Computer Architecture

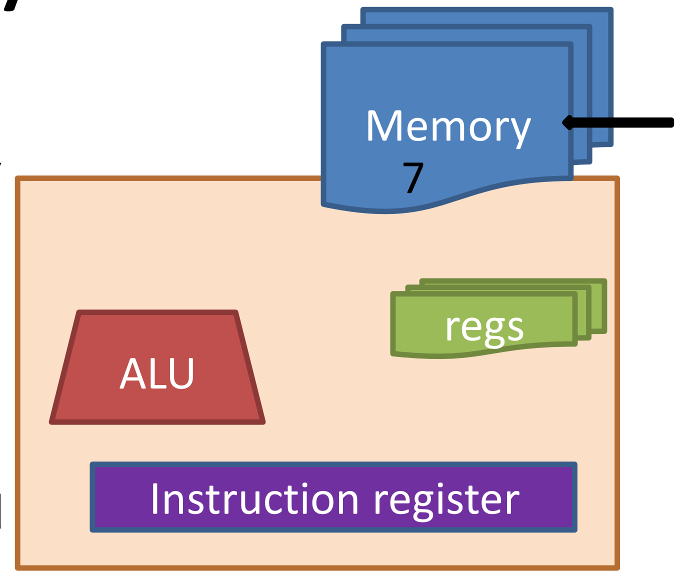
Dr. Esti Stein
(Partly taken from Dr. Alon Schclar slides)

Based on slides by:
Prof. Myung-Eui Lee
Korea University of Technology & Education
Department of Information & Communication

Taken from: **M. Mano/Computer Design and Architecture 3rd Ed.**

Execution Cycle

Instruction Fetch	Obtain instruction from program storage in memory
Instruction Decode	Determine required actions and instruction size
Operand Fetch	Locate and obtain operand data
Execute	Compute result value or status
Result Store	Deposit results in storage
Next Instruction	Determine next instruction (not the next in case of **branch**)

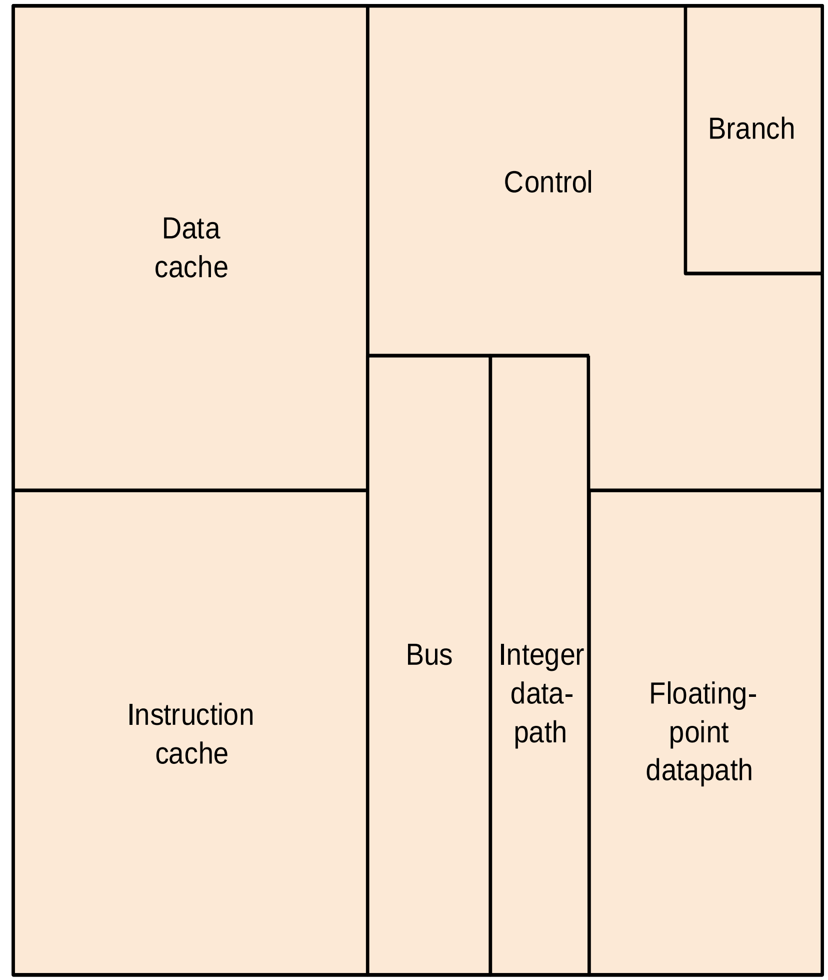


The Processor

- The “heart” (and brain) of the computer

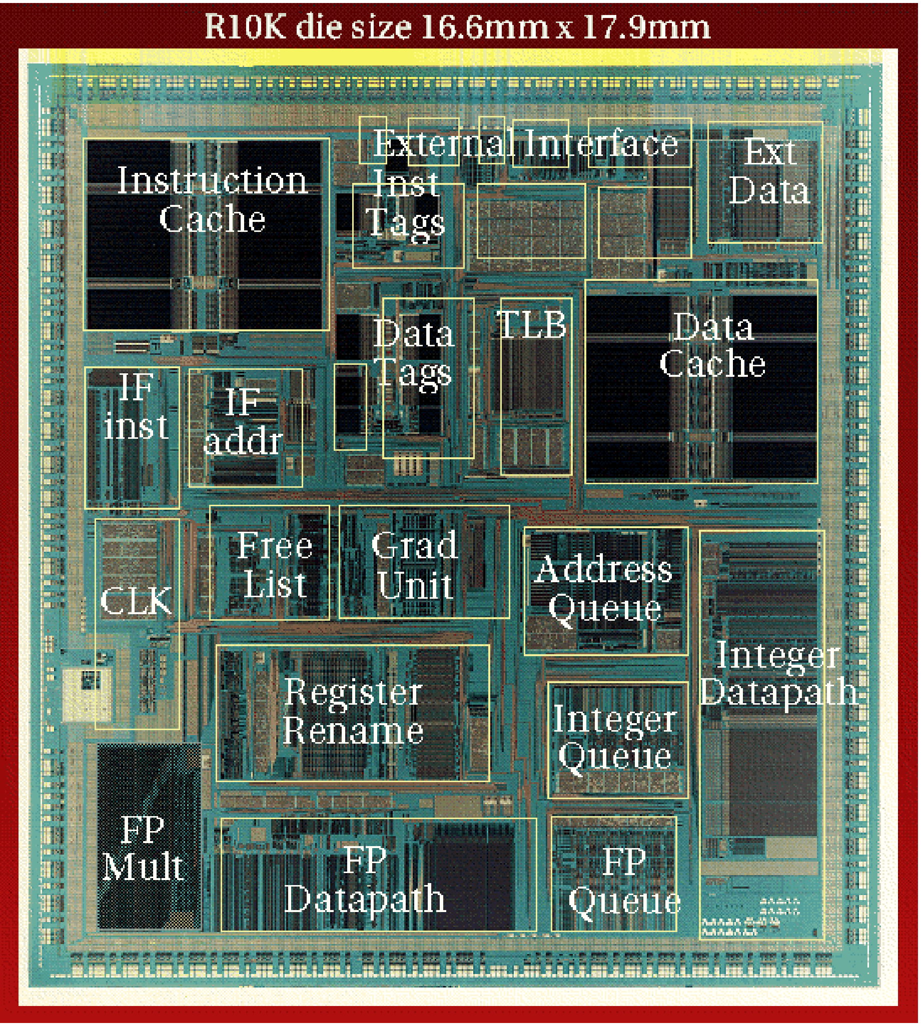
- Schedules instruction execution (Control)

