THE HIGH-SPEED ELECTRONIC DIGITAL COMPUTER

CMPT 110

Review from Last Lecture

- Elaborated on the definition of an algorithm.

Review from Last Lecture

- Elaborated on the definition of an algorithm.
- Defined informal (natural) and formal languages.

REVIEW FROM LAST LECTURE

- Elaborated on the definition of an algorithm.
- Defined informal (natural) and formal languages.
- Introduced pseudocode and flowcharts.

REVIEW FROM LAST LECTURE

- Elaborated on the definition of an algorithm.
- Defined informal (natural) and formal languages.
- Introduced pseudocode and flowcharts.
- Provided a brief overview of the history of algorithms and computer programs.

REVIEW FROM LAST LECTURE

- Elaborated on the definition of an algorithm.
- Defined informal (natural) and formal languages.
- Introduced pseudocode and flowcharts.
- Provided a brief overview of the history of algorithms and computer programs.
- Introduced Dijkstra’s Control Structures.

Review from Last Lecture

- Elaborated on the definition of an algorithm.
- Defined informal (natural) and formal languages.
- Introduced pseudocode and flowcharts.
- Provided a brief overview of the history of algorithms and computer programs.
- Introduced Dijkstra's Control Structures.
- Discussed two different fundamental loop structures.

REVIEW FROM LAST LECTURE

- Elaborated on the definition of an algorithm.
- Defined informal (natural) and formal languages.
- Introduced pseudocode and flowcharts.
- Provided a brief overview of the history of algorithms and computer programs.
- Introduced Dijkstra's Control Structures.
- Discussed two different fundamental loop structures.
- Classified algorithms by generation (H/W), by implementation (paradigms), by task, and by technique.

Review from Last Lecture

- Elaborated on the definition of an algorithm.
- Defined informal (natural) and formal languages.
- Introduced pseudocode and flowcharts.
- Provided a brief overview of the history of algorithms and computer programs.
- Introduced Dijkstra's Control Structures.
- Discussed two different fundamental loop structures.
- Classified algorithms by generation (H/W), by implementation (paradigms), by task, and by technique.
- Discussed the need for different languages.

Review from last lecture

- Elaborated on the definition of an algorithm.
- Defined informal (natural) and formal languages.
- Introduced pseudocode and flowcharts.
- Provided a brief overview of the history of algorithms and computer programs.
- Introduced Dijkstra's Control Structures.
- Discussed two different fundamental loop structures.
- Classified algorithms by generation (H/W), by implementation (paradigms), by task, and by technique.
- Discussed the need for different languages.
- Described VB in particular.

