REPAIRING TWO OF THE FIRST ELECTRONIC CALCULATORS

IME 84 and IME 86S

Jef Ongena (Brussels) and Brent Hilpert (Vancouver)

24 March 2026

Ithinx GmbH

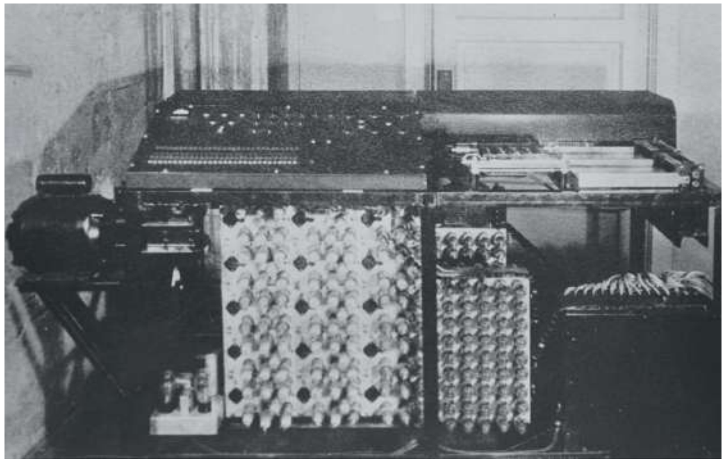
Köln, Germany

Introduction

The 1940-1965 era of computing

Atanasoff-Berry Computer (1937 - 42)

Considered the first electronic computing machine
Purpose built to solve systems of linear equations up to 29 variables



Iowa State College, Ames, USA

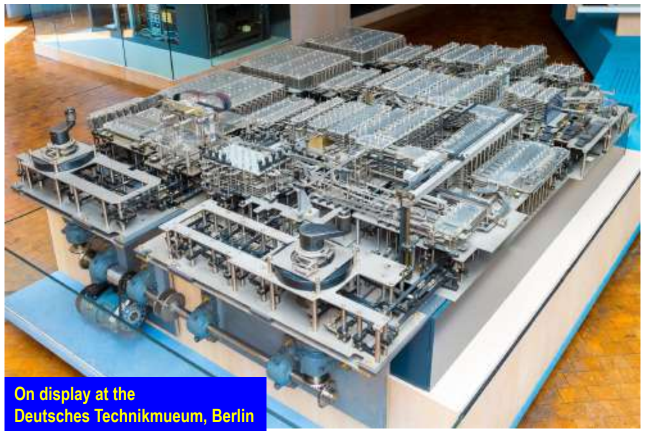
- ~ 300 radiovalves
- ~ 3000 bits memory
- ~ Clock : 60 Hz
- ~ 350 kg

Capacitor based Memory

Zuse Z1

The world's first programmable computer (conceptually) – 1938

Commands entered via paper tape
Mechanical design: fully mechanical, constructed from metal plates and pins
Binary system
Floating-point arithmetic



On display at the Deutsches Technikmueum, Berlin

Berlin, Brunnenstraße 39 (family home)

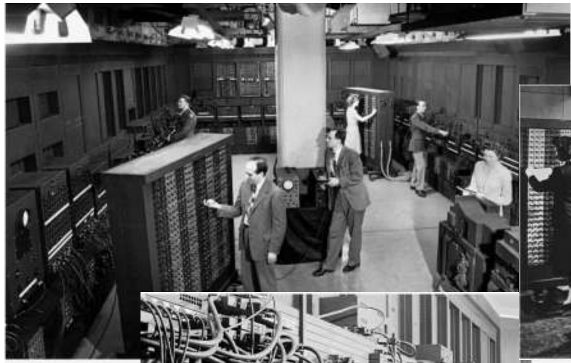
- ~ 16 floating-point numbers with 22 bits per number
- ~ Manually operated ~ 2 Hz (cranck…)
- ~ Weight: 1 ton

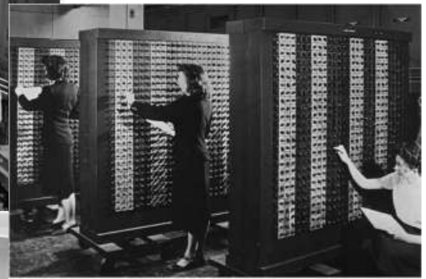
The first large electronic computer was Colossus (design 1943 operation 1944)

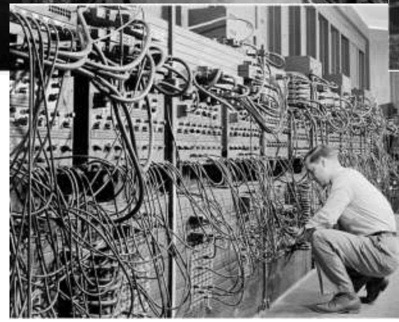
Bletchley Park, Milton Keynes, UK

- ~2500 radiovalves
- ~ 8-10 kW power consumption
- ~ 5 kHz
- ~ 5 tonnes
- Code breaking efforts in WW II

Most famous was the ENIAC (Electronic Numerical Integrator And Computer) (Feb. 1946)







Moore School of Electrical Engineering
University of Pennsylvania
Philiadelphia, USA

- ~18000 radiovalves
- ~ 150kW power consumption
- ~ 100 kHz clock frequency
- Ballistic and atomic calculations

The ENIAC in operation

Whirlwind Computer (design 1945 – operation 1951)
First real-time computer (flight simulations and military operations)

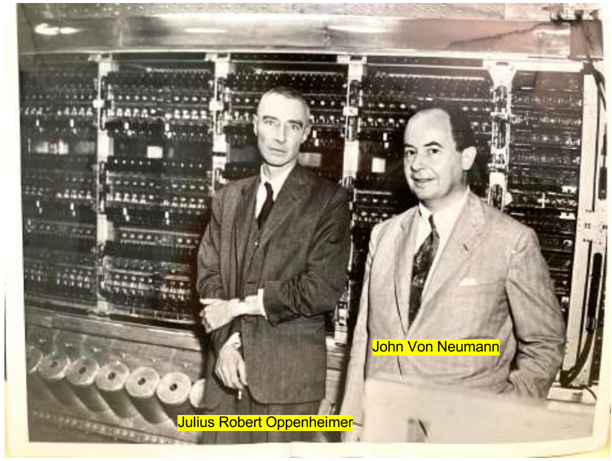
MIT Boston

- ~ 5000 radiovalves
- ~ 3000 bits memory
- ~ Addition ~80µs
- ~ many tonnes

Introduced
Magnetic core memory

Institute for Advanced Study (IAS) computer – John von Neumann

Influential computer architecture, copied by other institutions



Operational June 1952

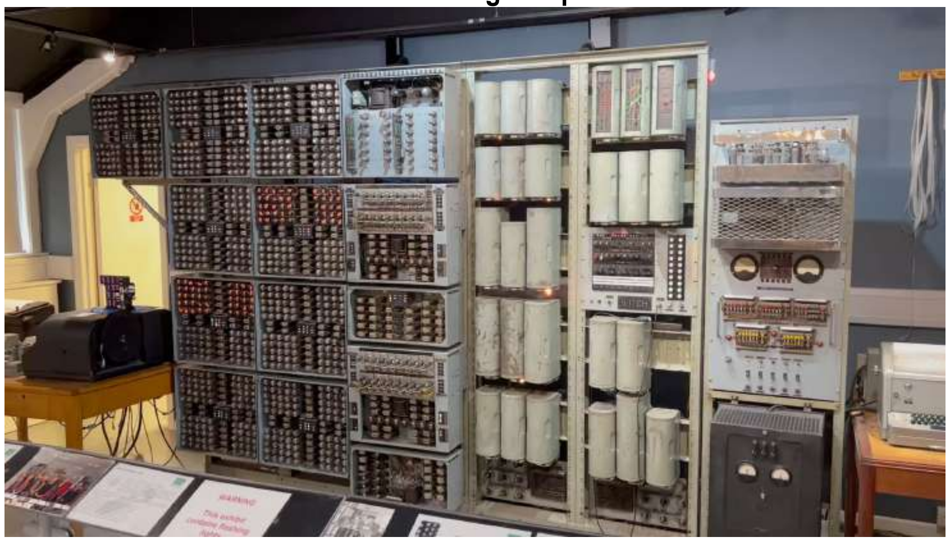
Institute for Advanced Study
Princeton, New Jersey, USA
- ~ 1700 vacuum tubes
- ~ 150kW power consumption
- ~ 20 kHz
- ~ 450 kg
- Stored program computer

Harwell Dekatron aka WITCH, UK, ~1950
The oldest still functioning computer in the world

Atomic Energy Research Esatablishment (AERE)
Harwell, UK
- ~ 200 vacuum tubes
- ~ 480 relais
- ~ 20 memories (~800 dekatrons)

Timing:
Clock ~ 100Hz
Division ~ 10sec

Harwell Dekatron aka WITCH, UK, ~1950
The oldest still functioning computer in the world



Also computers exclusively with relays ….
Fujitsu Japan : FACOM128B (~1952)

- ~ 5000 relais for calculations
- ~ 13000 relais for 13kbit memory

16 digit addition/ subtraction ~0.6s

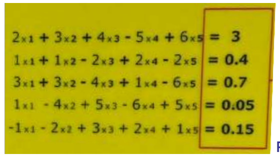
Automatic error correction

Used base 5 system (biquinary)

Early mechanical and electronic calculating machines (1940 -1965)

Also computers exclusively with relays ....
Fujitsu Japan : FACOM128B (~1952)

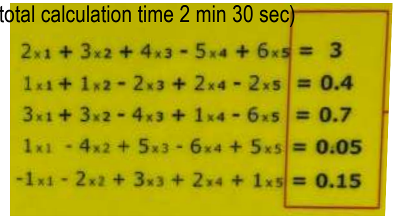
Example:
Solving a system of 5 equations
and 5 unknowns
(total calculation time 2 min 30 sec)

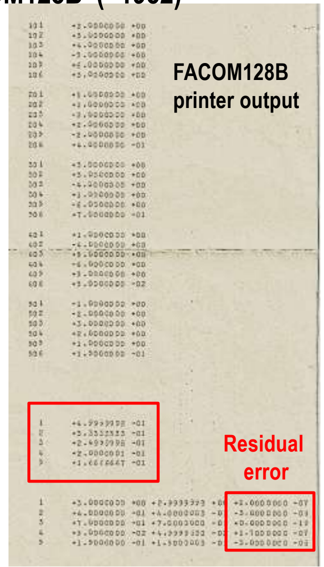


Early mechanical and electronic calculating machines (1940 -1965)

Also computers exclusively with relays ....
Fujitsu Japan : FACOM128B (~1952)

Example:
Solving a system of 5 equations
and 5 unknowns
(total calculation time 2 min 30 sec)

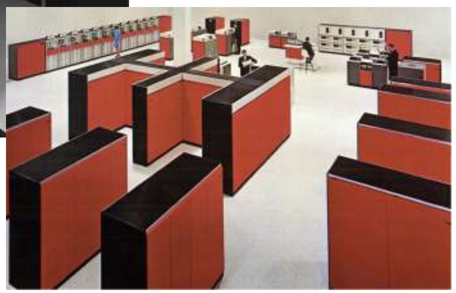




Early mechanical and electronic calculating machines (1940 -1965)

