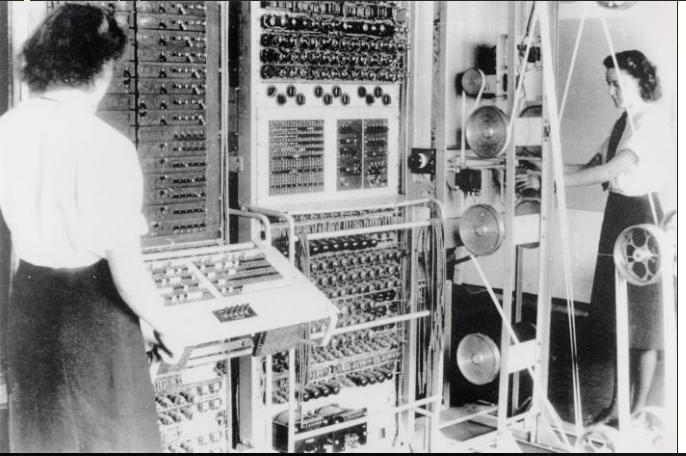
ADA

LOVELACE

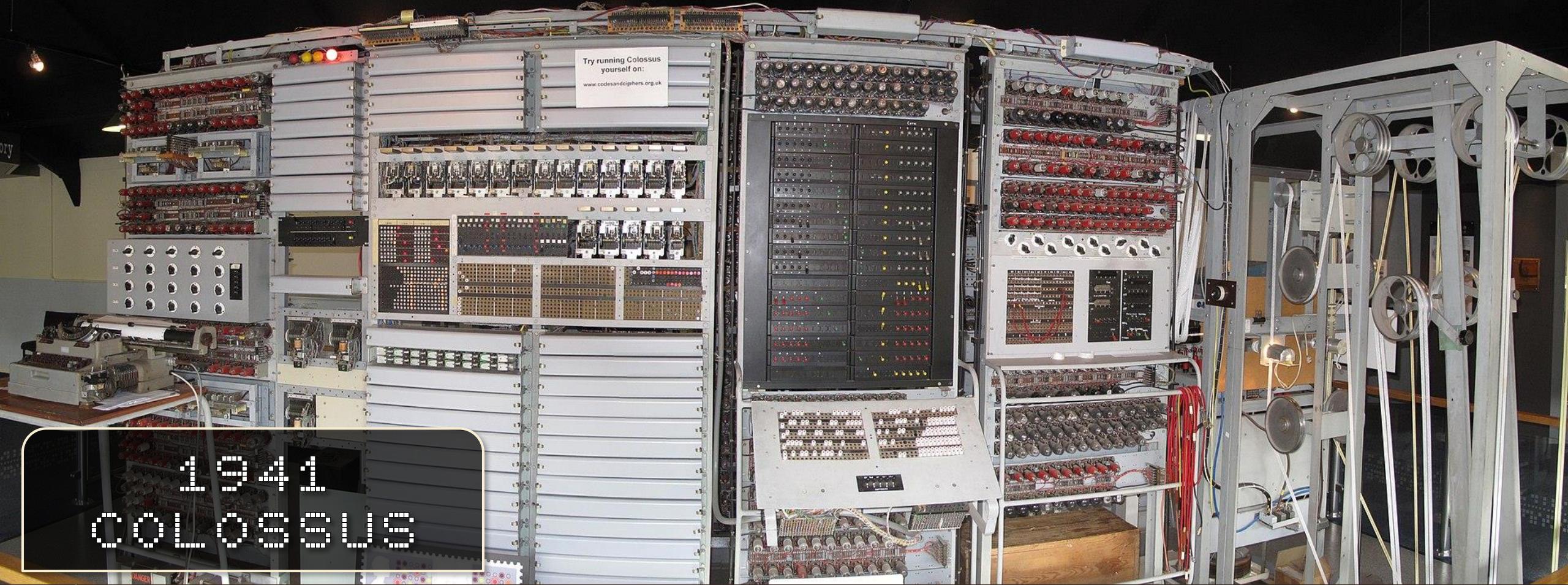
NOTE G

HORST ZUSE

1941
23



ALAN
TURING

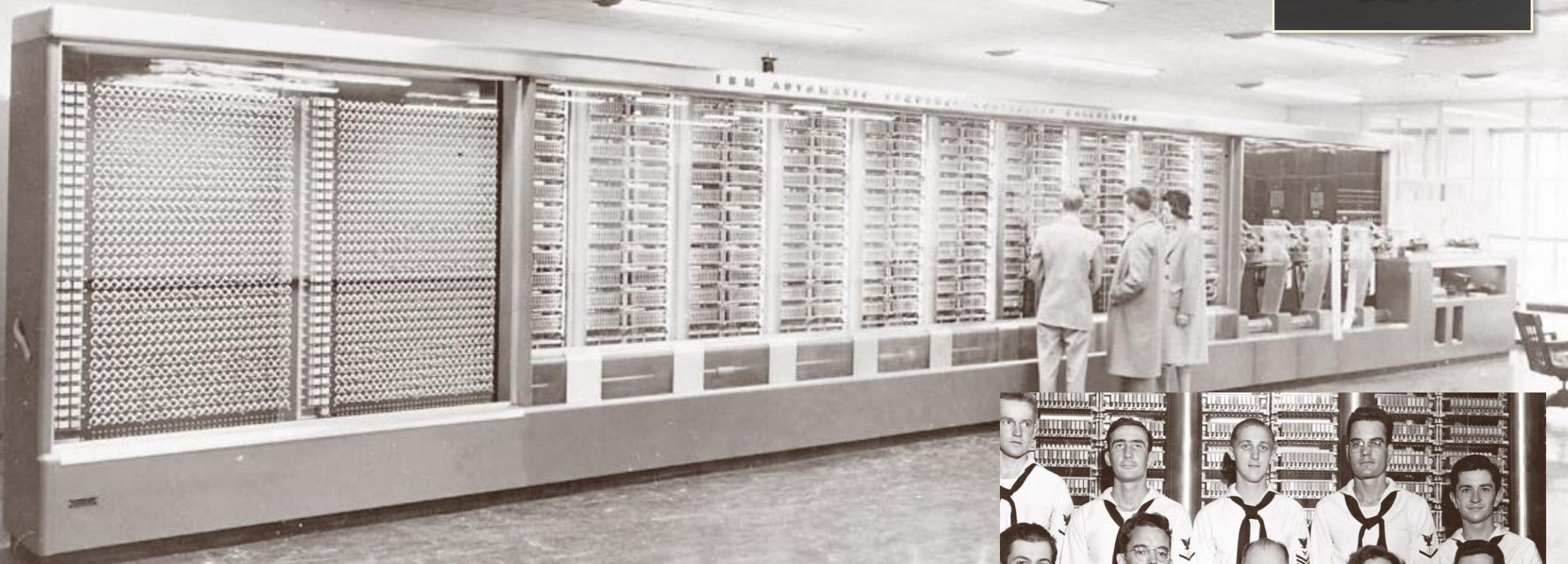


1941
COLOSSUS

1941
COLOSSUS



IBM



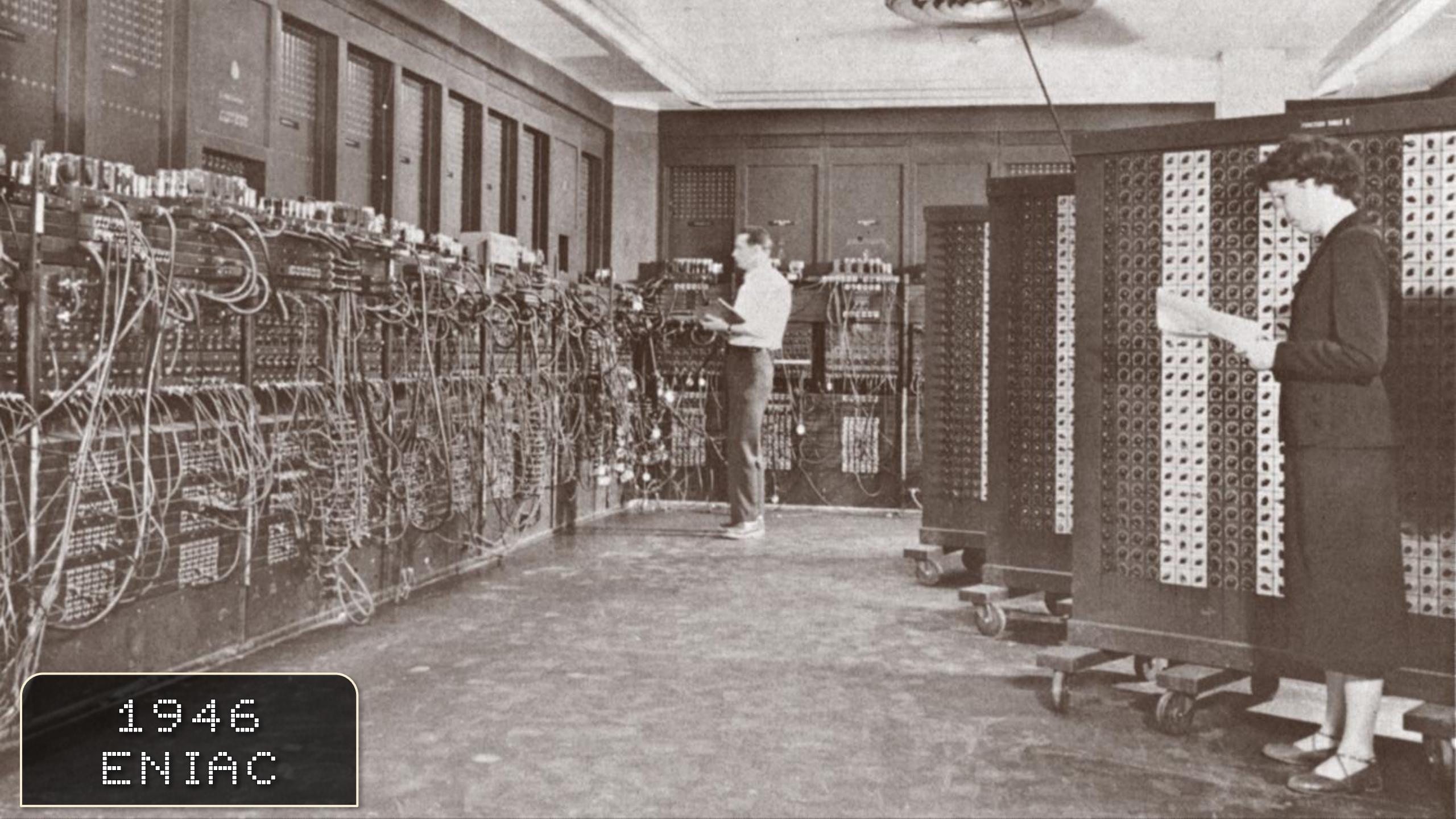
1943
HARVARD MARK I



1943
HARVARD MARK I



1946
ENIAC



1946 - ENIAC

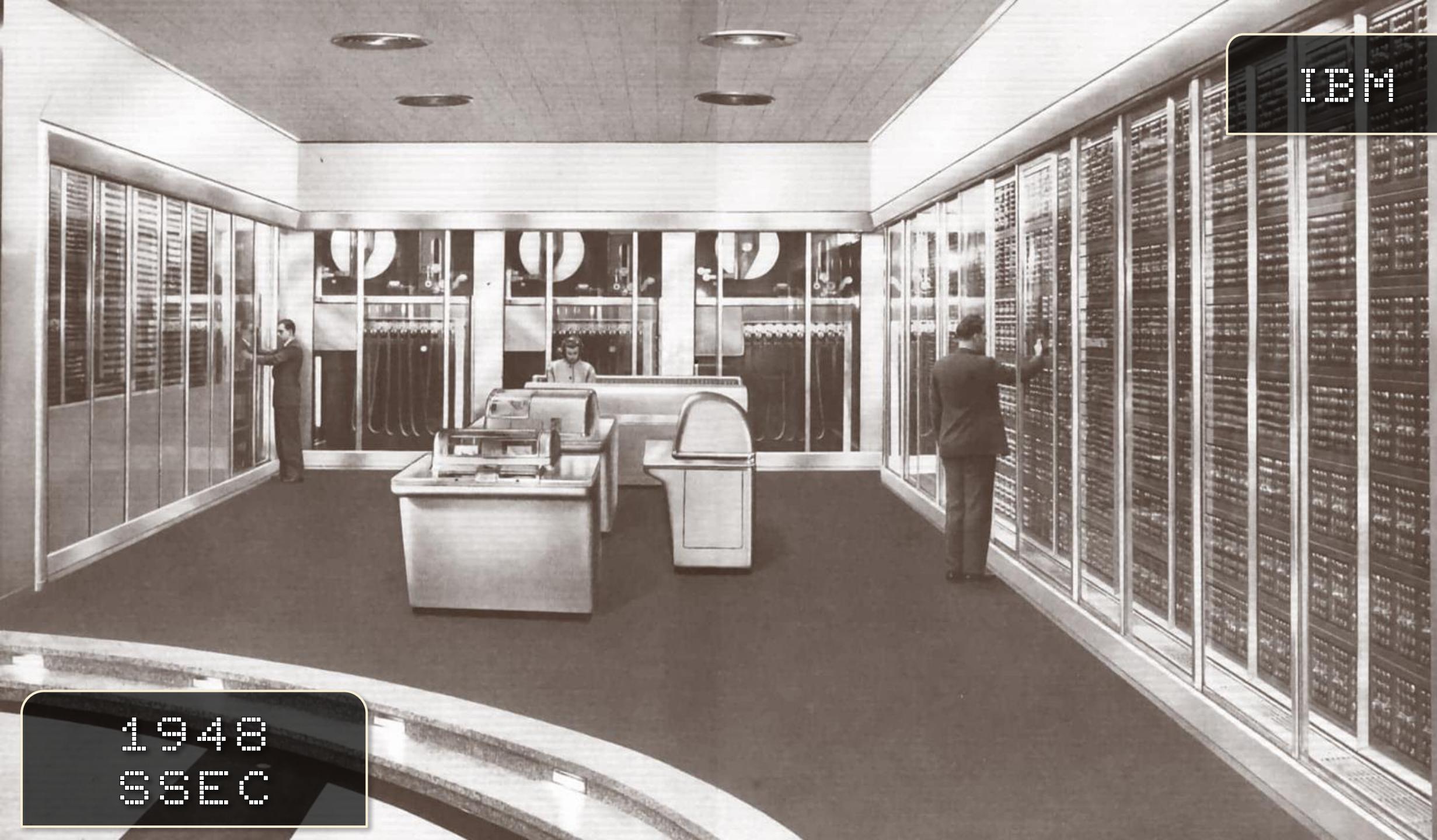