CURRENT OBJECTIVES

- Review the advantages of **electronics.**

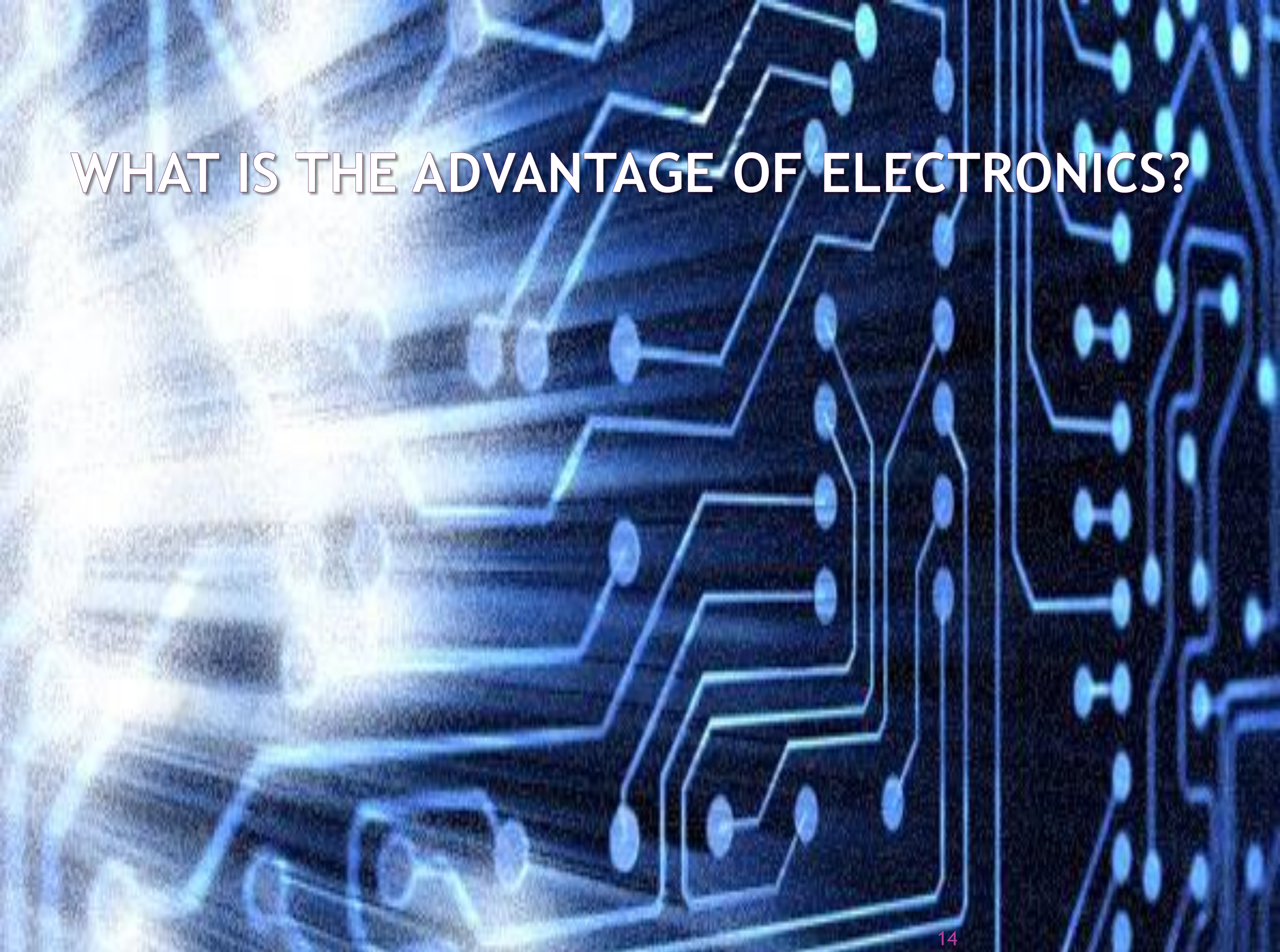
CURRENT OBJECTIVES

- Review the advantages of **electronics.**
- Review the **digital** basis of computers.

CURRENT OBJECTIVES

- Review the advantages of **electronics**.
- Review the **digital** basis of computers.
- Review the advantage of **binary**.

WHAT IS THE ADVANTAGE OF ELECTRONICS?



THE ADVANTAGE OF ELECTRONICS

- Mechanical devices are inherently slower than electronic devices (by a factor of 10^5).

  WHY?

THE ADVANTAGE OF ELECTRONICS

- Mechanical devices are inherently slower than electronic devices (by a factor of 10^5).

  WHY?

- A useful comparison is to compare the speed of a "main memory chip" to a "hard drive."

Mechanical

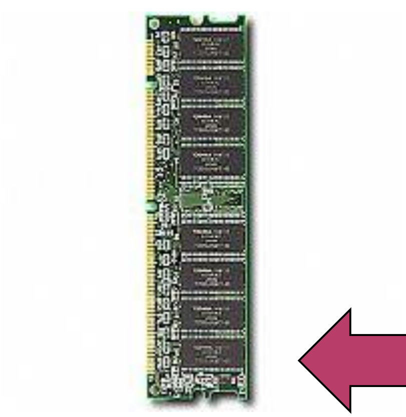
THE ADVANTAGE OF ELECTRONICS

- Mechanical devices are inherently slower than electronic devices (by a factor of 10^5).

  WHY?

- A useful comparison is to compare the speed of a "main memory chip" to a "hard drive."