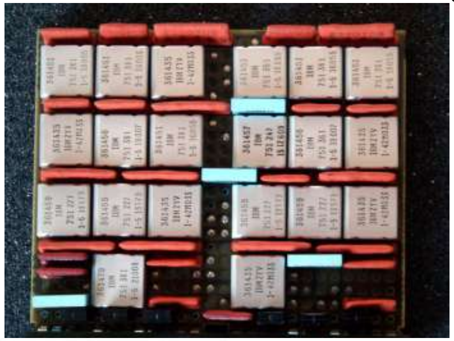
Very famous commercial computer: IBM 360
(announced April 1964)

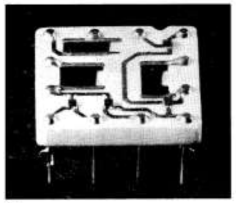
Small and large systems
~ 1 – 18 MHz
~ Solid Logic Technology:
Small modules containing diodes, transistors, resistors



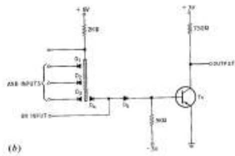
Early mechanical and electronic calculating machines (1940 -1965)

IBM 360 : Solid Logic Technology





(a)

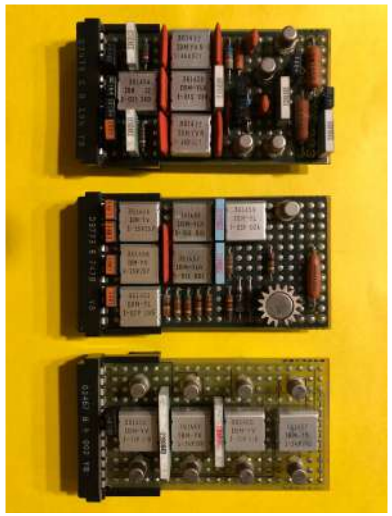


(b)

Power dissipation in mW

	On	Off	
Resistors	28	19	(All R: 5%)
Transistor	7	0	
D_1, D_2, D_3	0	2	
D_4	1	1	
D_5	1	1	
Total	37	23	

Figure 1 AND/OR INVERT **logic module.** (a) Completed AOI module, without overcoating. (b) Logic circuit.



Publication on magnet core memory in n J. Appl. Phys
Invented by Jay W. Forrester, MIT Boston

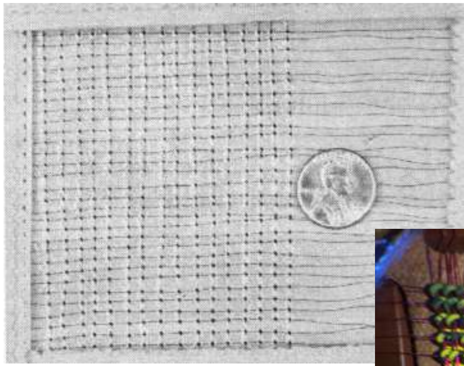
JOURNAL OF APPLIED PHYSICS VOLUME 22, NUMBER 1 JANUARY, 1951

Digital Information Storage in Three Dimensions Using Magnetic Cores*

JAY W. FORRESTER
Servomechanisms Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts
(Received June 5, 1950)

Present digital storage devices use two space coordinates or time and one space coordinate for selection switching, resulting in bulky construction or long access time. Three-dimensional arrays with efficient high speed selection appear possible after continued development of rectangular-hysteresis magnetic materials. An operating mode is suggested which depends on ability of the magnetic material to discriminate between two values of magnetizing force which differ by a 2:1 ratio. Only one magnetic core per binary digit is required. Tests show that most existing metallic magnetic materials switch in 20 to 10,000 microseconds and are too slow. Nonmetallic magnetic materials can now approach the required magnetic behavior; they switch in less than a microsecond.

Magnetic core memory

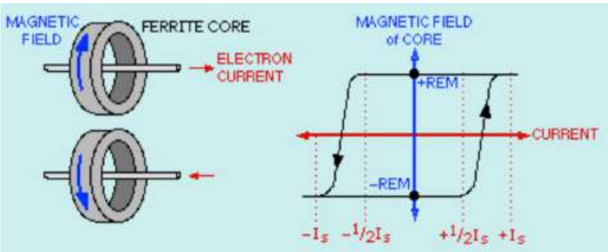


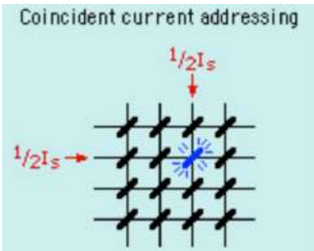
Very early prototype (16 x 30 bits)

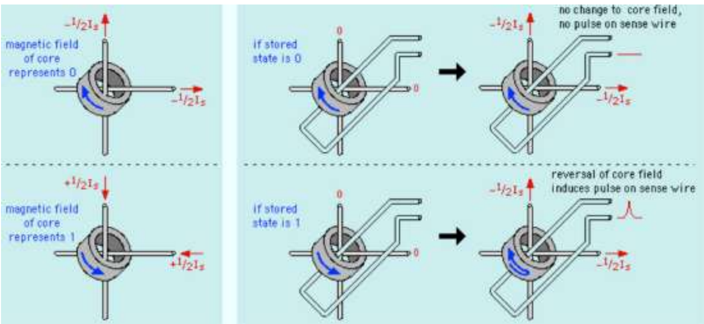
Used in IME84/86S

1 planar array
of IME86S
16 x 16 bits

Magnetic core memory: basics (~ 1950-1975)



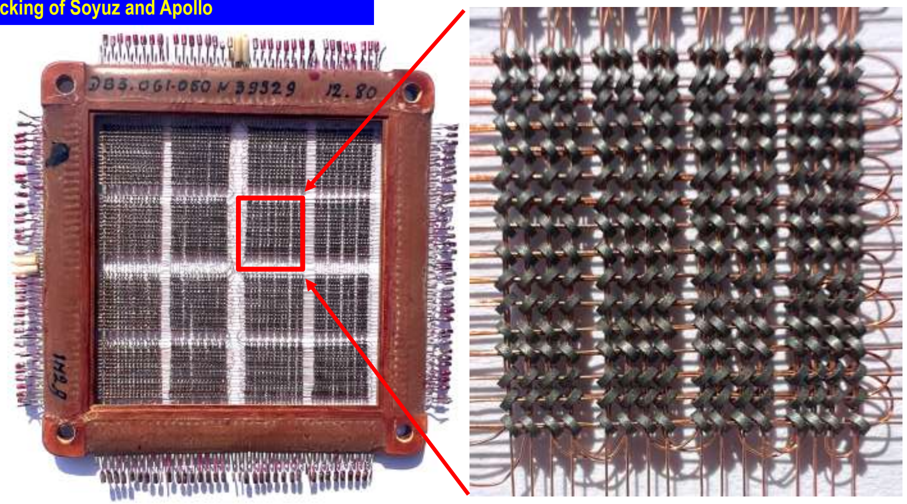




Used in the russian large computer
Большая электронно-счётная машина (БЕСМ)*

*Large electronic calculator

Computer used to perform calculations for the docking of Soyuz and Apollo

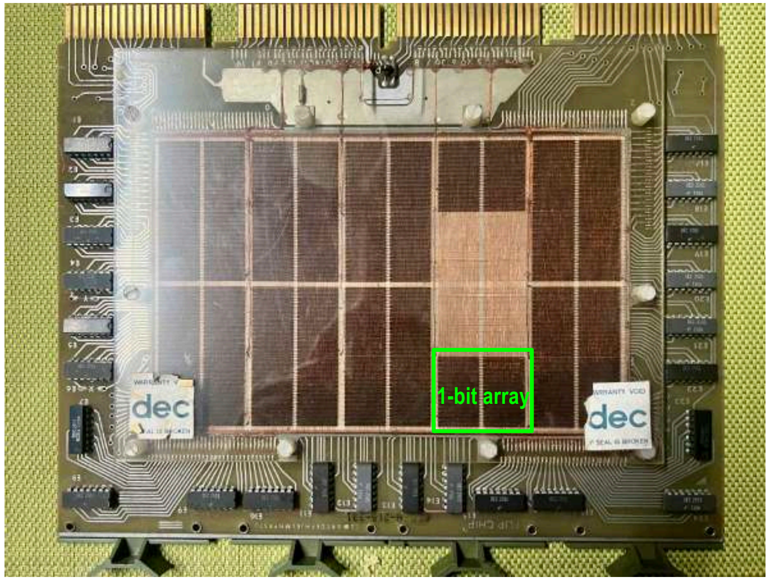


1 planar array 64 x 64 bits

16 x 16 bits

Early mechanical and electronic calculating machines (1940 -1965)

Part of the magnetic core memory of the Digital PDP-11 (1973)

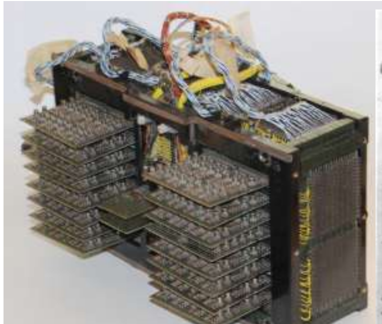


8 kWords
x 18 Bit/Word

(144 000
magnetic cores)

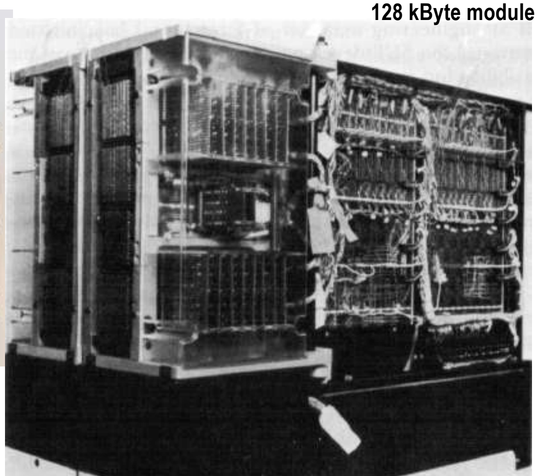
Early mechanical and electronic calculating machines (1940 -1965)

Magnetic core memory for IBM 360



Core subunit

128 kByte module



24-bit addresses on the IBM 360, meaning a maximum of 16 MB; in practice, up to 8 MB of magnetic core memory was installed.