- Executes the instructions (Datapath)

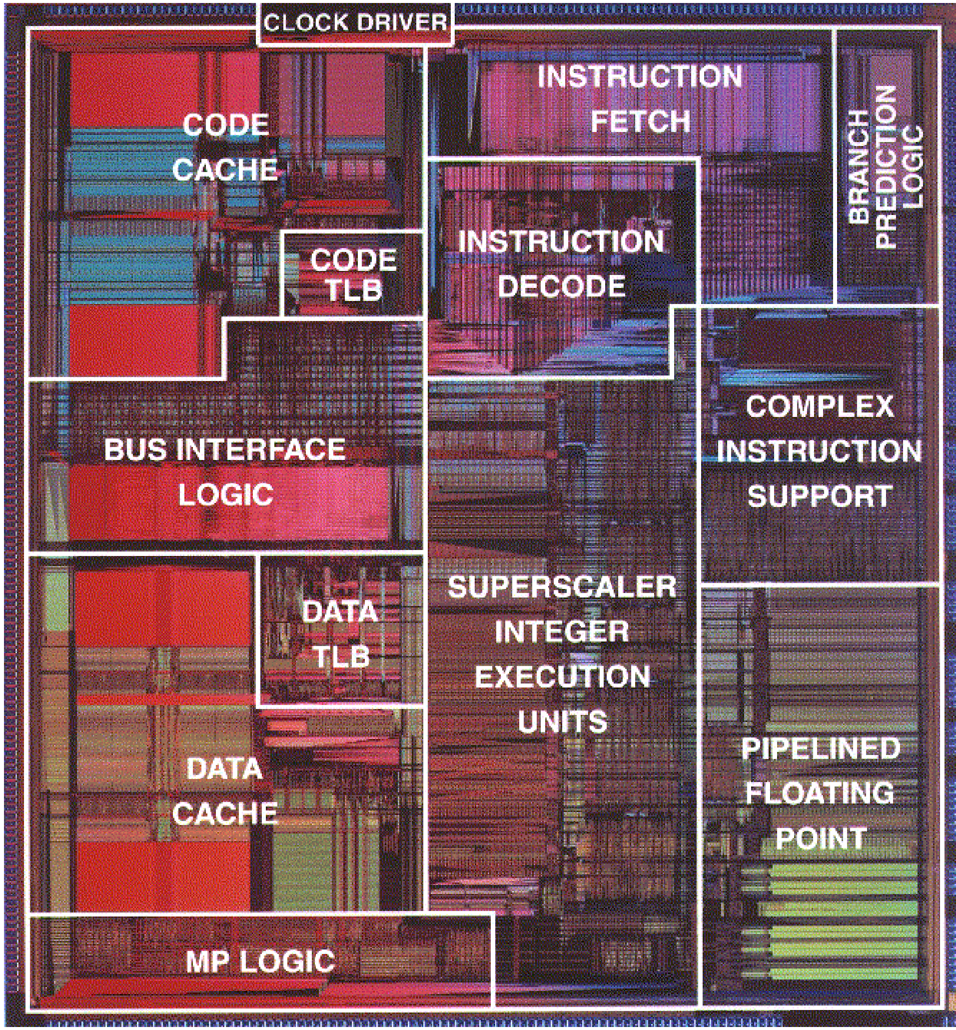


Schematic layout is of an Intel Pentium processor

Photos of Dies



MIPS R10000



Intel Pentium

Modern Microprocessors

- **Intel – Xeon, Pentium core2duo/Quad, Atom, Core i7, i5, i3**
- **AMD – Phenom, Athlon, Sempron**
- **Silicon Graphics - MIPS R4400, R5000, R8000, R10000, R12000**
- **Compaq (formerly Digital) - Alpha 21064, 21164, 21264, 21364.**
- **Motorola, Apple, IBM - PowerPC 601, 603, 604, 604e, 740, 750**
- **Sun - Sparc, HyperSparc, UltraSparc I/II/III**

Technology Development

Design with current technologies

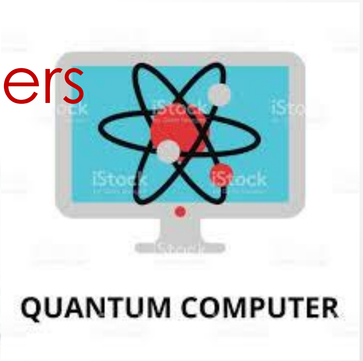
Design

Obsolete computers

Anticipate future technologies

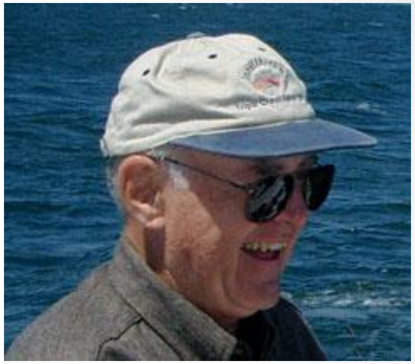
Design

Future computers



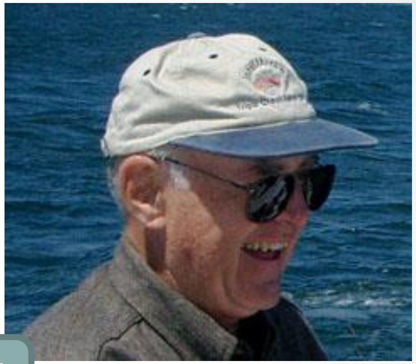
Moore's Law

Every ≈ 2 years the number of transistors in a chip is doubled



- The observation is named after Gordon Moore.
- He was the co-founder of Fairchild semiconductor and Intel.
- His 1965 paper described a doubling every year in the number of components per integrated circuit.
- He revised the forecast to doubling every two years.