- DE ENIAC WAS DE EERSTE ECHTE PROGRAMMEERBARE ELECTRONISCHE COMPUTER DIE TURING COMPLEET WAS.
- 18,000 VACUUM TUBES.
- 7,200 CRYSTAL DIODES.
- 1,500 RELAYS.
- 70,000 RESISTORS.
- 10,000 CAPACITORS.
- +/- 5,000,000 HAND-GESOLDEerde VERBINDINGEN. GEWICHT WAS MEER DAN 27 TON.
- GROOTTE WAS 170 VERIANTE METER.
- CONSUMEERDE 150 KW/H.

DE ENIAC
WERKTE IN HET
DECIMALE
TALSTELSEL!

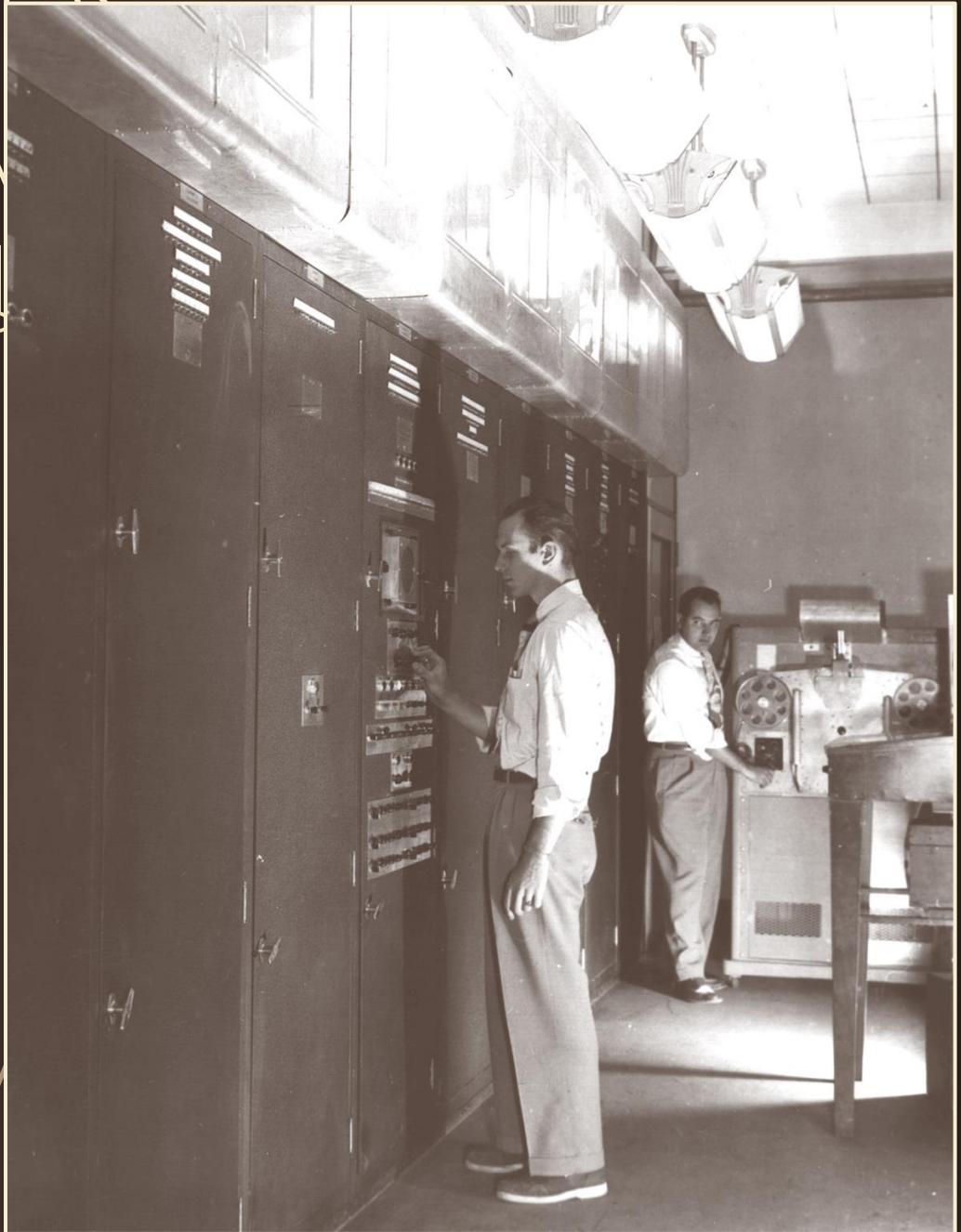
1946 - ENIAC



IBM



1948
SSEC



1949
EDVAC

OPVOLGER VAN
DE ENIAC, EN
WERKT VOLGENS
DE VON NEUMAN
ARCHITECTUUR.



First Draft of a Report on the EDVAC

by

John von Neumann



1947 1949
EDSAC

BRITSE OPVOLGER VAN DE ENIAC. SINDS 2018 KAN JE EEN REPLICA VAN DEZE COMPUTER TERUG VINDEN IN HET BLETCHLEY PARK COMPUTER MUSEUM IN LONDEN.