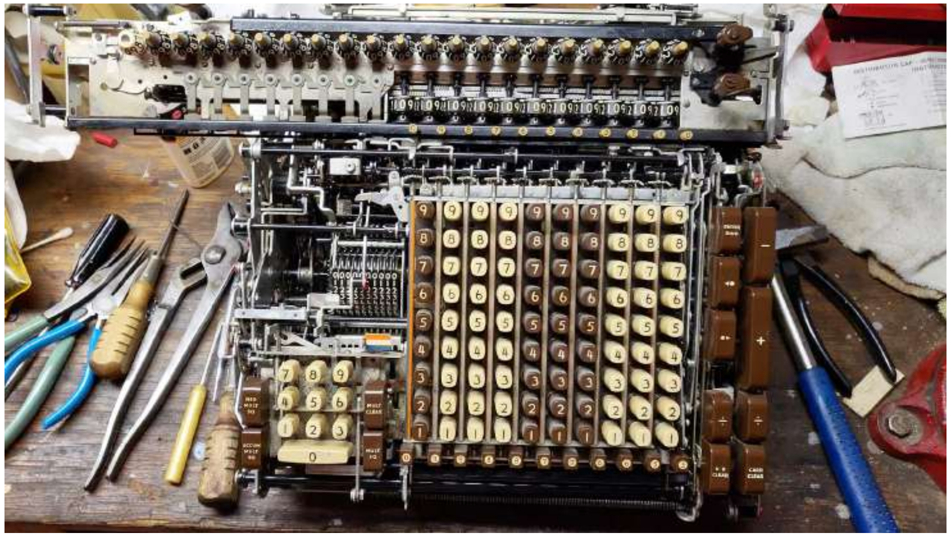
X and Y drivers
transistorized

Early mechanical and electronic calculating machines (1950-1965)

One of the most advanced electromechanical calculators
Friden Model SRQ (~1962)

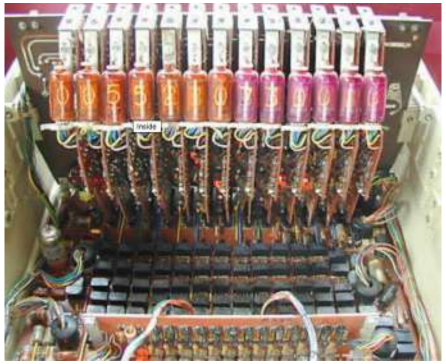
Early mechanical and electronic calculating machines (1950-1965)

Friden Model SRQ calculating sqrt (390625) = 625



Early mechanical and electronic calculating machines (1950-1965)

Anita MkVII (UK, Oct. 1961)



An electronic copy of a mechanical calculator

Used Selenium-diodes (1V forward voltage)

Very simple to detect a broken Se diode:
very pungent, poisonous smell of H_2Se
mix of rotten eggs, sewage, notoriously
long lasting… not recommended…

Diodes, majority of vacuum tubes,
few transistors, dekatrons

The future …. Quantum computing ?

Huge progress
Over the past 5 years

Quantum computing department in FZ-Jülich

Early mechanical and electronic calculating machines (1950-1965)

Industria Macchine Elettroniche (IME84)

(Design 1960 – Production 1964)

Electronic mimic of mechanical calculator

Exclusively Germanium transistors and diodes

Keyboard much more like modern calculators
Operating philosophy is close to that of a mechanical calculator

Early mechanical and electronic calculating machines (1950-1965)

Industria Macchine Elettroniche (IME86S)

(Production 1966)

Much closer to a modern calculator

Exclusively Germanium transistors and diodes

4 user memories, 3 registers, square root, modern operating philosophy, correct DP and sign management

The broken calculating machines
IME-84 and IME-86S

Designed by Massimo Rinaldi in the 1960s
Commercially available 1964 (IME-84) and 1966 (IME-86S)
Produced by the company IME (Industria Macchine Elettroniche)
Located in Pomezia (south of Roma, at the Adriatic Sea)

Both contain hundreds of Germanium diodes and transistors
No Integrated Circuits used

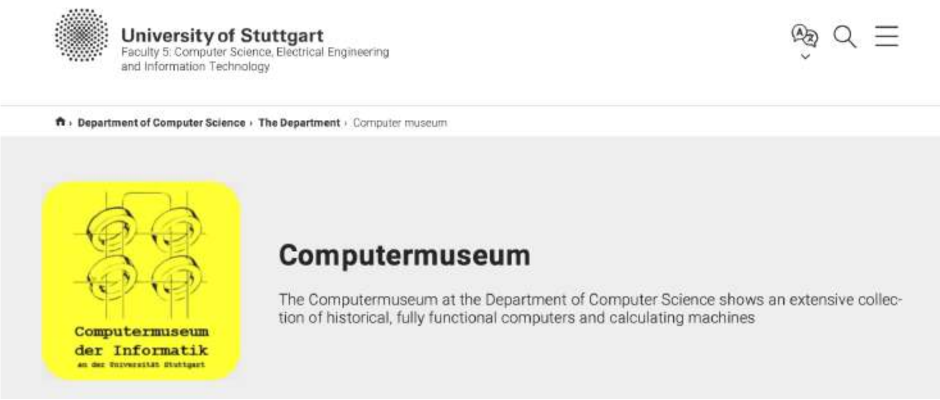
Constructed in so-called Diode AND-OR Logic

Part I
Repairing the IME 84

Aufbau des IME84

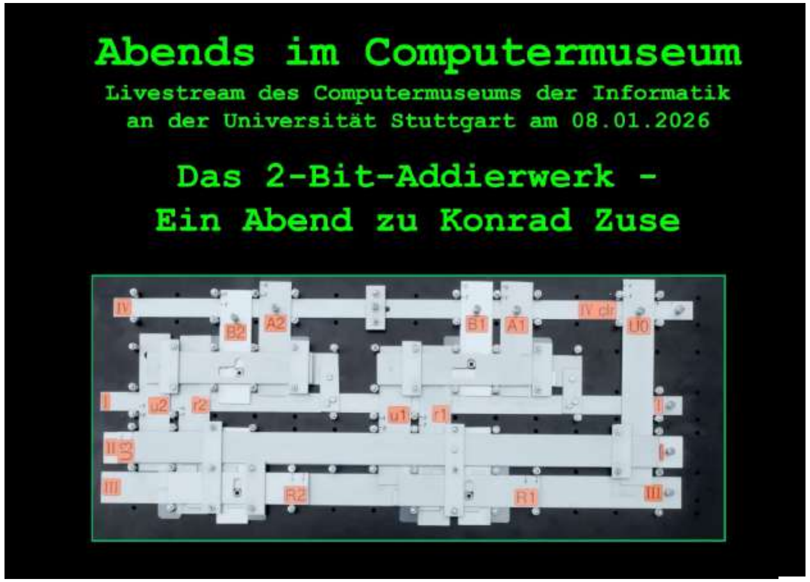
Layout of IME84

Reverse engineering started using documentation from Klemens Krause (Computermuseum Univ. of Stuttgart)



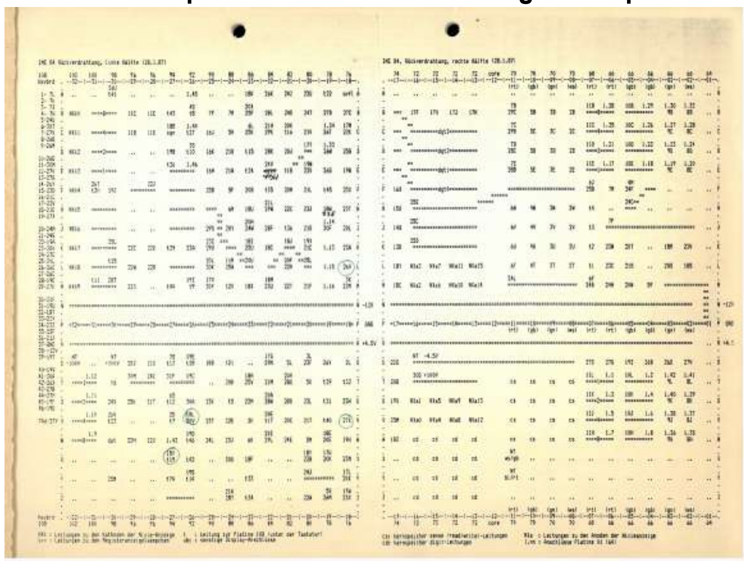
A visit (in person or online) to this museum is highly recommended
See e.g. the youtube overview video https://www.youtube.com/watch?v=FTL_TxrWq4w
and the monthly videos e.g. "Abends im Computermuseum"

Computermuseum Univ. of Stuttgart)
Very interesting museum and informative monthly video's !



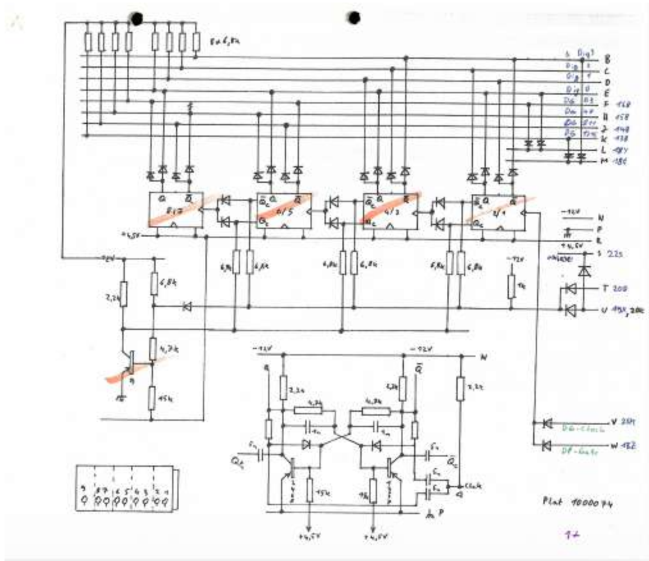
Reverse engineering started using documentation from Klemens Krause (Computermuseum Univ. of Stuttgart)

Backplane interconnections – a great help !