Moore's Law



Every ≈2 years the number of transistors in a chip is doubled

1982 - Osborne Executive portable computer : Z80, 4MHz CPU

2007 - Apple iPhone: ARM11, 412 MHz CPU

Moore's Law

Every ≈ 2 years the number of transistors in a chip is doubled

Technology

Processor speed $\times 2$ every ≈ 2 years

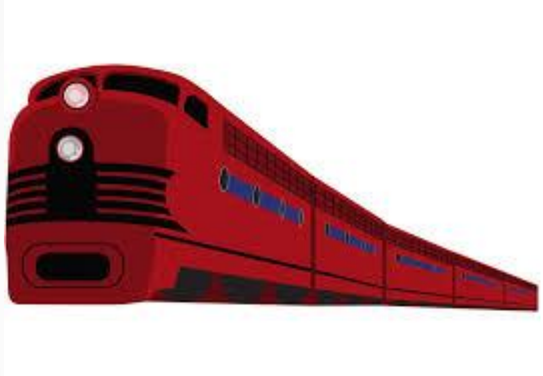
Energy/operation $\times$ ½ every ≈ 2 years

Memory capacity $\times 2$ every ≈ 2 years

Computer architecture

QUIZ

Suppose that **trains** are also doubling the speed every two years:



In 1980 the speed was 300 km/h

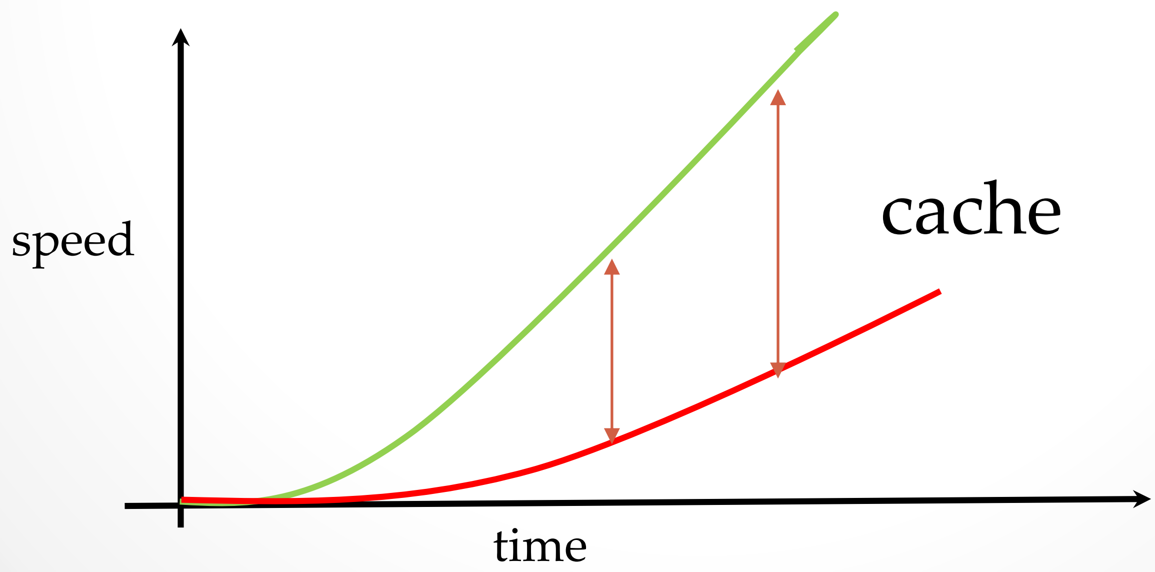
In 2020 the speed is ___________ km/h

Memory Wall

Instruction /sec × 2 every ≈2 years

Memory capacity × 2 every ≈2 years

Memory latency × 1.1 every ≈2 years



Volatile Memory

- The main memory holds its content as long as the power is **on**
- Turning **off** the power will cause the data in memory to be **lost**
- Memory that “forgets” its contents is called ***volatile***

Dr. Alon Schclar, Academic College of Tel Aviv-Yaffo, 2016

Nonvolatile Memory

- In **contrast**, the content of a ***hard drive*** and ***Disk-on-key*** is maintained even without power
  - A hard drive is **magnetic**
  - A disk-on-key is constructed from ***persistent*** memory building blocks (**Flash**)
- These are ***nonvolatile*** memory devices
- Used to store programs and data **between computer ON periods**

Dr. Alon Schclar, Academic College of Tel Aviv-Yaffo, 2016

Storage Devices

- The volatile memory which holds the program while it is running is called the ***main*** or ***primary memory***
- The nonvolatile memory used to store the programs between runs is called ***secondary memory***

Dr. Alon Schclar, Academic College of Tel Aviv-Yaffo, 2016

Why use volatile memory at all?

- Volatile memory access time is < 50ns (10^{-9})
  - Cache memory access time is < 10 ns
- Hard disk access time is ~ 10ms (10^{-3})
  - Making volatile memory **6 orders** of magnitude (1,000,000) times faster
- So why don't we use volatile hard disks?
  - **Cost**: A gigabyte of **disk** is > 100 times **cheaper** than a gigabyte of DRAM
  - of course electric current will have to be maintained
- DRAM sizes for desktop systems are **1GB-1TB**
- Hard disk sizes are **500GB – 2TB**

Flash memory

- A non-volatile memory device that
  - is **not magnetic**
  - Smaller and lighter than HDs
  - Does not contain mechanical parts
    - **Robust** to severe shock and temperature changes
    - **Low energy** consumption
    - Reliable and thus popular in **military** systems
  - First appearance: **M-systems** Disk-on-key
    - A startup from Kfar-Saba (purchased by SunDisk in 2006)

Flash memory

- **Trends**: replace magnetic HDs by **SSDs** - **S**olid **S**tate **D**rives
- A small obstacle: **cost**
  - **$260 - SSD 1TB**
  - **$47 - notebook HD 1TB**
  - **$120 - USB 3 HD 4TB (external)**
  - **$113 - SATA HD 4TB (internal)**

http://www.pricewatch.com/hard_removable_drives/

Dr. Alon Schclar, Academic College of Tel Aviv-Yaffo, 2016

I/O Devices

<u>Three categories</u>

- Input - keyboard, mouse, joystick, scanner, CDROM, microphone, digital camera ...
- Output - printer, display, speakers ...
- Input/Output - hard disks, DVD-R/CD-R, Disk-on-Key, touch screen, Interactive keyboard ...