Mechanical

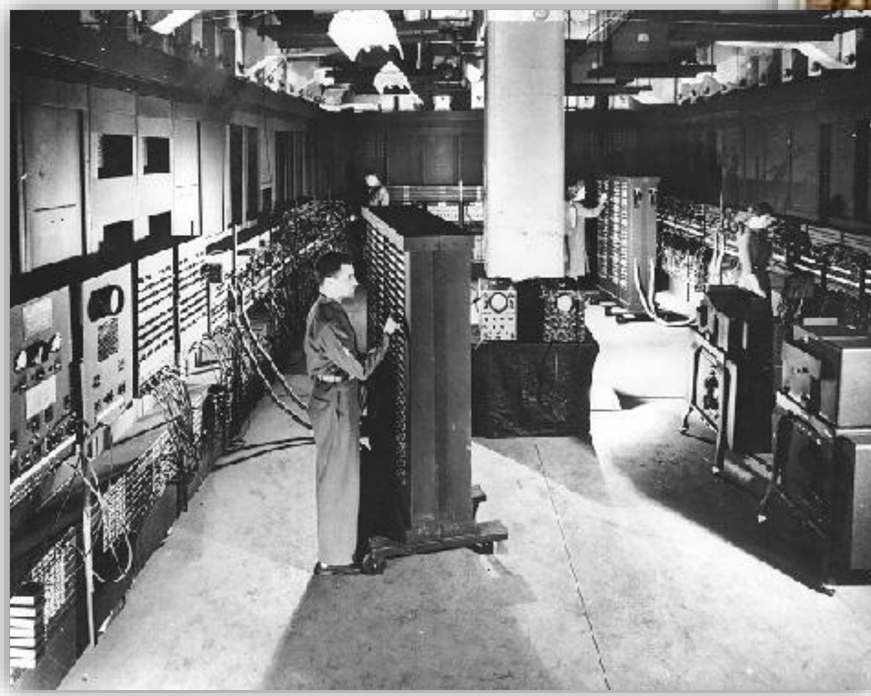


No moving "mechanical" parts

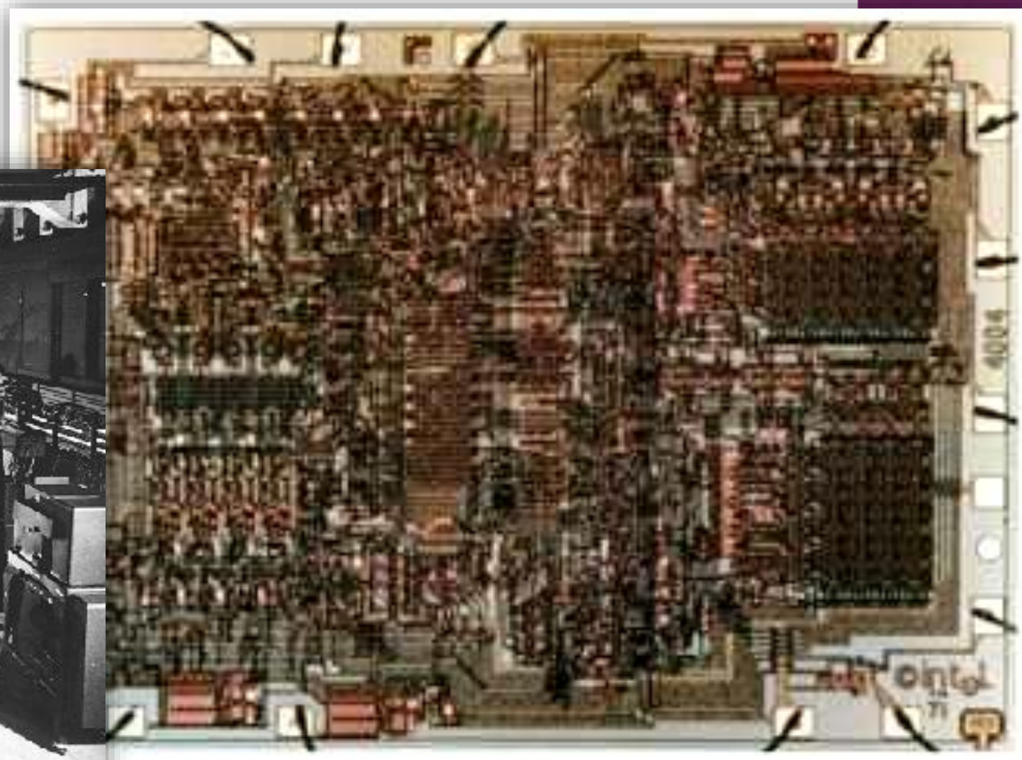
THE ADVANTAGE OF ELECTRONICS

- Electronic devices can be made smaller.