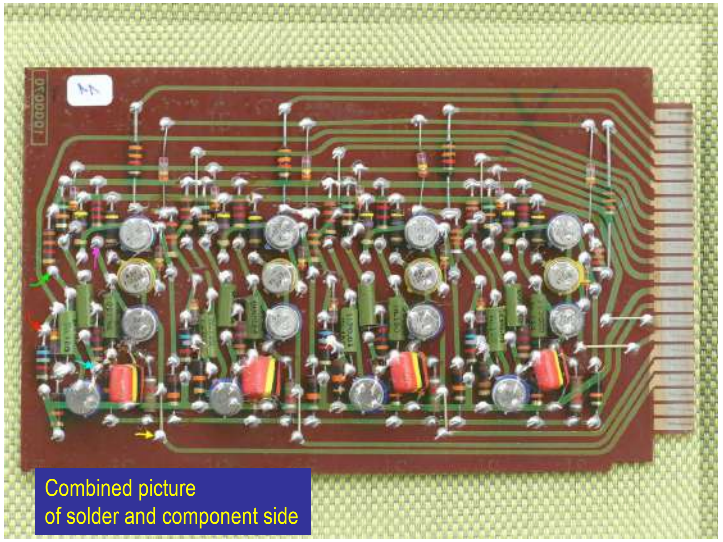
Reverse engineering started using documentation from Klemens Krause (Computermuseum Univ. of Stuttgart)

A large number of pcbs already (partially) reverse engineered



Reverse engineering based upon inspection of individual PCBs

Thanks to the invaluable help of Mr. Brent Hilpert (Vancouver)



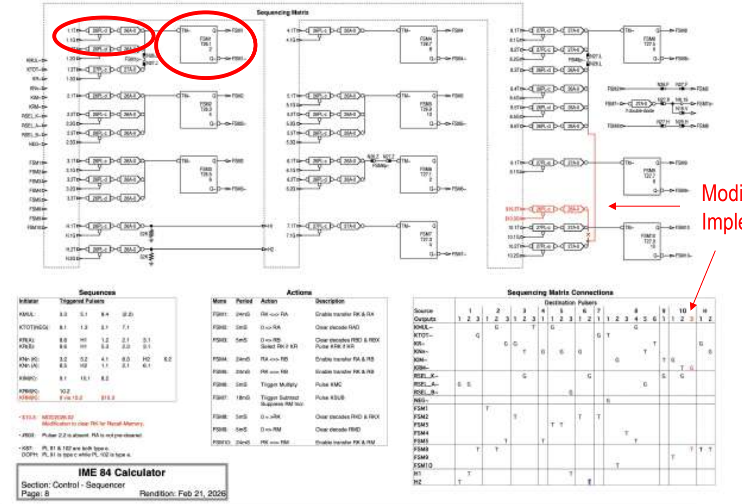
Combined picture of solder and component side

Detailed understanding of the internal machine architecture

State Machine based on Monostables

Gated Pulsers

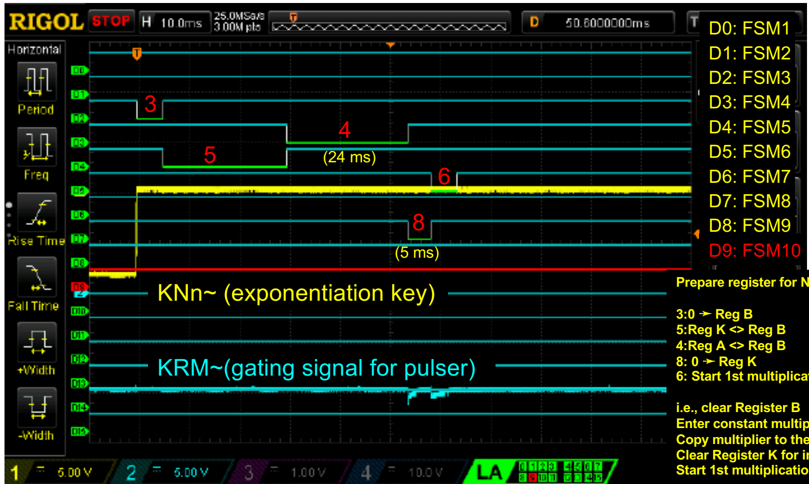
Monostable FF



Modification Implemented

Detailed understanding of the internal machine architecture

Example of sequence (3-5-4-8-6 exponentiation)



Prepare register for N^n^

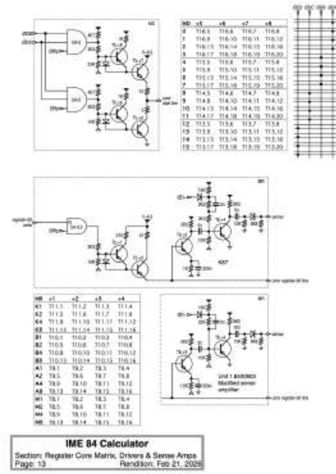
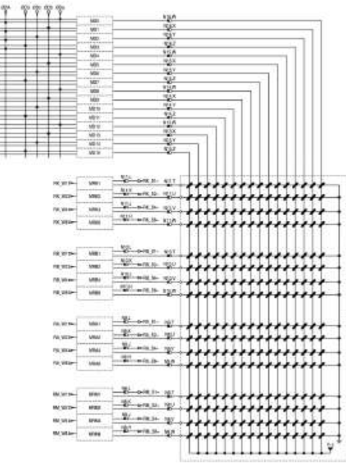
3:0 → Reg B
5:Reg K <> Reg B
4:Reg A <> Reg B
8: 0 → Reg K
6: Start 1st multiplication

i.e., clear Register B
Enter constant multiplier into B
Copy multiplier to the multiplicand
Clear Register K for intermediate results
Start 1st multiplication

Different lenghts for periods (24/5 ms) – not needed for the operation.
Remnants of an earlier development?

A few pages of the full reverse engineering

Magnetic Core Memory

IME84 - The first model of its kind, whose design mimicks that of mechanical calculators

Several imperfections (and variations in production)

Overflow in input is OR'ed with already present input

No clearance of RA before division
- During division RA contains the rest of the division
- A next division should start with RA = 0, otherwise variations in results of division

No clearance of RA before N^n
- Leading to erroneous starting values and results for N^n

No clearance of RK with Recall memory
- RK ≠ 0 : OR of result in RM and RK… thus leading to wrong recalled values in RK

No comma management when inputting a sequence of additions/subtractions…
- Once having pressed the comma, it will continue to shift left for all subsequent numbers being input…
- E.g. 1.23 and then 345 results in 1.2345 …

Limited comma management: difficult to use with non integer values
Negative values not dealt with properly for division and multiplication.
- Reminscent of the operation of mechanical or electro-mechanical calculators (Friden,..)

Similar to the way mechanical or electromechanical calculating machines work (Friden, Merchant,...)

A few examples of the anomalies

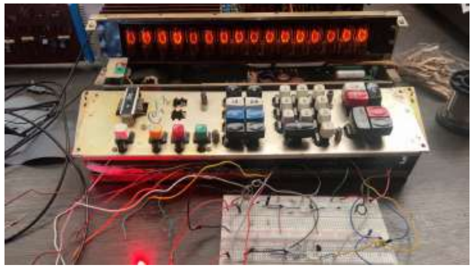
Anomalous comma behaviour when adding

Anomalous comma behaviour for N^n

Anomalous comma behaviour with division

Correct comma behaviour with multipiication

A few examples of the anomalies

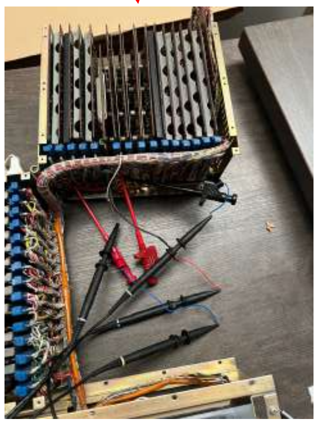


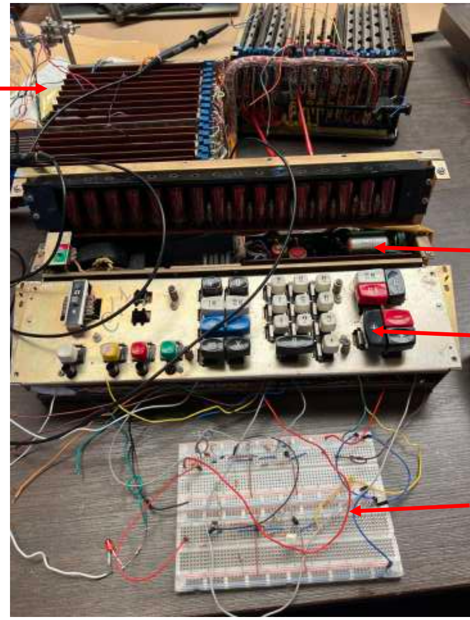
Anomalous N^n due to non zero RA

Anomalous division due to non zero RA

The IME84 during repairs/ investigation

~32 pcbs nicely mounted for acces



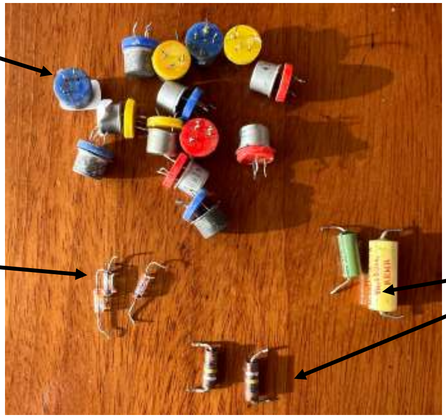


Power supply

Keyboard

Breadboard

Final result – several 'death bodies' found



Improve circuit reliability and user interface of IME84

Improved circuit reliability

Stabilising one of the Finite State Machine states (FSM7)

KNn pulser margins were set too tight, resulting in spurious behaviour.

Improved user interface

Clearance of RA before division

Clearance of RA before N^n

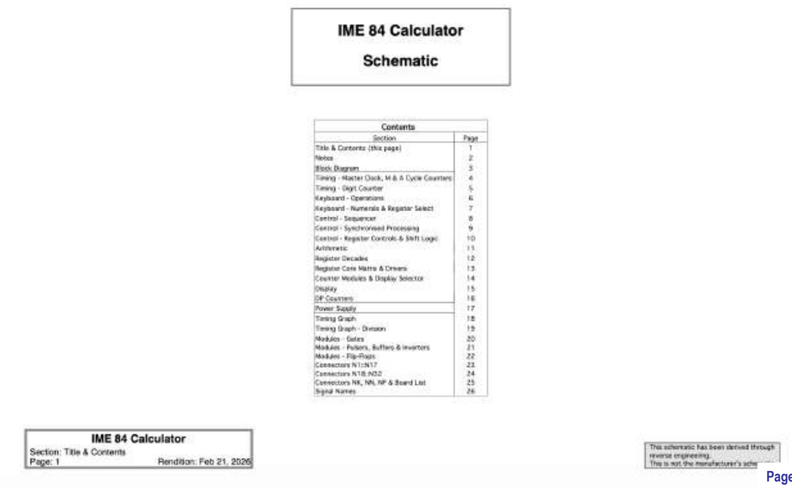
Clearance of RK with Recall memory

Improved comma management when inputting additions or subtractions

A few pages from the complete reverse-engineering analysis

Thanks to the invaluable assistance of Mr. Brent Hilpert (Vancouver)

Circuit diagram of the IME84 (26 pages)