The Graphic Display

- The output image is composed of a matrix of picture elements, or ***pixels,*** which can be represented as a matrix of bits called a ***bitmap***.
- This **image resides** on the **graphic card** which is the interface between processor and memory to the display.
- **Resolution** – the maximal matrix size that can be displayed

Display sizes and resolutions

	x (width)	y (height)	Mega pixels	Aspect Ratio	Percentage of difference in pixels					Typical sizes	Non-wide version	Uses
Name					Wide XGA	WSXGA	WSXGA+	WUXGA	WQXGA			
Wide XGA	1366	768	1.05	1.78	0%	−19%	−41%	−54%	−74%	15"–19"	XGA	Normal use; viewing 720p (1280x720) video content.
WSXGA Wide XGA+	1440	900	1.3	1.6	24%	0%	−27%	−44%	−68%	15"–19"	XGA+	
WSXGA+	1680	1050	1.76	1.6	68%	36%	0%	−23%	−57%	20"–22"	SXGA+	
WUXGA	1920	1200	2.3	1.6	120%	78%	31%	0%	−44%	23"–28"	UXGA	Viewing 2 full pages of text side by side; viewing 1080p (1920x1080) video content.
WQXGA	2560	1600	4.1	1.6	290%	216%	132%	78%	0%	30"+	QXGA	Advanced graphic design; other professional

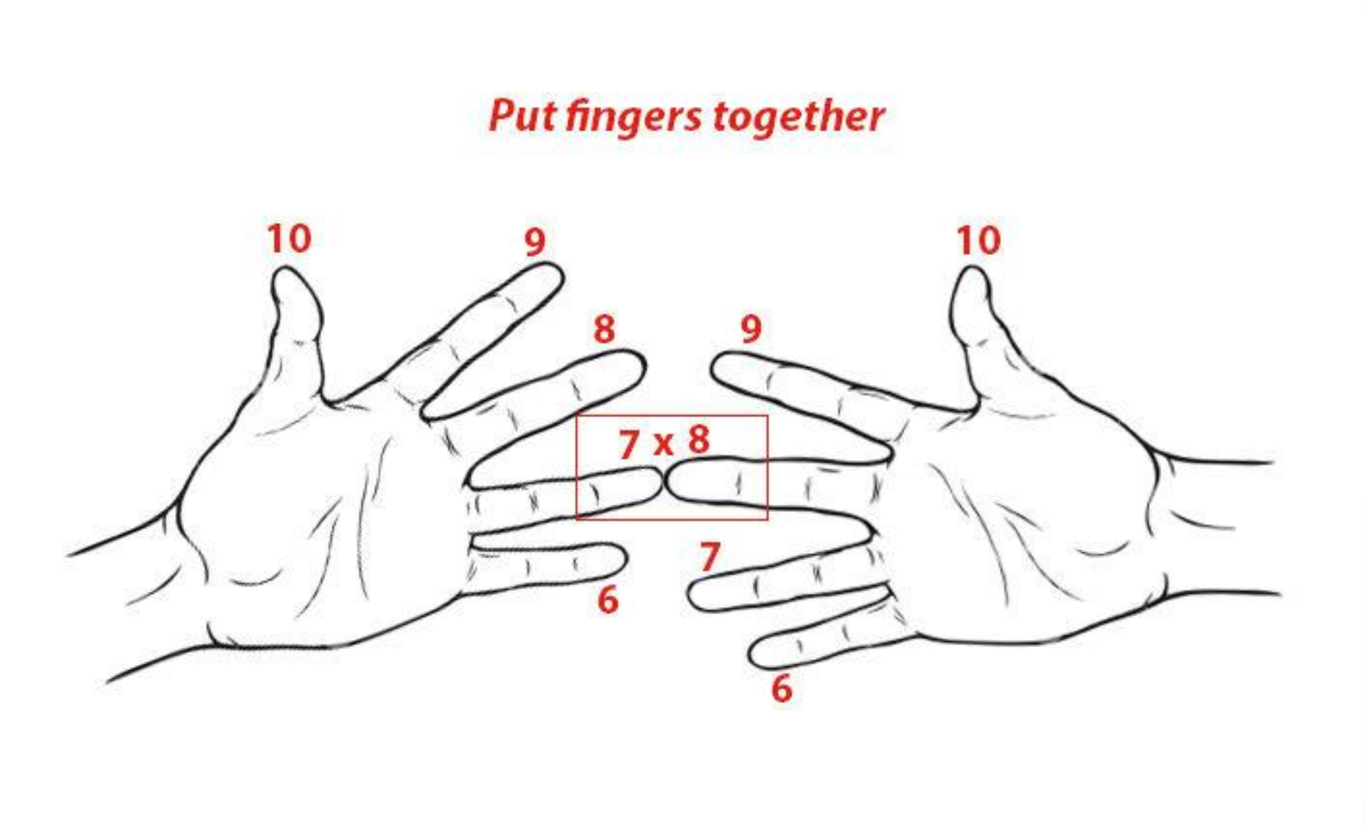
High Definition TV: 1080 rows X 1920 cols

Color and Size

- A display that has 1 bit per pixel will be a black and white display.
- A ***gray-scale*** display supports 256 different shades of black and white. 8 bits per pixel are required.
- There are colored displays with 8, 16 and 24 (**true color**) bits per pixel.
  - A graphic card with 7MB of memory can contain a matrix of 1920x1200 24 bit pixels.