DEZE COMPUTER WAS OOK VOLLEDIG DIGITAAL EN WERKTE VOLGENS DE VON NEUMANN COMPUTER ARCHITECTUUR.

DE EERSTE (BRITSE) COMMERCIELE COMPUTER VOOR BUSINESS APPLICATIES. LEO KWAM VOORT UIT EEN ENGELS CATERING BEDRIJF DAT INVESTEERDE IN DE COMPUTER INDUSTRIE OM DE COMPUTER CAPACITEIT TE COMMERCIALISEREN. HET WAS GEBASEERD OP DE EDSAC.

1951
LEO



History of the LEO Computer

J. PRESPER ECKERT
JOHN MAUCHLY

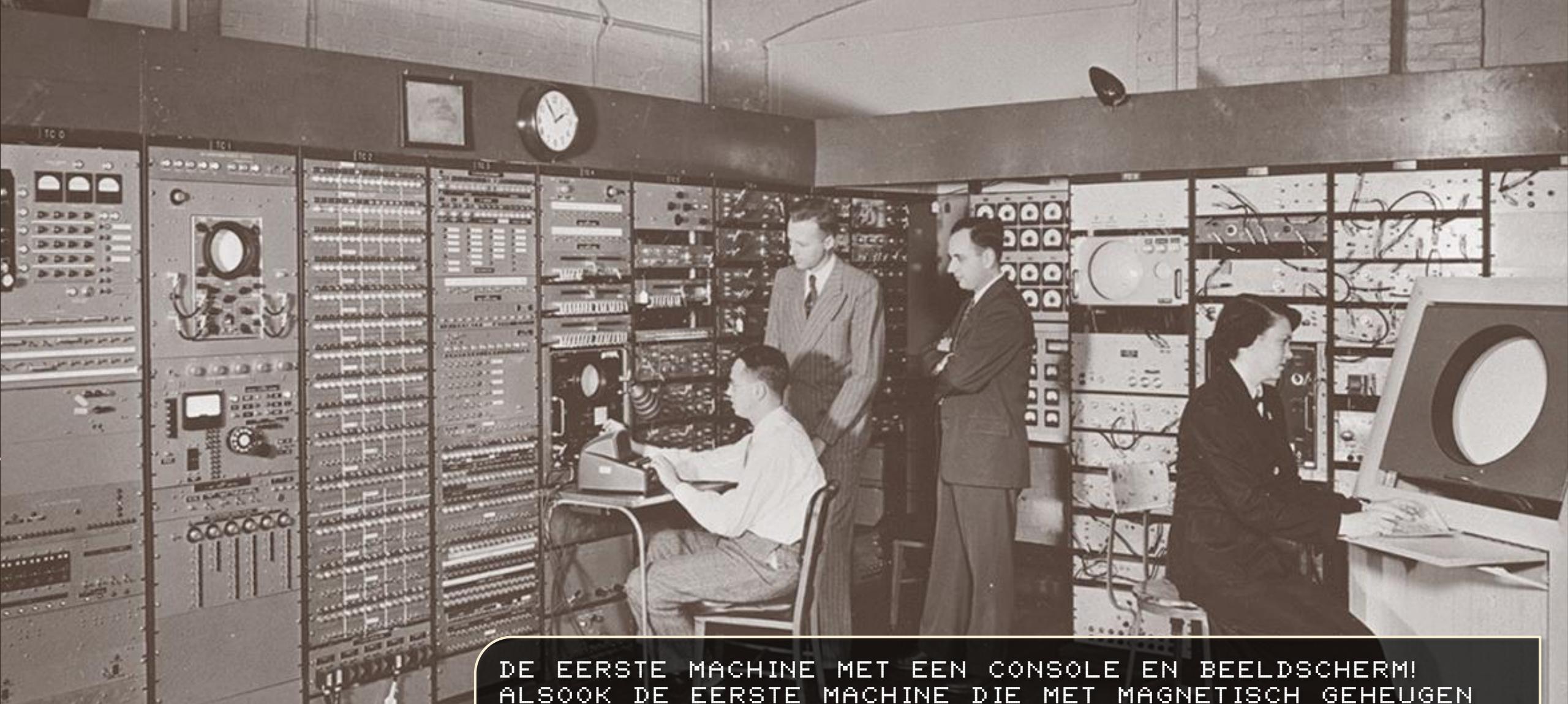
1951
UNIVAC I

DE AMERIKAANSE COMMERCIELE COMPUTER GESTICHT DOOR DE CREATORS VAN DE ENIAC. ER ZIJN ERG VEEL UNIVAC I COMPUTERS ONTWIKKELD EN IN BEDRIJF GENOMEN. MERK OP: DE MAGNEETBAND INSTALLATIES IN DE ACHTERGROND.

JOHN VON NEUMANN

1952
IAS

DE IAS MACHINE WAS DE EERSTE ELECTRONISCHE COMPUTER GEBOUWD IN HET INSTITUUT VOOR GEAVANCEERDE STUDIES IN PRINCETON, NEW JERSEY. DE MACHINE WORDT SOMS OOK DE VON NEUMANN MACHINE GENOEMD, OMDAT HET DOOR JOHN VON NEUMANN WAS ONTWORPEN, PROFESSOR BIJ PRINCETON UNIVERSITEIT EN PRINCETON INSTITUUT.