IME 84 Calculator

Schematic

Contents

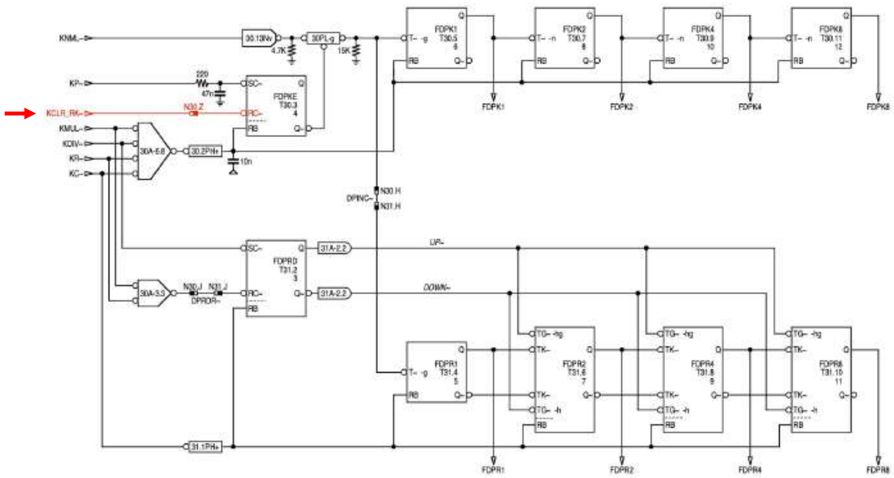
Section	Page
Title & Contents (this page)	1
Notes	2
Block Diagram	3
Timing - Master Clock, M & A Cycle Counters	4
Timing - Digit Counter	5
Keyboard - Operations	6
Keyboard - Numerals & Register Select	7
Control - Sequencer	8
Control - Synchronised Processing	9
Control - Register Controls & Shift Logic	10
Arithmetic	11
Register Decades	12
Register Core Matrix & Drivers	13
Counter Modules & Display Selector	14
Display	15
DP Counters	16
Power Supply	17
Timing Graph	18
Timing Graph - Division	19
Modules - Gates	20
Modules - Pulsers, Buffers & Inverters	21
Modules - Flip-Flops	22
Connectors N1:N17	23
Connectors N18:N32	24
Connectors NK, NN, NP & Board List	25
Signal Names	26

IME 84 Calculator
Section: Title & Contents
Page: 1 Rendition: Feb 21, 2026

This schematic has been derived through reverse engineering.
This is not the manufacturer's sche

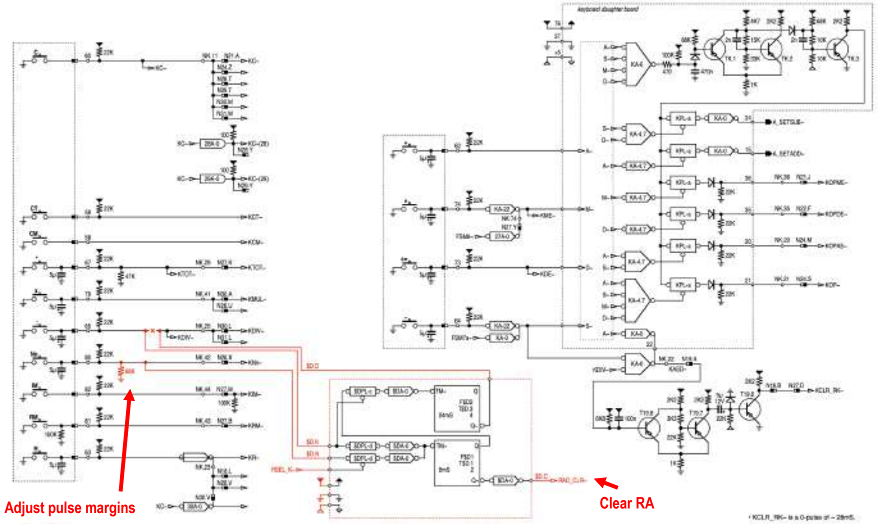
Improve circuit reliability of IME84

Modification to allow the DP to be set & fixed for the duration of a summation.



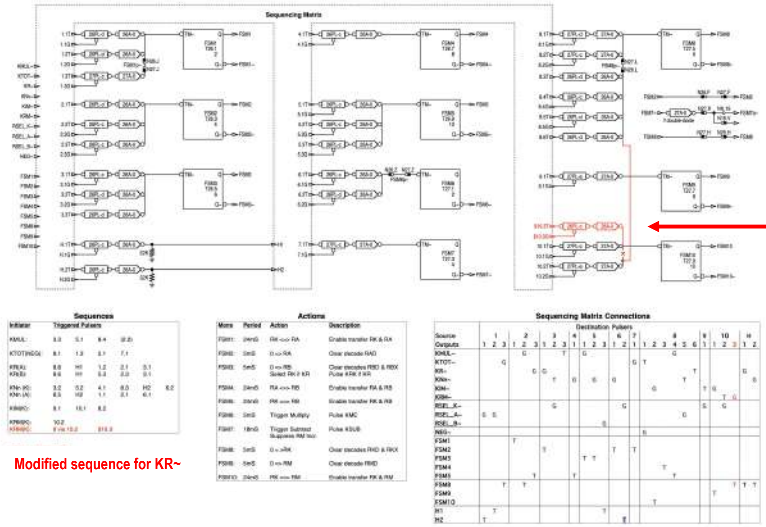
Improvements implemented

Modification to clear RA for division and exponentiation and adjust KNn~ pulse margins.



Improvements implemented

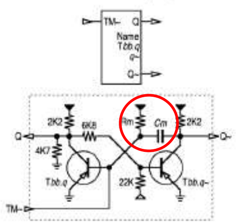
Modification to clear RK before Recall-Memory.



Improvements implemented

Improve stability of FSM7 by adapting resistor/capacitor values for the period of the monostable

Monostable Flip-flops



MONO	Cm	Rm	mS
#903:			
FSM1	470n	68K	24
FSM2	100n	68K	5
FSM3	100n	68K	
FSM4	470n	68K	
FSM5	470n	68K	
FSM6	100n	68K	
FSM7	220n	100K	18
FSM8	100n	68K	5
FSM9	100n	68K	
FSM10	470n	68K	
DOPH:			
FSM6	"	33K	
FSM7	470n	47K	

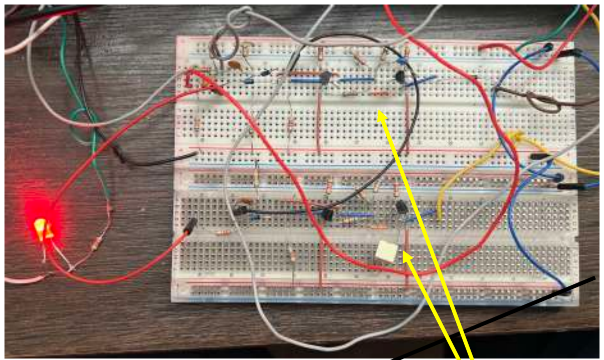
FSM7: MOO196X.A
Stabilise FSM7 by changing
RC values per DOPH above.

Change R to 47 kΩ and C to 470 nF
47 kΩ × 470 nF ≈ 100 kΩ × 220 nF
➛ same period for the monostable flip-flop

However: The lower resistance causes I_b to increase
➛ larger I_b reduces sensitivity to noise / interference with other signals

Final result – needed to implement additional PCB to improve user interface

Clearance of RA before division and exponentiation
Tested on breadboard

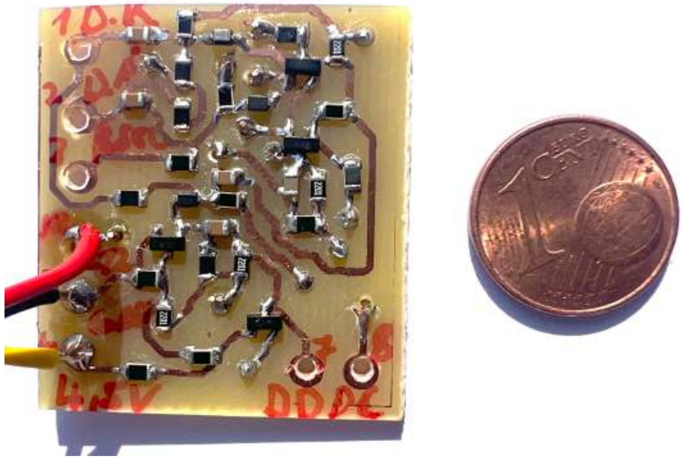


Broken red lamp for reg K replaced by red LED

Two monostables to clear RA before division and N^n

Final result – needed to implement additional PCB to improve user interface

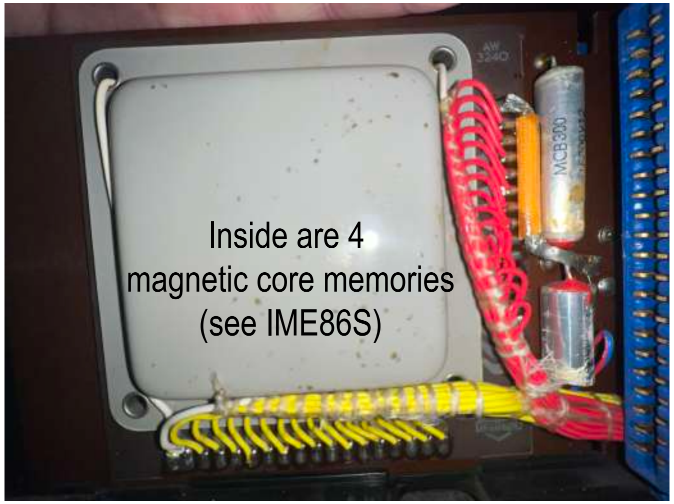
Final version: with SMD components



Magnetic core memory in IME84

1 planar array 16 x 16 bit

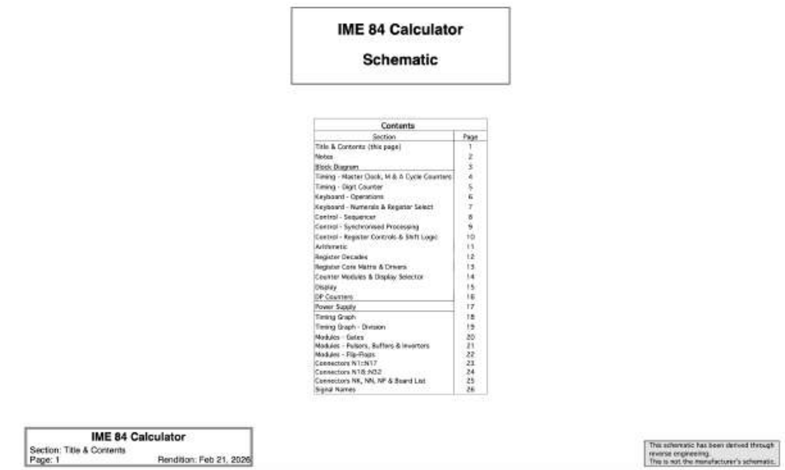
4 registers with 16 digits and 4-bit / digit



A few pages of the full reverse engineering

Thanks to the invaluable help of Mr. Brent Hilpert (Vancouver)

Schematic of IME84 (26 pages)