A Brief History of Computer Architecture

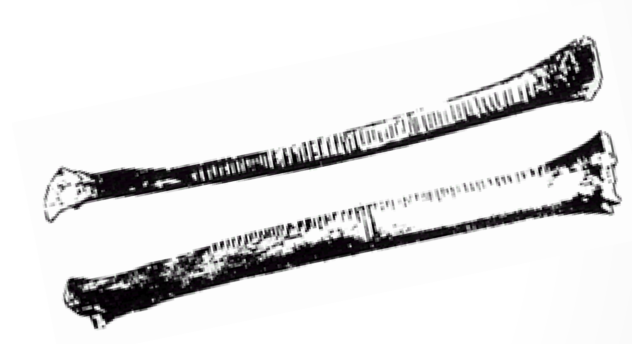
The Latin word "*digit*" is used to refer to a finger (or toe)

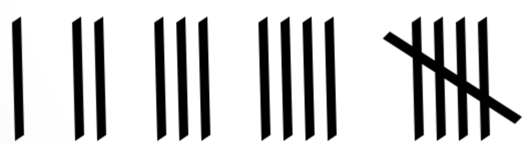


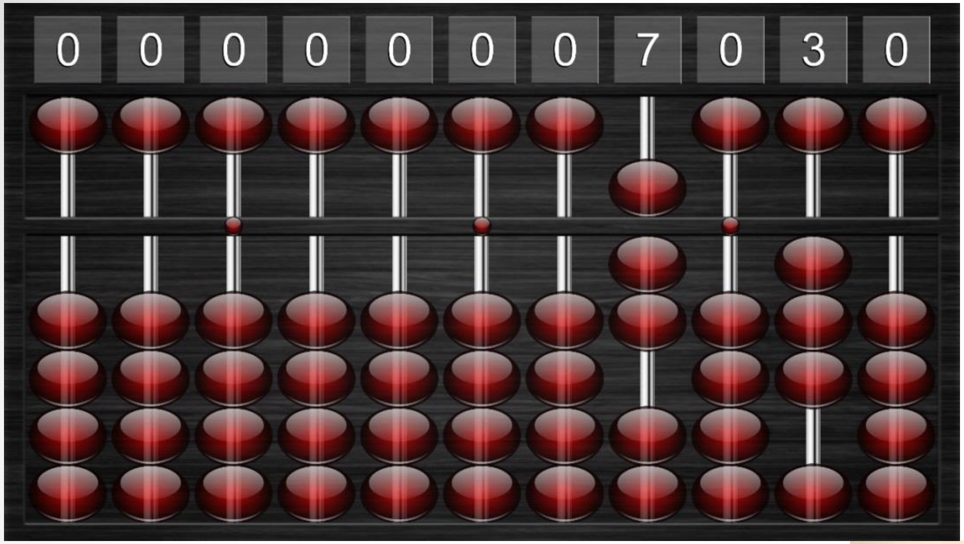
Tally System

A wolf's jawbone more than 20,000 years old, with fifty-five notches in groups of five.

Tally System





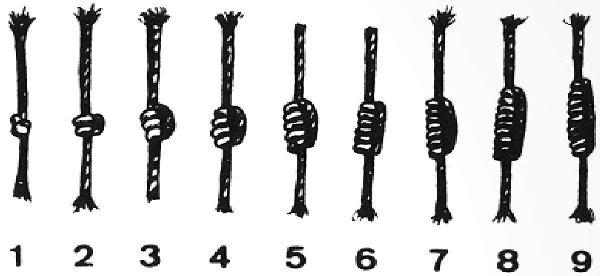


[Abacus]()

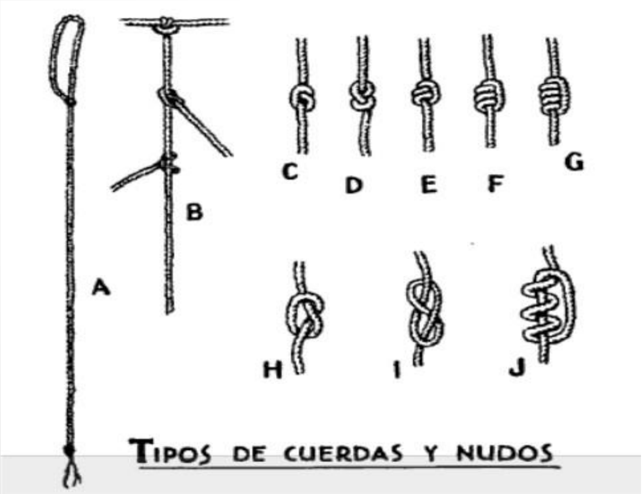
The Inca's quipu

Collecting data and keeping records:

- monitoring tax obligations
- census records
- calendrical information
- military organization



The cords contained numeric and other values encoded by knots in a base ten positional system



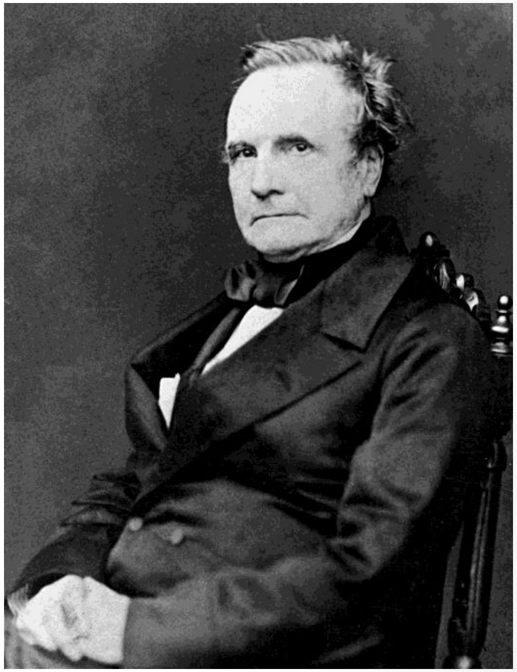
Early non-computer Algorithm

- The Algorithm is related to a 12th century Tashkent scholar: **Muhammad ibn Musa Al'Khowarizmi**
- Invented the algorithm for arithmetic we learn in elementary school
  - Add, subtract etc …