1951
WHIRLWIND

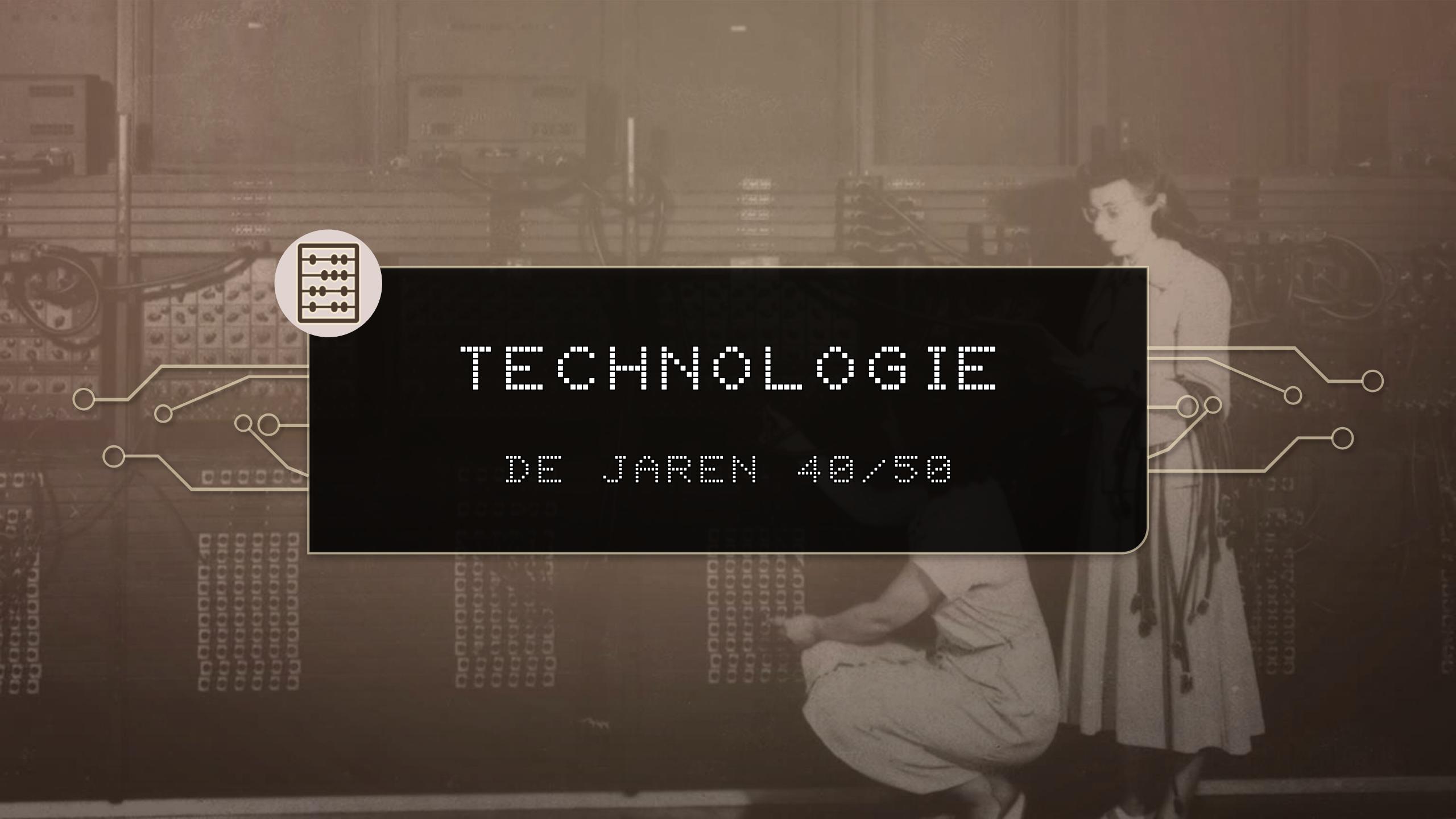
DE EERSTE MACHINE MET EEN CONSOLE EN BEELDSCHERM!
ALSOOK DE EERSTE MACHINE DIE MET MAGNETISCH GEHEUGEN
WERKTE. DIT IN TEGENSTELLING TOT ALLE VORIGE MODELLEN,
DIE MET ELECTRO-STATISCHE OPSLAG OF OPSLAG VIA
VERTRAGING CYCLI WERKten (OP BASIS VAN KWIK). DOOR DEZE
VERBETERINGEN KON DE WHIRLWIND GEHEUGEN PARALLEL
BENAADEREN IPV SEQUENTIEEL, EN DUS VEEL SNELLER WERKEN!

OOK IN BELGIE WAREN WE PIONIERS! HET BELL LABS, TOEN BTMC, STUURDE AL VANAF 1946 INGENIEURS NAAR DE US EN DE UK OM DE COMPUTER TECHNOLOGIE TE LEREN, GEINSPIREERD DOOR DE ENIAC, EDVAC EN DE EDSAC. MAAR HET WAS DE HARVARD MARK IV DIE DE ULTIEME INSPIRATIE BOOD VOOR DE BOUW VAN DEZE COMPUTER IN DE BOERENTOREN IN ANTWERPEN.