IME 84 Calculator

Schematic

Contents

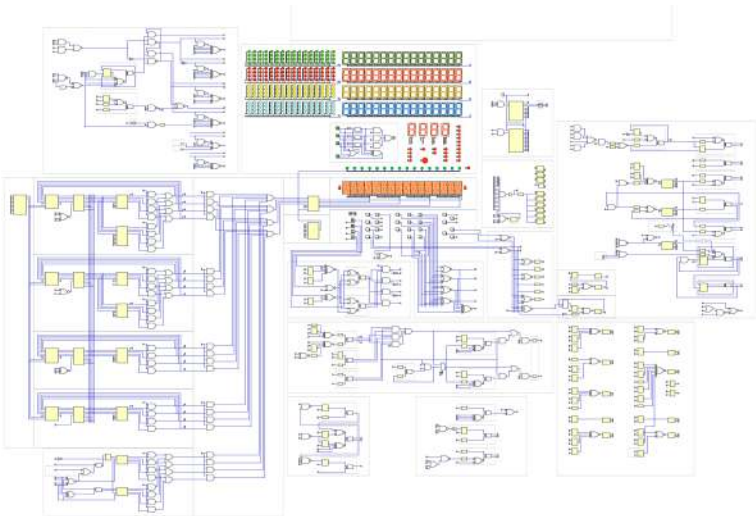
Section	Page
Title & Contents (this page)	1
Notes	2
Block Diagram	3
Timing - Master Clock, M & A Cycle Counters	4
Timing - Digit Counter	5
Keyboard - Operations	6
Keyboard - Numerals & Register Select	7
Control - Sequencer	8
Control - Synchronised Processing	9
Control - Register Controls & Shift Logic	10
Arithmetic	11
Register Decades	12
Register Core Matrix & Drivers	13
Counter Modules & Display Selector	14
Display	15
DP Counters	16
Power Supply	17
Timing Graph	18
Timing Graph - Division	19
Modules - Gates	20
Modules - Pulsers, Buffers & Inverters	21
Modules - Flip-Flops	22
Connectors N1:N17	23
Connectors N18:N32	24
Connectors NK, NN, NP & Board List	25
Signal Names	26

IME 84 Calculator
Section: Title & Contents
Page: 1 Rendition: Feb 21, 2026

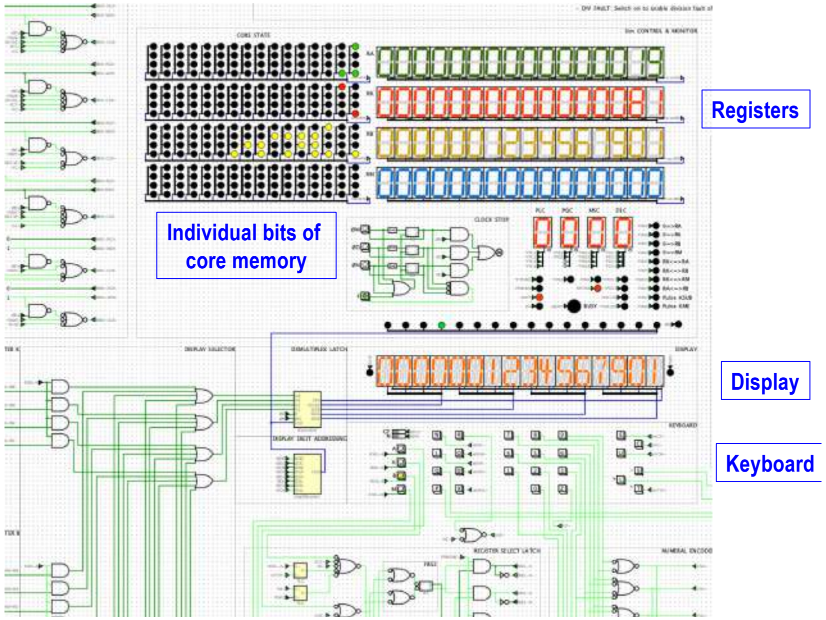
This schematic has been derived through reverse engineering.
This is not the manufacturer's schematic.

Simulation of IME84 using the logic Digital (Prof. Neemann)

Full simulation (based upon the reverse engineering efforts)



Simulation of IME84 using the logic Digital (Prof. Neemann)



A meeting in Trastevere with Stefano Rinaldi (son of the designer !)

Ristorante Il Ciak
Vicolo del Cinque, 21, 00153 Roma RM

Warmly recommended

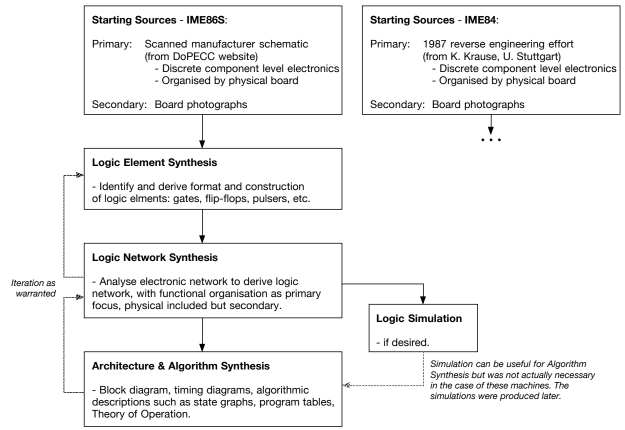
Part II
Repairing the IME 86S

IME86S - A much more developed version of the IME84

The IME-84 and 86 in Comparison

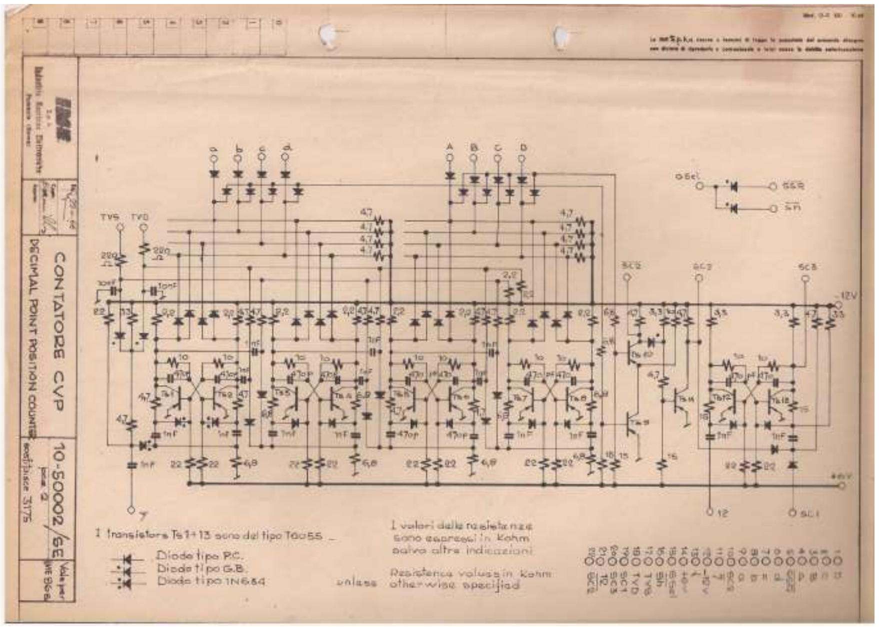
IME-84		IME-86
• Counter-based arithmetic, no binary adder.	→	Same.
• 2D - 2Wire core organisation.	→	2-1/2D - 2Wire core.
• Mixed control architecture, monostable-based sequencer.	→	More cohesive microprogram architecture, QM counter sequencer, synchronised.
• 2-bit subroutine sequence counter for MP &DV.	→	3-bit subroutine sequence counter to additionally perform square root.
• User interface mimics mechanical calculator limitations.	→	User interface improved to take advantage of capabilities with electronics (more-algebraic entry, chaining mixed calculations, etc.).
• 4 number registers.	→	7 number registers.
• Single sign flag, incomplete sign management.	→	Bank of sign flags, proper sign management.
• Weak decimal-point management.	→	Consistent, fixed-position DP management.

Reverse Engineering Workflow
for the IME86S & IME84,
both being Discrete-Component Machines



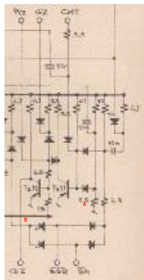
Having prior component-level schematics for the IME machines was helpful, but usually such is not available and the primary source of course becomes the physical machine itself.

Reverse engineering based upon full schematic

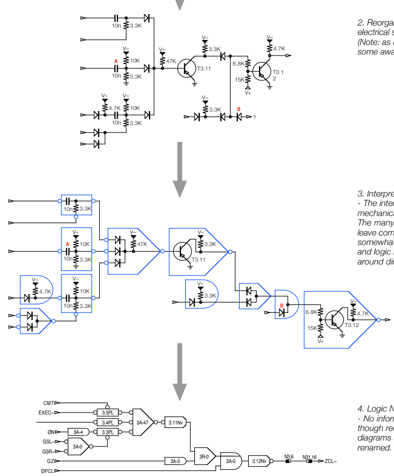


Reverse engineering based upon full schematic

Logic Element & Network Synthesis



1. Electrical Network
IME Original Schematic
portion of page
10-50003_EnterShift



2. Reorganise to make some degree of consistent electrical sense of the network.
(Note: as drawn for this step does already presume some awareness of logic-element structure.)

3. Interpret to logic.
- The interpretion to logic is not always a simple mechanical modular identify-&-substitute.
The many optimisations, tailorings and 'tunings' can leave comparitively large networks of passives to be somewhat obtuse as to the direction of signal flow and logic intention. Note for example, the network around diode B and its resolution to logic.

4. Logic Network
- No information from the original diagram is lost, though requires some reference to element-module diagrams and tables, and some symbols have been renamed.

In practice (and with practice) it is generally not necessary to explicitly draw out the intermediate stages.

Logic implementation of IME86S (and IME84)

- Active Devices: discrete Ge transistors
- Vcc = –12V
- Vbb = +6V, to pull bases below cutoff

- Most gates constructed from discrete Ge diodes and resistors in what may be called "Diode-AND-OR-Logic"
(DARL is a term applied in retrospect, not a term of the period)

- DARL allows a limited cascading of gates without an active device at every stage.

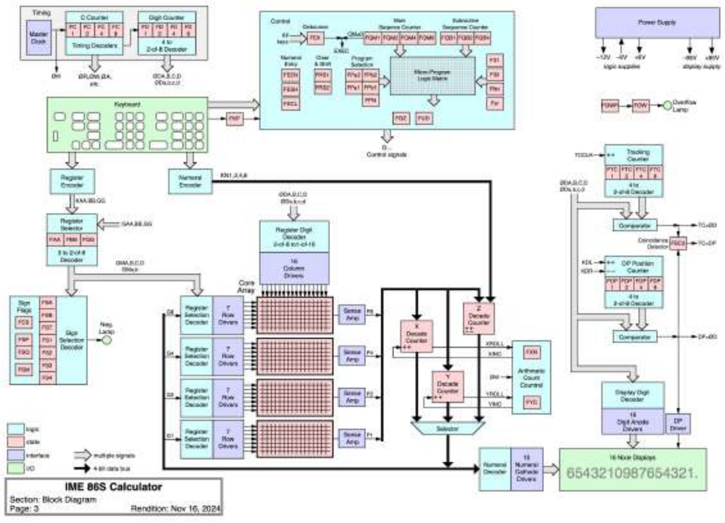
- The general DARL form was common in the discrete days, minimising the number of transistors needed, keeping costs down.

- Numerous optimisations applied to reduce component count, tailored to individual circuits.

- DARL is *not* the same as the integrated era DTL.