  WHY?



ENIAC, 1945 (filled a room)



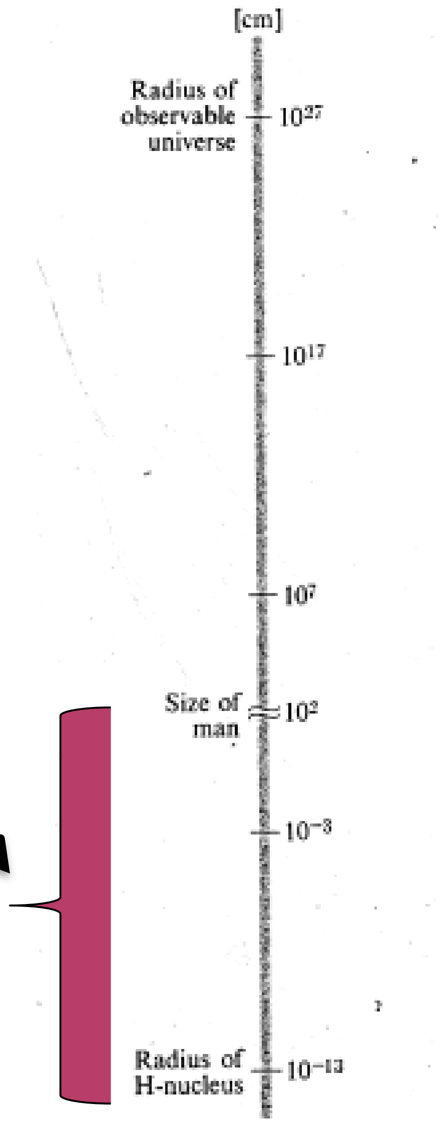
Intel 4004, 1971 (1/4” x 1/8”)