Charles Babbage

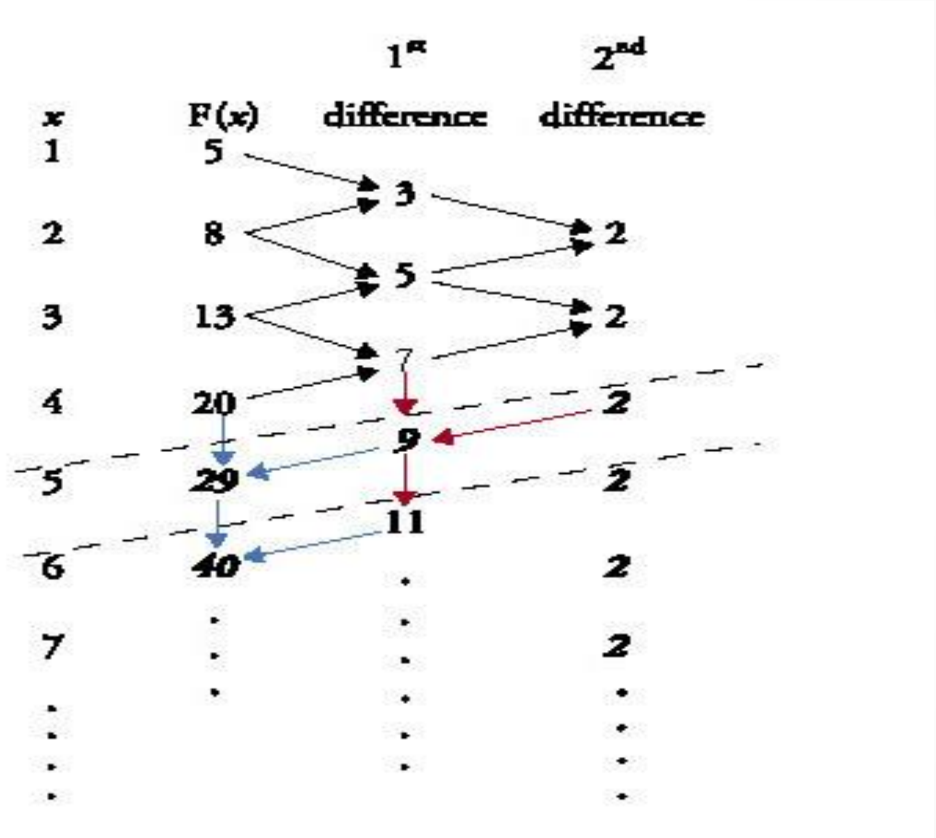
The difference engine

1791 –1871



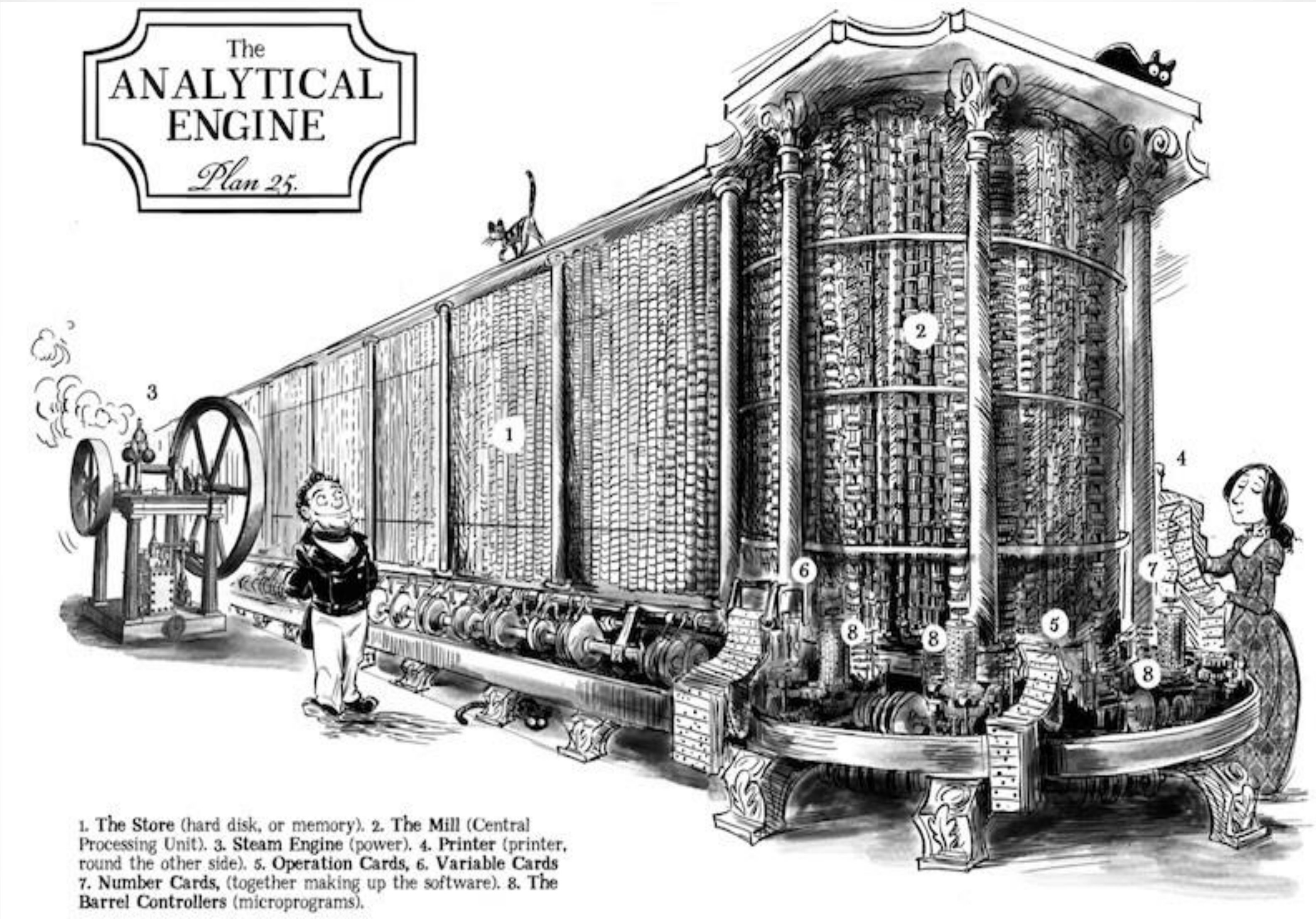
Based on Leibnitz wheel

$$F(x) = x^2 + 4$$



[Charles Babbage's machine]()

[A demo in Computer History Museum]()



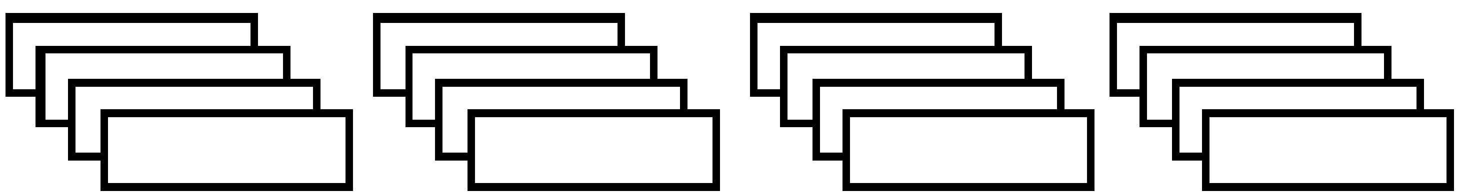
The ANALYTICAL ENGINE
Plan 25.
1
2
3
4
5
6
7
8
1. The Store (hard disk, or memory). 2. The Mill (Central Processing Unit). 3. Steam Engine (power). 4. Printer (printer, round the other side). 5. Operation Cards, 6. Variable Cards 7. Number Cards, (together making up the software). 8. The Barrel Controllers (microprograms).