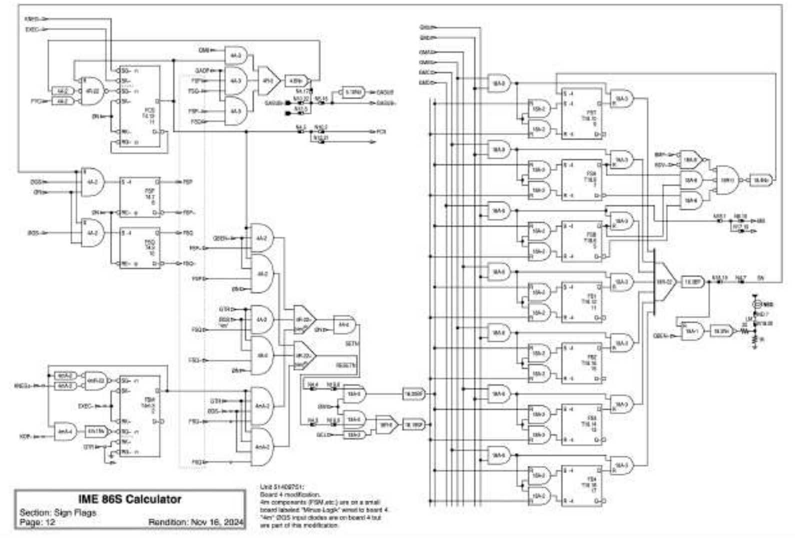
Reverse engineering based upon full schematic

Block diagram of the machine



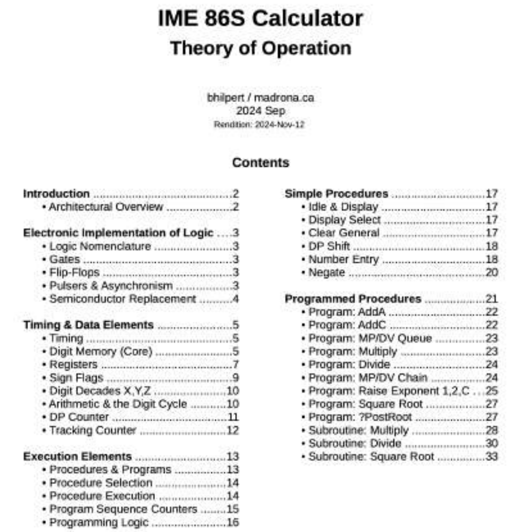
Reverse engineering based upon full schematic

Example page of the schematic



Detailed understanding of the internal machine architecture

Theory of operation IME86S (30 pages) by Mr.Brent Hilpert



IME 86S Calculator

Theory of Operation

bhilpert / madrona.ca
2024 Sep
Rendition: 2024-Nov-12

Contents

Flip-flops and asynchronism

• These machines come near the end of the discrete era. For smaller digital systems such as these, asynchronous practices were still common, to save on the costs of fully synchronous design.

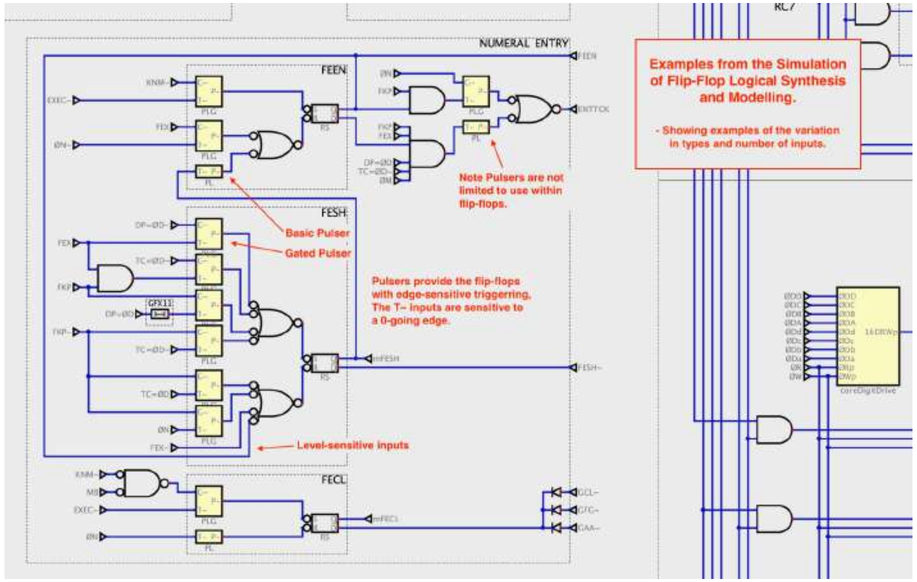
• Flip-flops are capacitor-triggerred with a single active stage (transistor pair). The capacitors provide a degree of storage function, in a sense functioning as a master stage to the active-pair slave stage.

• Capacitors are also used extensively for generating other trigger-pulse actions, outside of flip-flops.

• For logic-synthesis purposes, these capacitive circuits can be modularised into logic elements which will be referred to as 'pulsers'. Pulsers generate a brief pulse when an appropriate edge (transition) occurs on their trigger input.

• Flip-flops do not appear as the standard JK- and D-types prevalent today. Rather, they are essentially SR-types to which a range of input types may be applied, with a varying number and selection of these inputs.

Further details on the variations of the flip-flops used in the IME84



Microprograms (IME86 Control Structure)

• For most procedures, keypresses are recorded into a triple of the most recent 3 keyboard activities, via the Procedure-Select flags (FPa1,2,FPb1,2,etc.).

• E.g. the key sequence:

1 2 X 3 4 =

will be recorded into a state indication:

PEQ•N•PPMP

meaning that Multiply was pressed prior, then a Number entered, then Equals pressed.

• When the final key of this sequence is pressed, this state indicator selects and invokes a microprogram for execution.

• The 4-bit QM counter begins a full 16-state cycle, with microinstructions being executed at selected (not all) QM count-states, per the activated microprogram lines.

• The microinstruction lines activate control lines going out to other functional sections of the calculator, mostly the register and arithmetic sections.

• In time, a microinstruction occupies and is executed across one number cycle. A microprogram then, is effectively a sequence of number cycles, during each of which some action is performed on number registers.

• In this graphic from the simulation, the red circles with T indicate the microprogram triple (as two signals) for PEQ•N•PPMP.

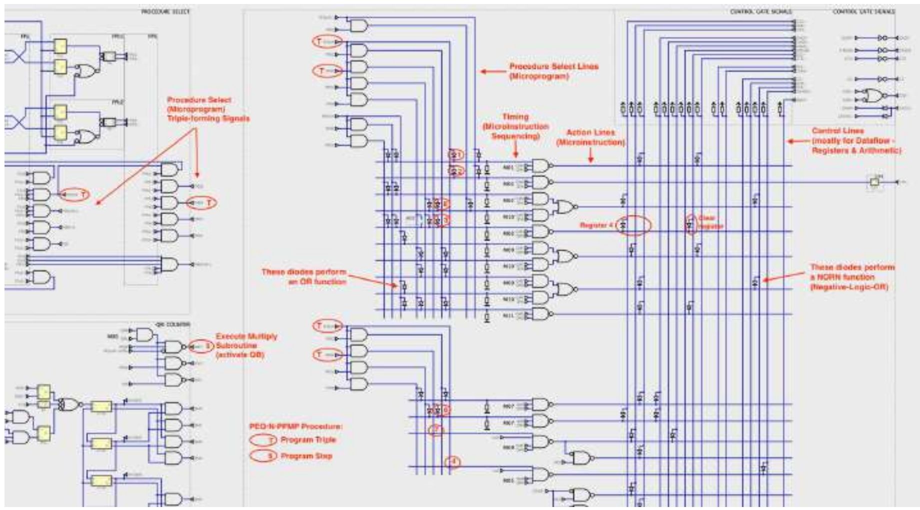
• The red circles with numbers indicate the program steps.
Note this is not the same as the QM count, e.g. program step 8 occurs when QM=10.

• See also the program listing [TOP.23].

• E.g. Step 3 of the microprogram occurs at time QM=3.
The first diode on the action (microinstruction) line selects Register 4 (GGG=1,GBB=0,GAA=0), the second diode activates GCL: during the number cycle of QM=3, Register 4 (RT) will be cleared.

• At step 5 (QM=6), the QB counter is activated to perform the multiply subroutine. QM pauses here while the subroutine executes.

Details of microprograms (IME86 Control Structure)



Repair components

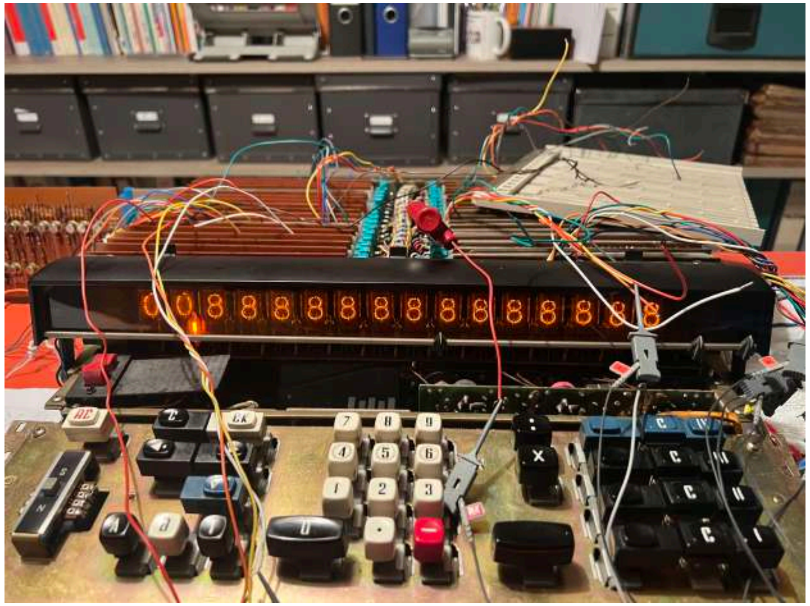
Large number of Ge diodes (total: 1045) and PNP Ge transistors (total: 408)
Most of them no longer available commercially
Known to be unreliable on the long term

Replacements for Ge transistors
PNP silicon transistor close equivalents

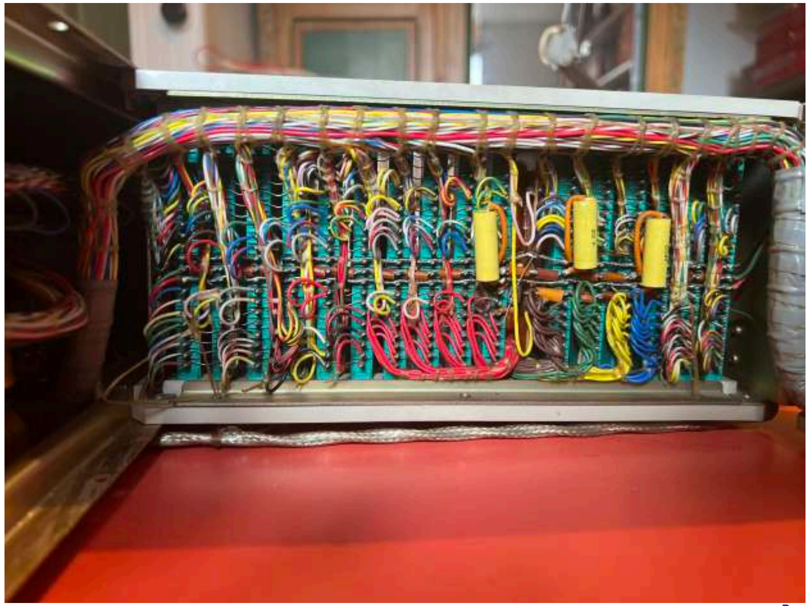
For logic operations : **2N3906**
For drivers of the core memory : **2N2907**
Nixie drivers (voltages > 100V) : **MPSA92, 42**