SUMMARY: ADVANTAGE OF ELECTRONICS

- Speed
- Miniaturization
- Other reasons as well

SUMMARY: ADVANTAGE OF ELECTRONICS

- Speed
- Miniaturization
- Other reasons as well



SUMMARY: ADVANTAGE OF ELECTRONICS

- Speed
- Miniaturization
- Other reasons as w

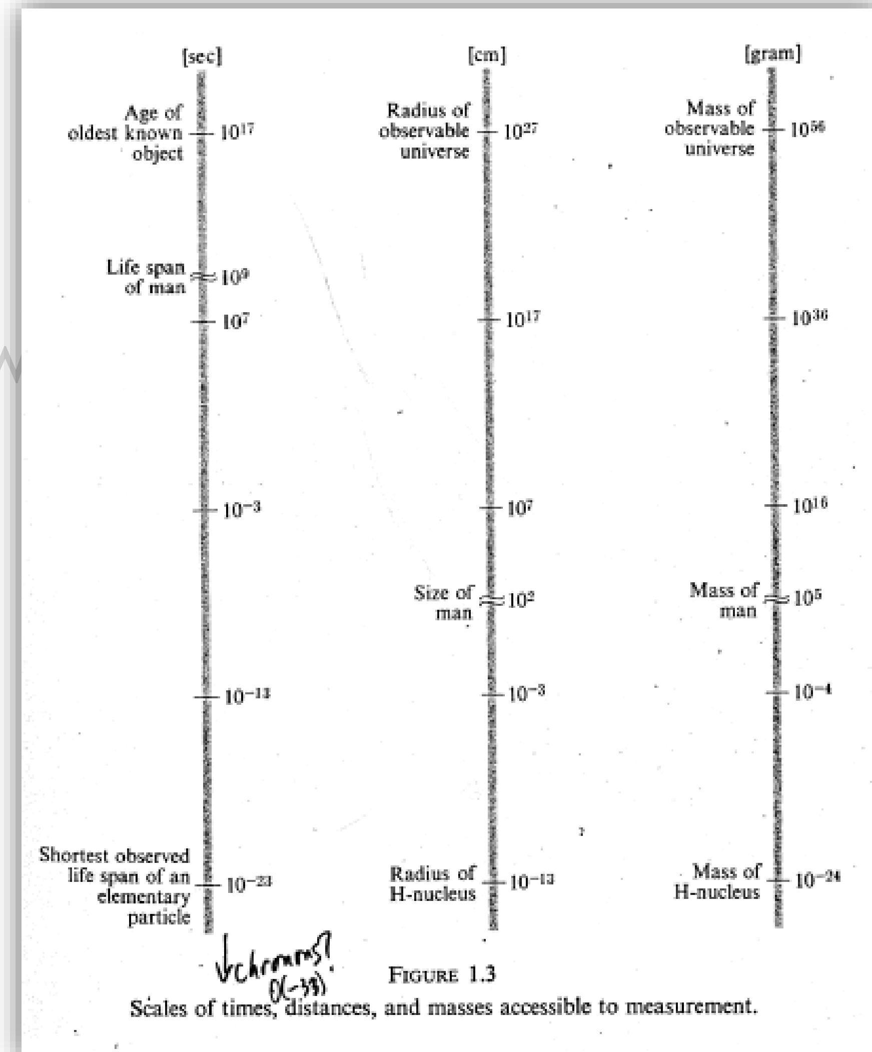
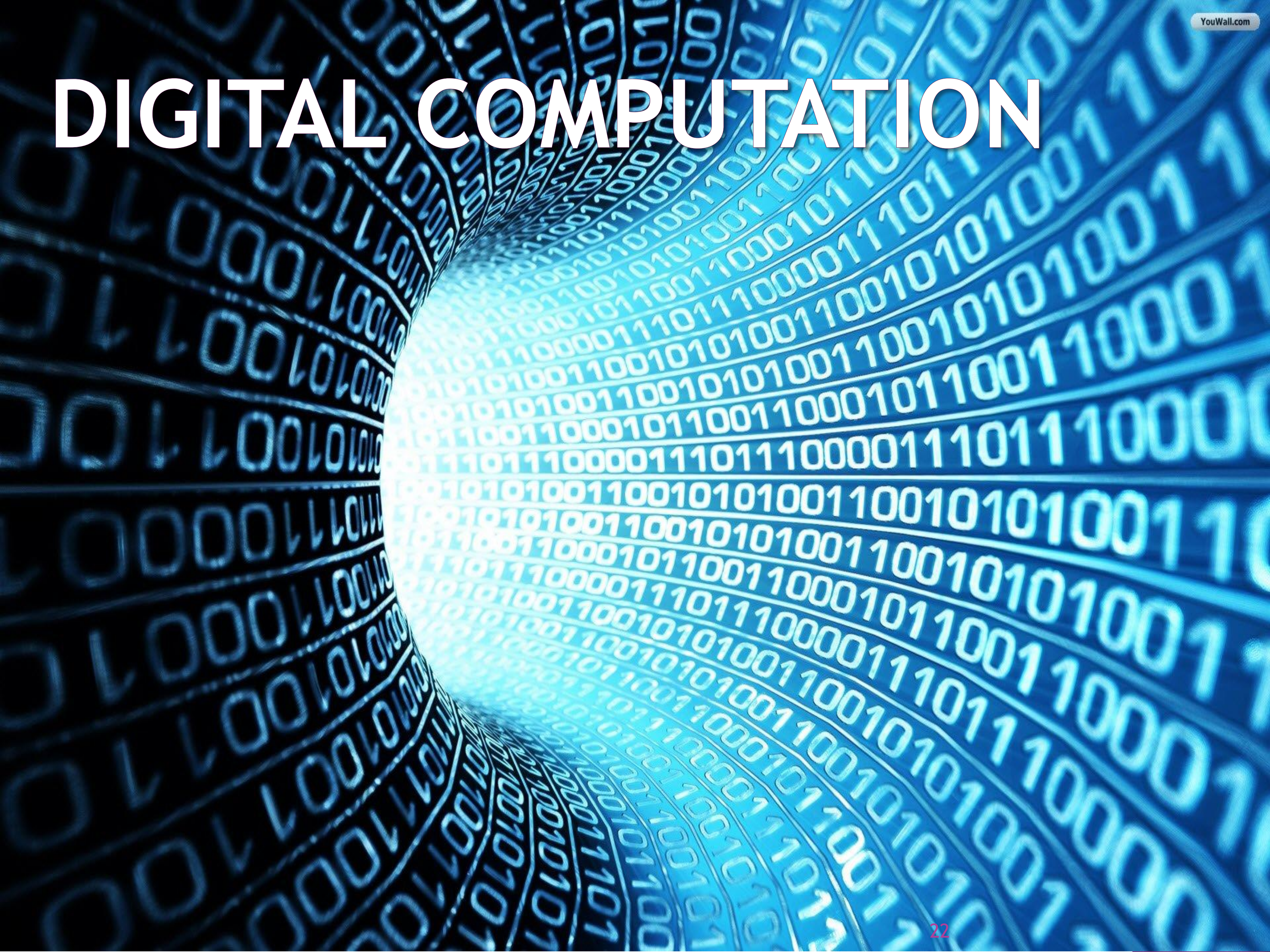