Analytic Engine
The first conception of a general purpose computer

1. The *store* in which all variables to be operated upon, as well as all those quantities which have arisen from the results of the operations are placed.
2. The *mill* into which the quantities about to be operated upon are always brought.

The *program*

Operation variable1 variable2 variable3



An operation in the *mill* required feeding two punched cards and producing a new punched card for the *store*.

An operation to alter the sequence was also provided!

Analytical Engine - example

In such an operation as $(a\,b + c)\,d$, we should require:

- 4 variable cards with the numbers a, b, c, d
- an operation card directing the machine to multiply a and b together
- a record of the result, namely the product $ab = p$, as a fifth variable card
- an operation card directing the addition of p and c
- a record of the result, namely the sum $p+c=q$, as a 6th variable card
- an operation card directing the machine to multiply q and d together
- a record of the result, namely the product $qd=p_2$, either printed as a final result or punched in a seventh variable card

Analytical Engine

- cycle time: 2.5 seconds to transfer a number from the store to a register in the mill
- addition: 3 seconds
- conditional statements

Analytical Engine

- The programming language was akin to modern day assembly languages.
- Loops and conditional branching were possible,
- The language as conceived would have been Turing-complete
- Three different types of punch cards were used:
  - one for arithmetical operations,
  - one for numerical constants,
  - one for load and store operations, transferring numbers from the store to the arithmetical unit or back.
- There were three separate readers for the three types of cards.

The first programmer
Ada Byron *aka* "Lady Lovelace" 1815-52

She was in touch with:
Faraday, Charles Dickens, De Morgan, etc..

Babbage called her *The Enchantress of Numbers*

Ada Byron a.k.a "Lady Lovelace"

She wrote and published a set of notes, explaining the Analytical Engine

.ht

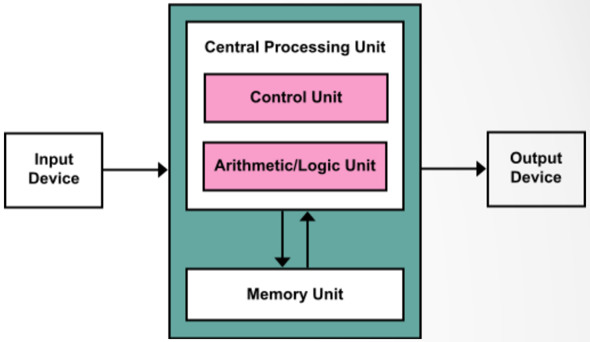
Lovelace's notes were labelled alphabetically from A to G.
In note G, she **describes an algorithm** for the Analytical Engine to compute Bernoulli numbers.
It is considered the first algorithm ever specifically tailored for implementation on a computer
Ada has been cited as the first computer programmer for this reason

Diagram for the computation by the Engine of the Numbers of Bernoulli. See Note G. (page 7