1955
IRSIA-FNRS



6322

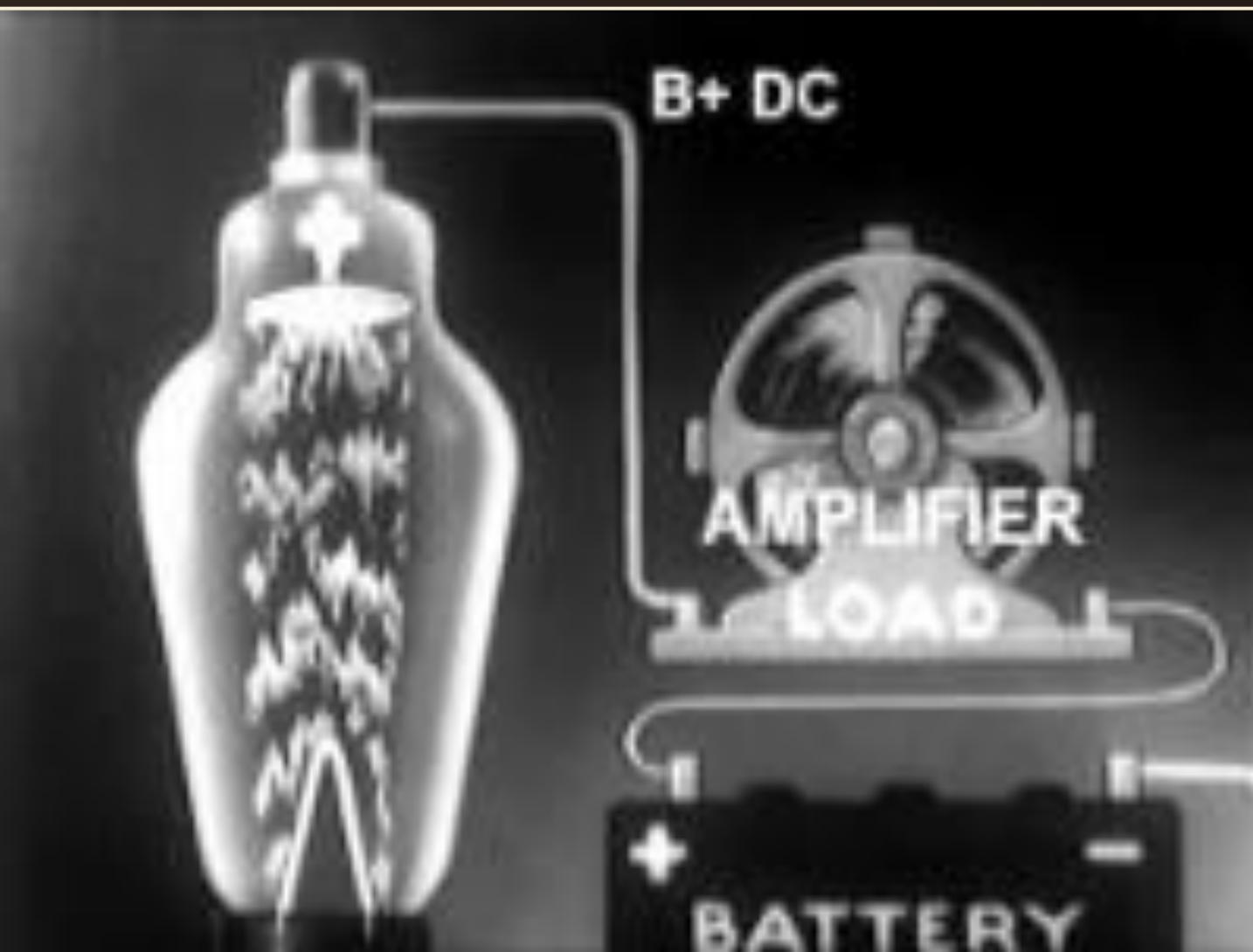


TECHNologie

DE JAREN 40 / 50



VACUUM BUIZEN



VERTRAGINGSLIJNEN