FIGURE 1.3
Scales of times, distances, and masses accessible to measurement.

Note that there are physical limitations

DIGITAL COMPUTATION



DEFINITION

- **Digital System**: Any system that can manipulate discrete elements of information.

FACTS

- Digital computers dominate modern computing.

FACTS

- Digital computers dominate modern computing.
- Analog computers were invented first.

FACTS

- Digital computers dominate modern computing.
- Analog computers were invented first.
- Some kinds of calculations are done better on analog computers.

DIGITAL VERSUS ANALOG

Think of a watch with a continuously varying second hand (analog), compared to a digital clock for representation of information.

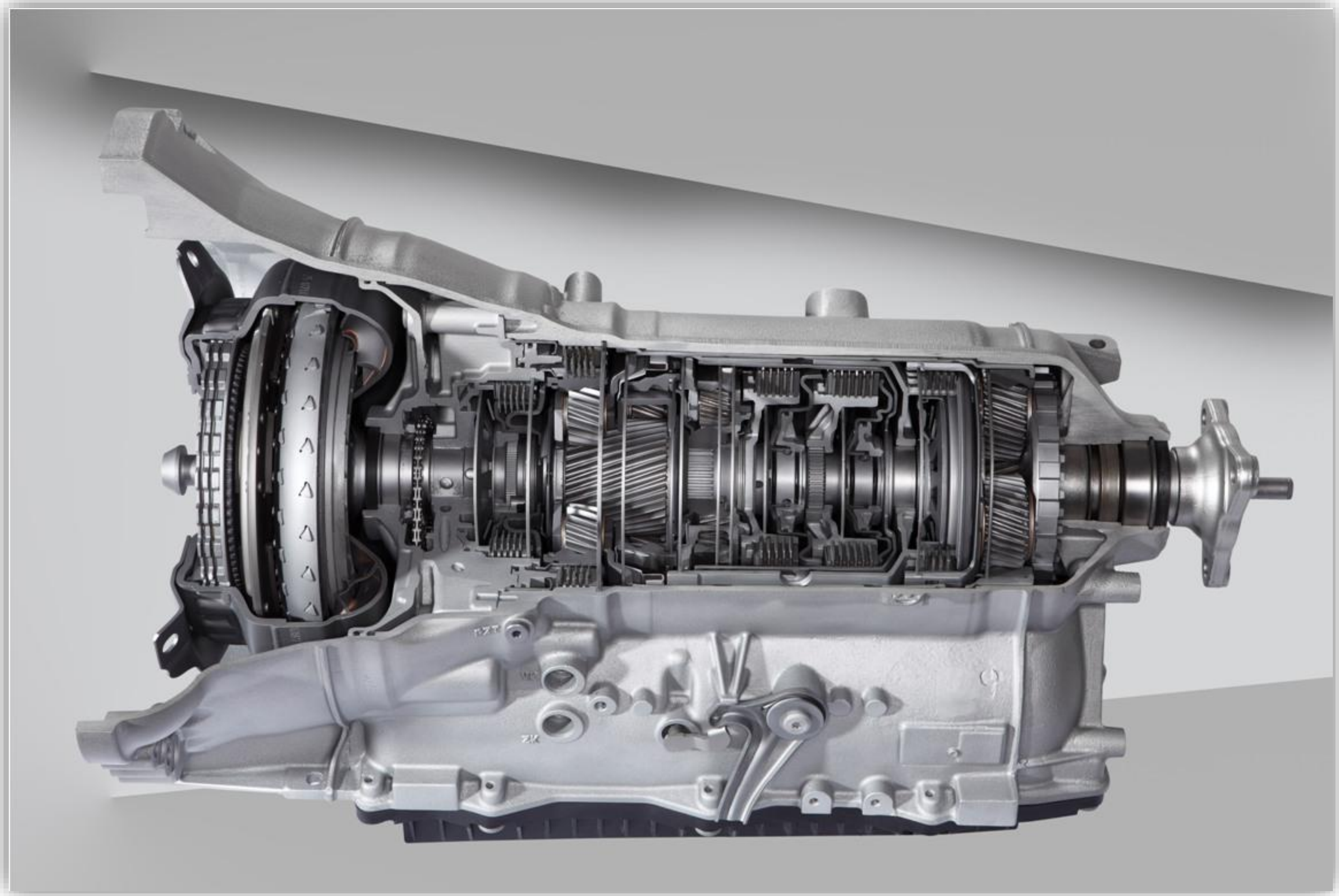
ANALOG COMPUTERS

- An analog computer works on a constant supply of varying signals (and not necessarily even electrical).

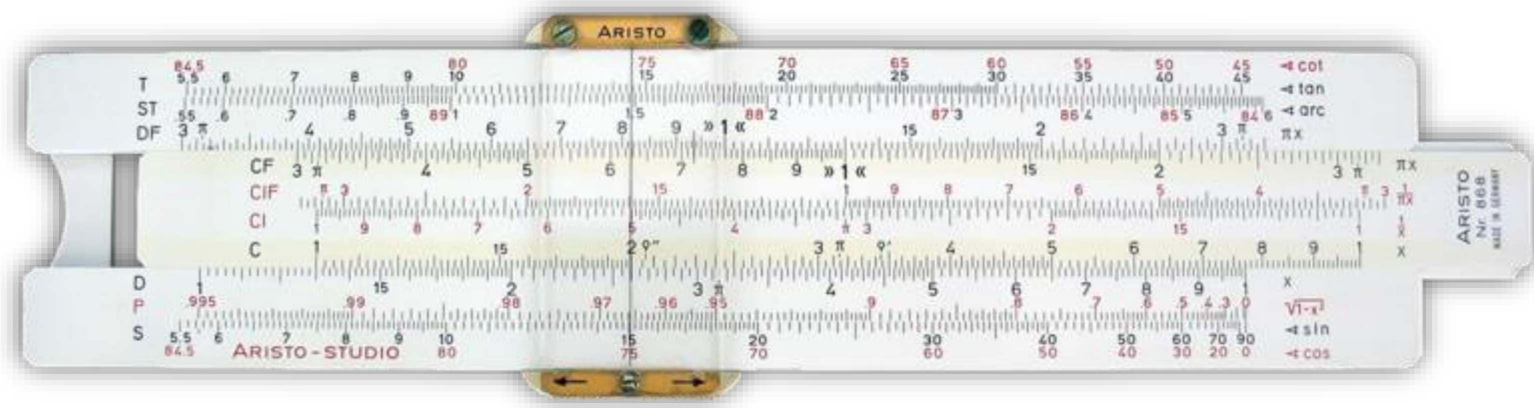
ANALOG COMPUTERS

- An analog computer works on a constant supply of varying signals (and not necessarily even electrical).
- For instance, an automatic transmission in a car is pretty much an analog computer. As the pressure of the pump and the signals on the governor and modulator valve vary, the fluid goes through different valves in the control body and controls which transmission bands are applied at what particular time.