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. $^1V_1$ ◯ 0 0 1 $\boxed{1}$	$^1V_2$ ◯ 0 0 2 $\boxed{2}$	$^1V_3$ ◯ 0 0 4 $\boxed{n}$	Working Variables. $^0V_4$ ◯ 0 0 0	$^0V_5$ ◯ 0 0 0	$^0V_6$ ◯ 0 0 0	$^0V_7$ ◯ 0 0 0	$^0V_8$ ◯ 0 0 0	$^0V_9$ ◯ 0 0 0	$^0V_{10}$ ◯ 0 0 0	$^0V_{11}$ ◯ 0 0 0	
1	×	$^1V_2 \times {}^1V_3$	$^1V_4, {}^1V_5, {}^1V_6$	$\left\{\begin{matrix} ^1V_2 = {}^1V_2 \\ ^1V_3 = {}^1V_3 \end{matrix}\right\}$	$= 2n$	...	2	n	$2n$	$2n$	$2n$						
2	−	$^1V_4 - {}^1V_1$	$^2V_4$	$\left\{\begin{matrix} ^1V_4 = {}^2V_4 \\ ^1V_1 = {}^1V_1 \end{matrix}\right\}$	$= 2n-1$	1	...	...	$2n-1$								
3	+	$^1V_5 + {}^1V_1$	$^2V_5$	$\left\{\begin{matrix} ^1V_5 = {}^2V_5 \\ ^1V_1 = {}^1V_1 \end{matrix}\right\}$	$= 2n+1$	1	...	...	...	$2n+1$							
4	÷	$^2V_5 \div {}^2V_4$	$^1V_{11}$	$\left\{\begin{matrix} ^2V_5 = {}^0V_5 \\ ^2V_4 = {}^0V_4 \end{matrix}\right\}$	$= \frac{2n-1}{2n+1}$	...	...	...	0	0	...	...	...	...	...	$\frac{2n-1}{2n+1}$	
5	÷	$^1V_{11} \div {}^1V_2$	$^2V_{11}$	$\left\{\begin{matrix} ^1V_{11} = {}^2V_{11} \\ ^1V_2 = {}^1V_2 \end{matrix}\right\}$	$= \frac{1}{2} \cdot \frac{2n-1}{2n+1}$	...	2	...	...	...	...	...	...	...	...	$\frac{1}{2} \cdot \frac{2n-1}{2n+1}$	
6	−	$^0V_{13} - {}^2V_{11}$	$^1V_{13}$	$\left\{\begin{matrix} ^2V_{11} = {}^0V_{11} \\ ^0V_{13} = {}^1V_{13} \end{matrix}\right\}$	$= -\frac{1}{2} \cdot \frac{2n-1}{2n+1} = A_0$	...	...	...	...	...	...	...	...	...	...	0	
7	−	$^1V_3 - {}^1V_1$	$^1V_{10}$	$\left\{\begin{matrix} ^1V_3 = {}^1V_3 \\ ^1V_1 = {}^1V_1 \end{matrix}\right\}$	$= n - 1 \ (= 3)$	1	...	n	...	...	...	...	...	...	$n-1$		
8	+	$^1V_2 + {}^0V_7$	$^1V_7$	$\left\{\begin{matrix} ^1V_2 = {}^1V_2 \\ ^0V_7 = {}^1V_7 \end{matrix}\right\}$	$= 2 + 0 = 2$	...	2	...	...	...	...	2					
9	÷	$^1V_6 \div {}^1V_7$	$^3V_{11}$	$\left\{\begin{matrix} ^1V_6 = {}^1V_6 \\ ^0V_{11} = {}^3V_{11} \end{matrix}\right\}$	$= \frac{2n}{2} = A_1$	...	...	...	...	...	$2n$	2	...	...	...	$\frac{2n}{2} = A_1$	
10	×	$^1V_{21} \times {}^3V_{11}$	$^1V_{12}$	$\left\{\begin{matrix} ^1V_{21} = {}^1V_{21} \\ ^3V_{11} = {}^3V_{11} \end{matrix}\right\}$	$= B_1 \cdot \frac{2n}{2} = B_1 A_1$	...	...	...	...	...	...	...	...	...	...	$\frac{2n}{2} = A_1$	$B_1 \cdot$
11	+	$^1V_{12} + {}^1V_{13}$	$^2V_{13}$	$\left\{\begin{matrix} ^1V_{12} = {}^0V_{12} \\ ^1V_{13} = {}^2V_{13} \end{matrix}\right\}$	$= -\frac{1}{2} \cdot \frac{2n-1}{2n+1} + B_1 \cdot \frac{2n}{2}$	...	...	...	...	...	...	...	...	...	...	....	
12	−	$^1V_{10} - {}^1V_1$	$^2V_{10}$	$\left\{\begin{matrix} ^1V_{10} = {}^2V_{10} \\ ^1V_1 = {}^1V_1 \end{matrix}\right\}$	$= n - 2 \ (= 2)$	1	...	...	...	...	...	...	...	...	$n-2$		
13	−	$^1V_6 - {}^1V_1$	$^2V_6$	$\left\{\begin{matrix} ^1V_6 = {}^2V_6 \\ ^1V_1 = {}^1V_1 \end{matrix}\right\}$	$= 2n - 1$	1	...	...	...	...	$2n-1$						
14	+	$^1V_1 + {}^1V_7$	$^2V_7$	$\left\{\begin{matrix} ^1V_1 = {}^1V_1 \\ ^1V_7 = {}^2V_7 \end{matrix}\right\}$	$= 2 + 1 = 3$	1	...	...	...	...	...	3					
15	÷	$^2V_6 \div {}^2V_7$	$^1V_8$	$\left\{\begin{matrix} ^2V_6 = {}^2V_6 \\ ^2V_7 = {}^2V_7 \end{matrix}\right\}$	$= \frac{2n-1}{3}$	...	...	...	...	...	$2n-1$	3	$\frac{2n-1}{3}$				
16	×	$^1V_8 \times {}^3V_{11}$	$^4V_{11}$	$\left\{\begin{matrix} ^1V_8 = {}^0V_8 \\ ^3V_{11} = {}^4V_{11} \end{matrix}\right\}$	$= \frac{2n}{2} \cdot \frac{2n-1}{3}$	...	...	...	...	...	...	...	0	...	...	$\frac{2n}{2} \cdot \frac{2n-1}{3}$	
17	−	$^2V_6 - {}^1V_1$	$^3V_6$	$\left\{\begin{matrix} ^2V_6 = {}^3V_6 \\ ^1V_1 = {}^1V_1 \end{matrix}\right\}$	$= 2n - 2$	1	...	...	...	...	$2n-2$						
18	+	$^1V_1 + {}^2V_7$	$^3V_7$	$\left\{\begin{matrix} ^2V_7 = {}^3V_7 \\ ^1V_1 = {}^1V_1 \end{matrix}\right\}$	$= 3 + 1 = 4$	1	...	...	...	...	...	4					
19	÷	$^3V_6 \div {}^3V_7$	$^1V_9$	$\left\{\begin{matrix} ^3V_6 = {}^3V_6 \\ ^3V_7 = {}^3V_7 \end{matrix}\right\}$	$= \frac{2n-2}{4}$	...	...	...	...	...	$2n-2$	4	...	$\frac{2n-2}{4}$	...	$\left\{\frac{2n}{2} \cdot \frac{2n-1}{3} \cdot \frac{2n-2}{3} = A_3\right\}$	
20	×	$^1V_9 \times {}^4V_{11}$	$^5V_{11}$	$\left\{\begin{matrix} ^1V_9 = {}^0V_9 \\ ^4V_{11} = {}^5V_{11} \end{matrix}\right\}$	$= \frac{2n}{2} \cdot \frac{2n-1}{3} \cdot \frac{2n-2}{4} = A_3$	...	...	...	...	...	...	...	...	0	...		
21	×	$^1V_{22} \times {}^5V_{11}$	$^0V_{12}$	$\left\{\begin{matrix} ^1V_{22} = {}^1V_{22} \\ ^0V_{12} = {}^2V_{12} \end{matrix}\right\}$	$= B_3 \cdot \frac{2n}{2} \cdot \frac{2n-1}{3} \cdot \frac{2n-2}{3} = B_3 A_3$	...	...	...	...	...	...	...	...	...	...	0	
22	+	$^2V_{\ } + {}^2V_{\ }$	$^3V_{\ }$	$\left\{\begin{matrix} ^2V_{12} = {}^0V_{12} \end{matrix}\right\}$	$= A_0 + B_1 A_1 + B_3 A_3$												

The EDVAC

***E**lectronic **D**iscrete **V**ariable **A**utomatic **C**omputer*



- proposed in August 1944
- first **stored-program computer**
- A computer with a **Von Neumann architecture**
- in April 1946 with an initial budget of US$100,000
- almost 6,000 vacuum tubes and 12,000 diodes
- consumed 56 kW of power
- covered 46 m² of floor space
- weighed 7,850 kg
- The full complement of operating personnel was **thirty** people per eight-hour shift
- a binary serial computer with automatic addition, subtraction, multiplication, programmed division and automatic checking
- serial memory capacity of 1,000 44-bit words.
- addition time was 864 μs, and multiplication time was 2,900 μs.
- a floating point arithmetic unit in 1958

Babbage's meaning in history

- John von Neumann (1903 - 1957): universal computing machine consisting of:
  - memory
  - input / output
  - arithmetic/logic unit (ALU)
  - control unit
- based on Babbage's ideas
- 95 % of modern computers are based on the von Neumann architecture