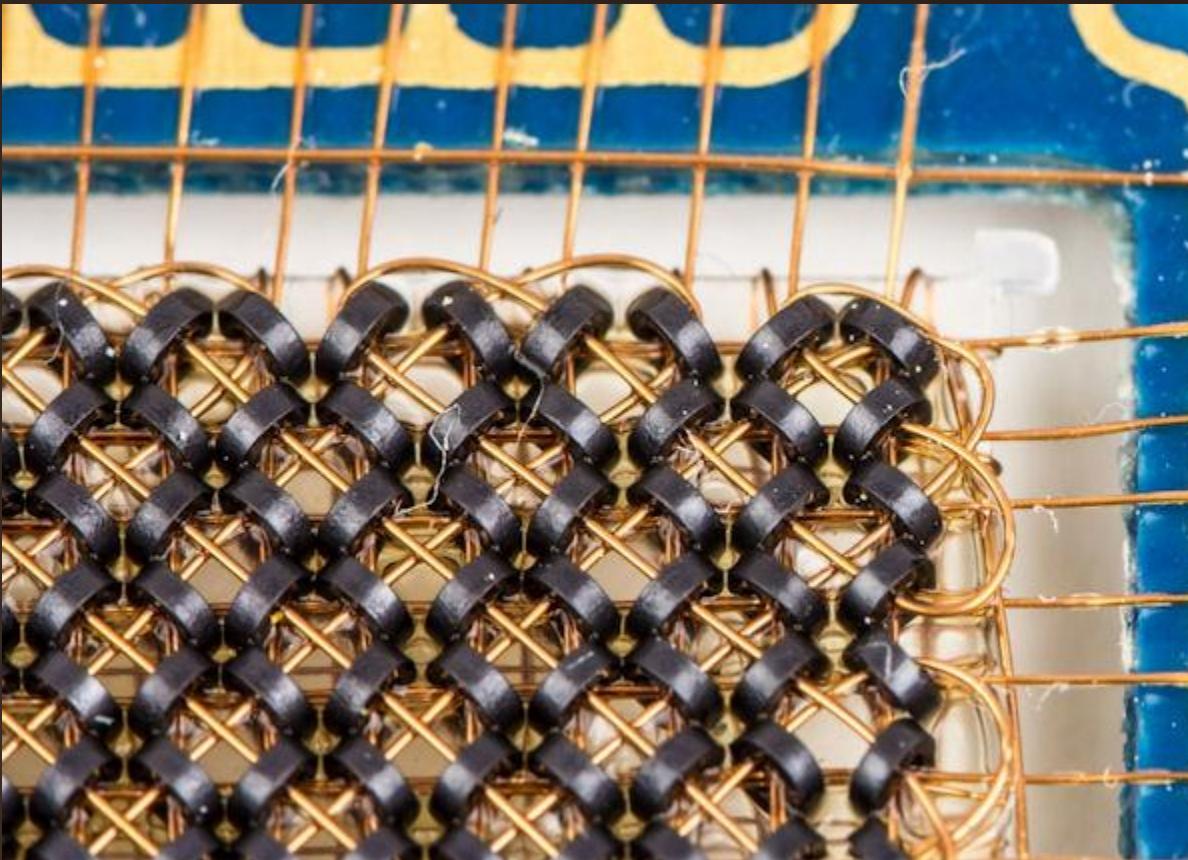
VERTRAGINGSLIJNEN



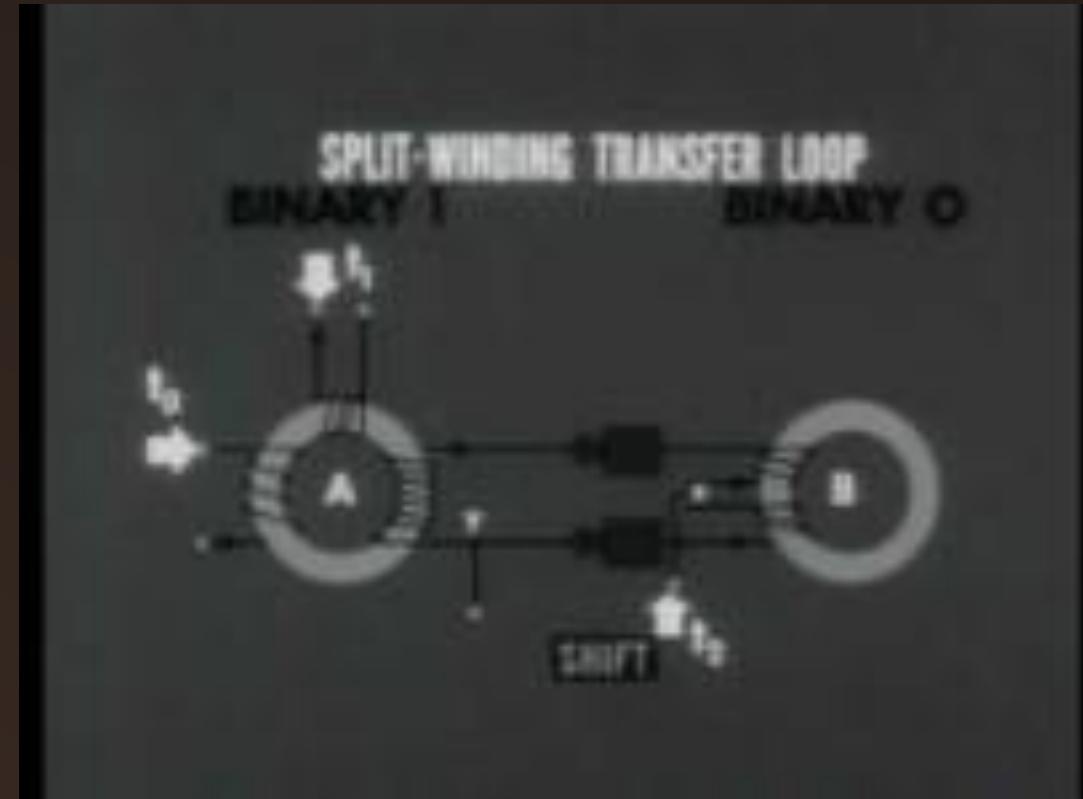
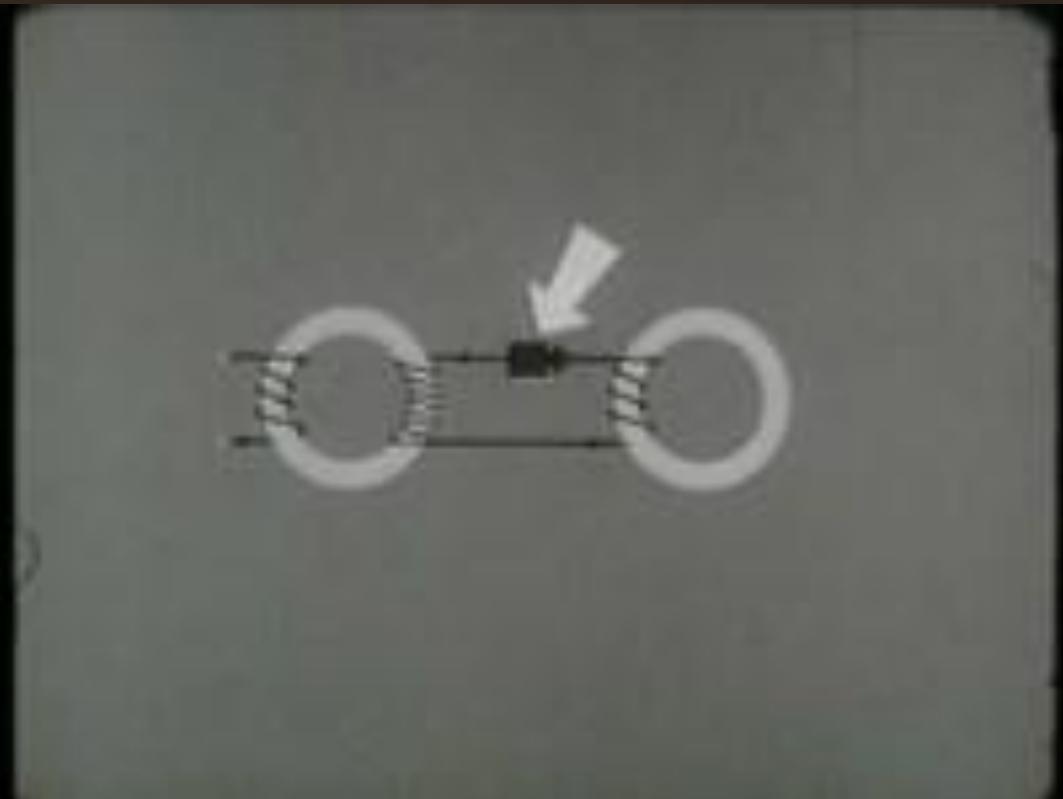
VERTRAGINGSLIJNEN