Replacements for Ge diodes
more delicate : 0.3V forward voltage (Ge) vs 0.7V for (Si)
BAT46 Schottky diode with 0.3V forward voltage and nearly identical V-I characteristic

The machine during repairs

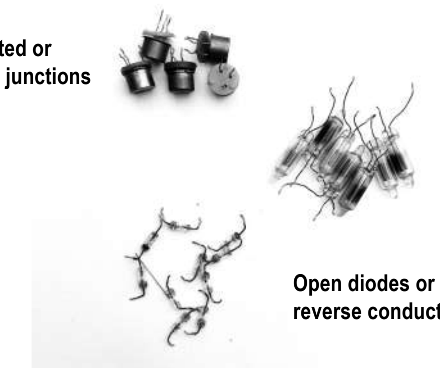


Very nice cabling of the pcbs; two halves that can easily be folded open

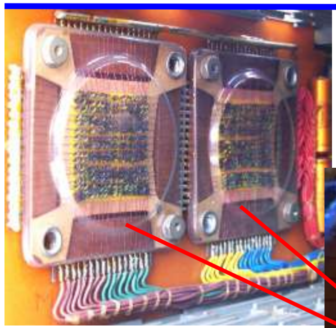


Result of the repair…

Several "corpses" discovered and repaired

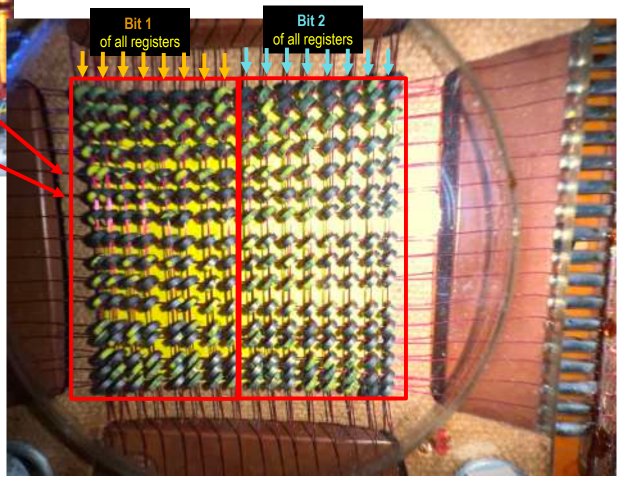


Magnetic core memory modules of IME86S



Bit 1 of all registers
Bit 2 of all registers
Bit 4 of all registers
Bit 8 of all registers

16 digits
4 bits per digit

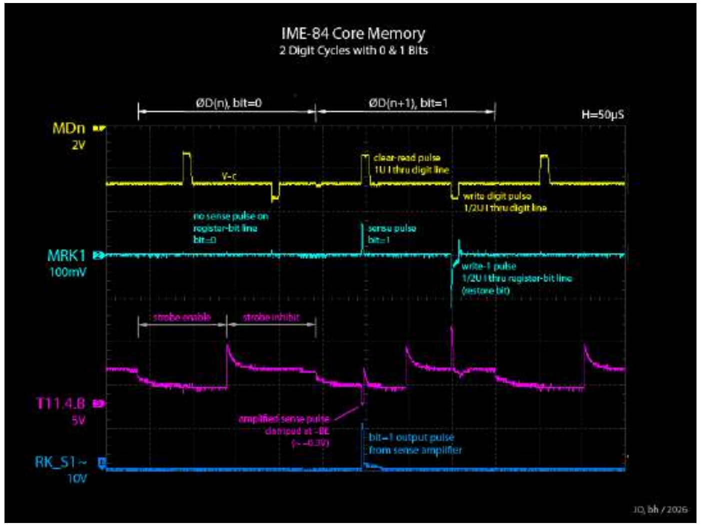


2 planar arrays 16 x 16 bit

Registers with 16 digits
4-bit / digit

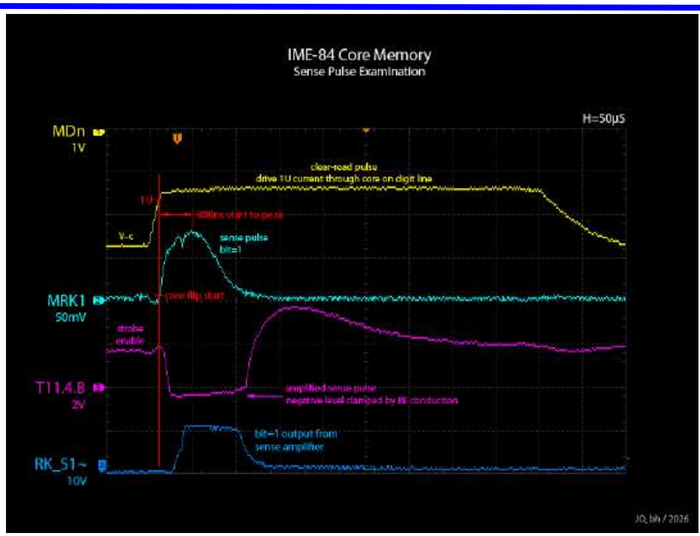
7 registers used, 1 spare

Speed and operation of core memory (from IME84)



To determine the status of a bit: apply a current pulse to set it to zero
The 1-bits produce a clear sense pulse, are converted to 0-bits and must therefore be restored!
A dedicated write pulse immediately afterwards is required to restore the 1 bits

Speed and operation of core memory (from IME84)



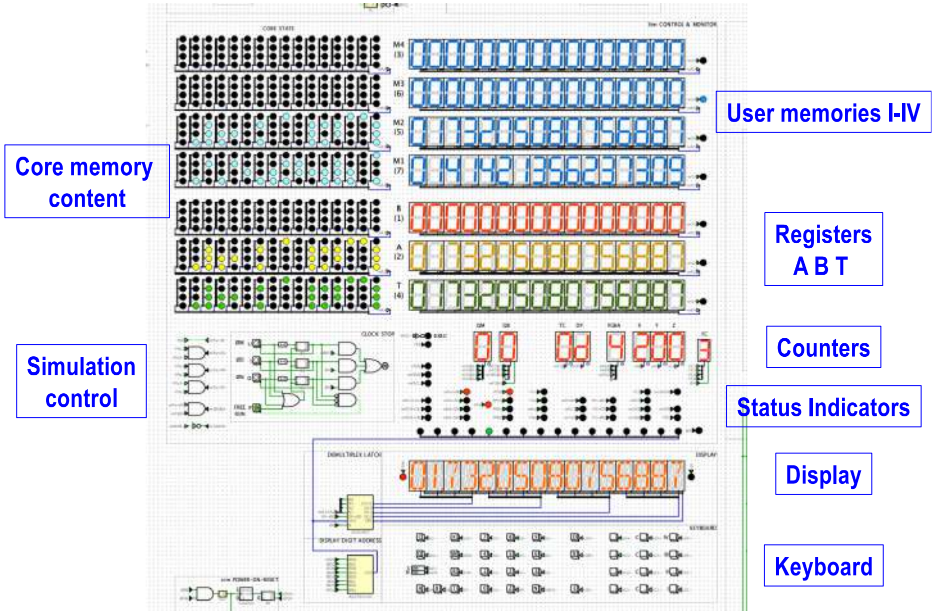
To determine the state of a bit: apply a current pulse to set it to zero
Switching time of a magnetic core ~ 1–2 µs

Example of square root extraction (adapted Toepler method)

3 10 01 1:	2		2			0	Subtract	0 0	
4 10 01 0:	2		2		A	1	subsequent	0 0	SQRT(2)
5 10 01 1:	2		1			1	odd numbers	0 0	
5 10 01 0:	2		1	1-3 = -2		3		0 1	
5 10 01 1:	2	B	-2	➛ neg		3		0 1	
5 10 01 1:	2		1	add back 3		3		0 1	
6 11 01 1:	2		1			3		1 0	
7 12 01 1:	2		1		C	2	Last subtracted odd	1 0	
0 12 01 1:	2		10			2	number -1	1 0	
1 12 01 1:	20		10			2		1 0	1
2 12 01 1:	20		100			2		1 0	
3 12 01 1:	200		100			2		1 0	
4 12 01 0:	200		100		D	21	* 10 and add 1	1 0	
5 12 01 1:	200		79			21		1 0	
5 12 01 0:	200		79			23	Subtract	1 1	
5 12 01 1:	200		56		A	23	subsequent	1 1	
5 12 01 0:	200		56			25	odd numbers	1 2	
5 12 01 1:	200		31			25		1 2	
5 12 01 0:	200		31			27		1 3	
5 12 01 1:	200		4			27		1 3	
5 12 01 0:	200		4	4-29 = - 25		29		1 4	
5 12 01 1:	200	B	-25	➛ neg		29		1 4	
5 12 01 1:	200		4	add back 29		29		1 4	
6 13 01 1:	200		4			29		14 0	14
7 14 01 1:	200		4		C	28	Last subtracted odd	14 0	
0 14 01 1:	200		40			28	number -1	14 0	
1 14 01 1:	2000		40			28		14 0	
2 14 01 1:	2000		400			28		14 0	
3 14 01 1:	20000		400			28		14 0	
4 14 01 0:	20000		400		D	281	* 10 and add 1	14 0	
5 14 01 1:	20000		119			281		14 0	
5 14 01 0:	20000		119	119-283 = - 164	A	283		14 1	
5 14 01 1:	20000	B	-164	➛ neg		283		14 1	
5 14 01 1:	20000		119	add back 283		283		14 1	
6 15 01 1:	20000		119			283		141 0	141
7 0 00 1:	20000		119		C	282		141 0	

square root = 141

Simulation of IME86S



Summary

A full reverse engineering and detailed understanding of the logic of two early fully transistorized calculators has been realized

Has allowed :

Full repair / Implementation of improvements

Full simulation of both IME84 and IME86S

Thanks to the philosophy “Defeat is not an option”…. (Churchill…)
and the dedication, knowledge and meticulous study of the calculator by Mr. Brent Hilpert (bhilpert@shaw.ca) in Vancouver.
All work done exclusively by transatlantic information exchange over the internet !

All info is now available at his website : madrona.ca

Not only calculator info, but also very didactic explanations on other topics e.g. vintage memory systems, vintage computing systems, …

Questions / Discussion

Thank you for your attention !