FERRIET GEHEUGEN



FERRIET GEHEUGEN



TRANSISTORS

The TRANSISTOR

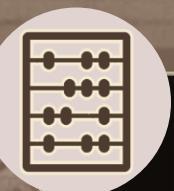
1957



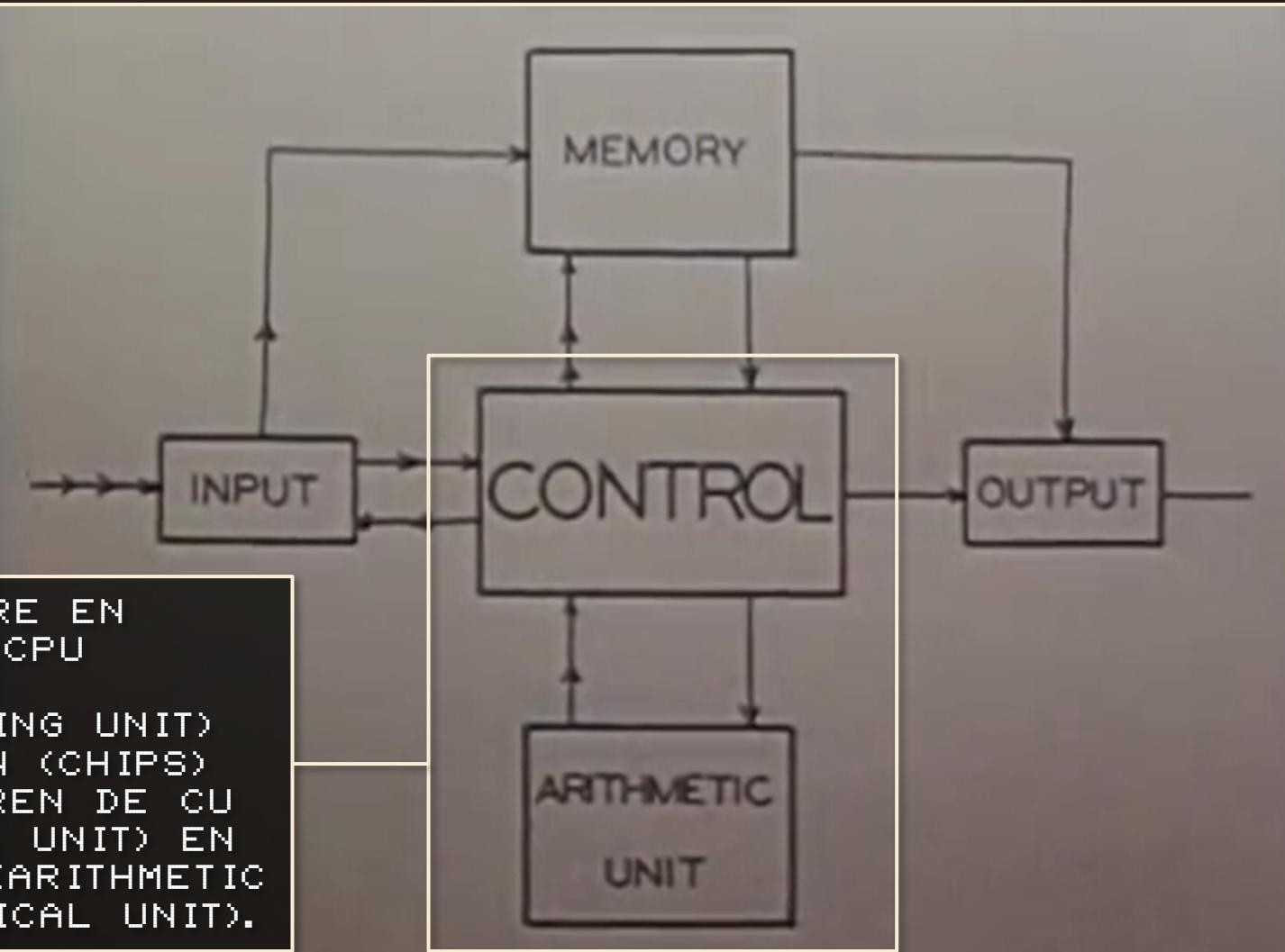


COMPUTER BOUW

DE JAREN 40 / 50



NEUMANN ARCHITECTUUR



DE LATERE EN
HUIDIGE CPU
(CENTRAL
PROCESSING UNIT)
SYSTEMEN (CHIPS)
COMBINEREN DE CU
(CONTROL UNIT) EN
DE ALU (ARITHMETIC
AND LOGICAL UNIT).

MACHINE INSTRUCTIES

TYPICAL ORDERS

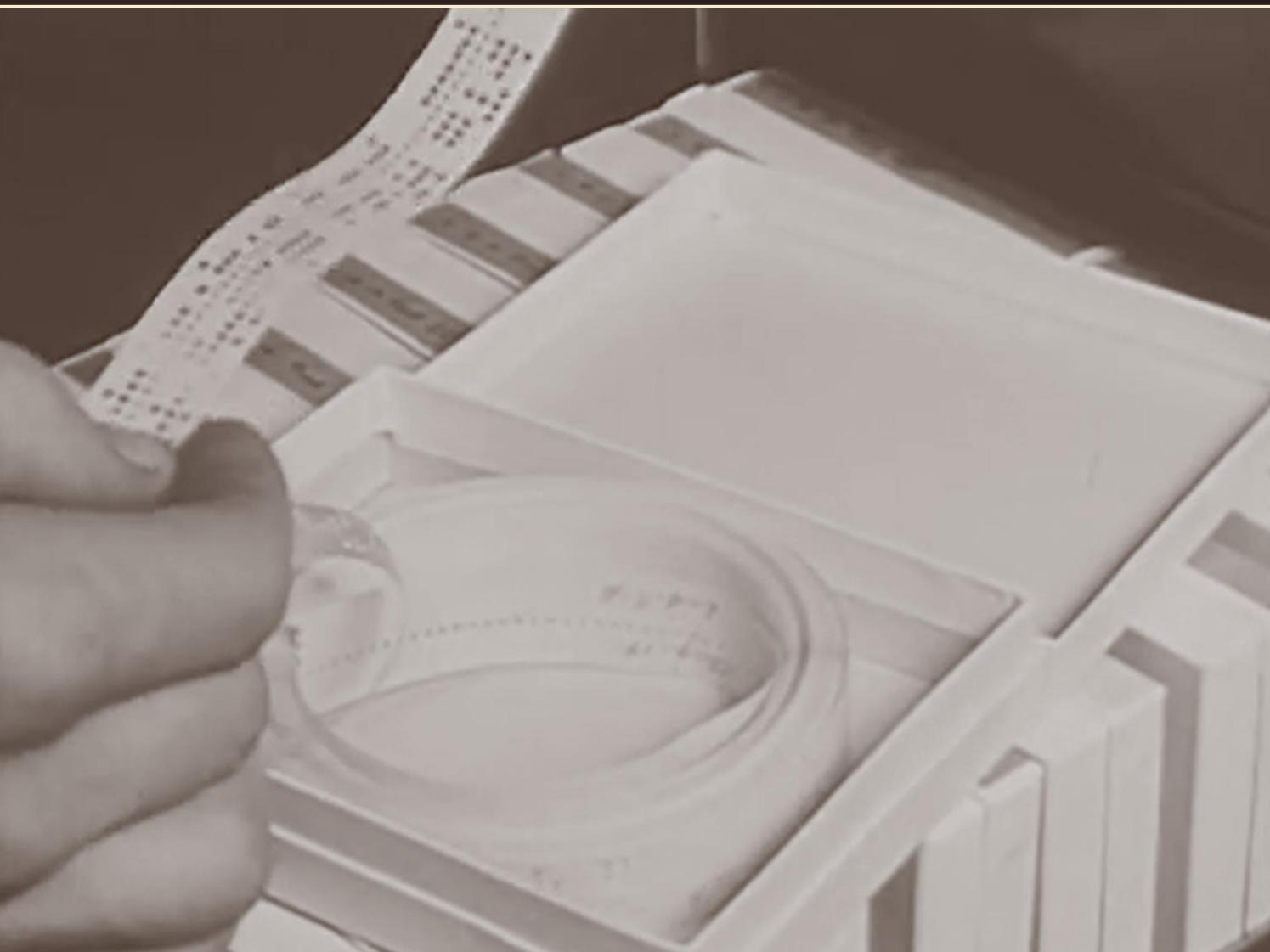
ORDER NO.	CODE SYMBOL	BINARY FORM	TYPE OF OPERATION INDICATED
5	BA	1011 1010	ADDITION
8	BF	1011 1111	SUBTRACTION
10	DA	1101 1010	MULTIPLICATION
12	DD	1101 1101	DIVISION
14	CB	1100 1011	TRANSFER
23	EE	1110 1110	SHIFT
28	EA	1110 1010	PRINT ANSWER

HIER ZIE JE EEN OVERZICHTJE VAN ENKELE ORDERS DIE DE MANIAC COMPUTER KON VERWERKEN. DIT IS GEWELDIG INTERESSANT, WANT JE ZIET HOE DE MANIAC ELKE INSTRUCTIE ALS EEN 8-BIT BINaire INSTRUCTIE IN DE CPU VERWERKTE. OM DE INSTRUCTIES VOOR DE MENS BEGRIJPBAAR TE MAKEN, KRIJGT ELKE MACHINE CODE EEN BESCHRIJVING. PROGRAMMEURS GEBRUIKTEN DEZE ORDER CODES IN EEN FLOW OM EEN PROGRAMMA TE MAKEN.

PROGRAMMEREN OP PAPIER



PROGRAMMA BIBLIOTHEEK



EMULATOR

INSTALLATIE EN PROBEREN