EXAMPLE OF AN ANALOG COMPUTER



ANOTHER EXAMPLE OF AN ANALOG COMPUTER



DIFFERENCE BETWEEN ANALOG AND DIGITAL COMPUTERS

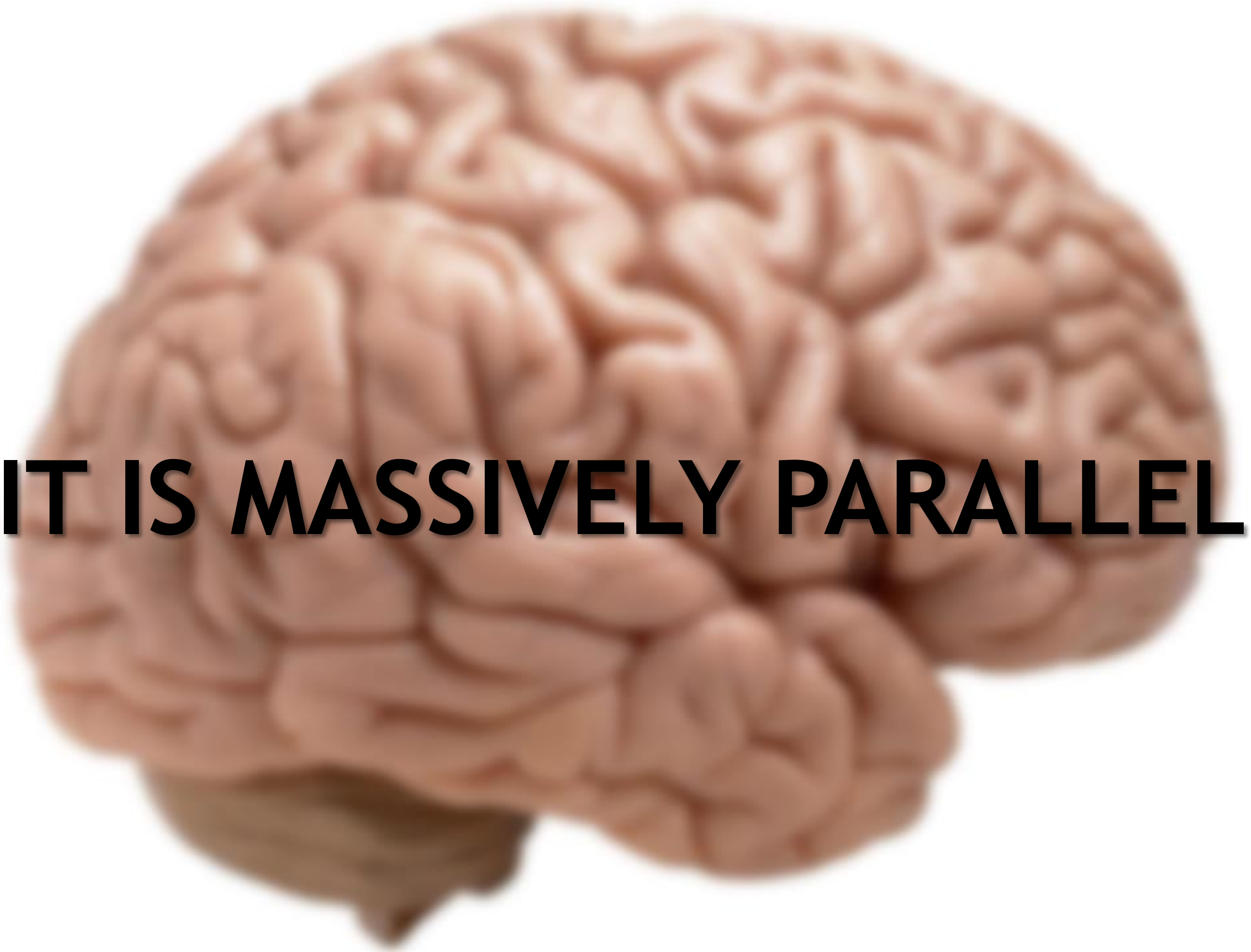
- Analog computers work in **parallel**, meaning that they can carry out multiple tasks simultaneously. A digital computer, even though it may work considerably faster, can only perform one calculation at any one instant. The only way around this in a digital computer is *parallel computing*, where a single machine has multiple processors, and even then, programs must often be rewritten to take advantage of this.

DIFFERENCE BETWEEN ANALOG AND DIGITAL COMPUTERS

- Analog computers work in **parallel**, meaning that they can carry out multiple tasks simultaneously. A digital computer, even though it may work considerably faster, can only perform one calculation at any one instant. The only way around this in a digital computer is *parallel computing*, where a single machine has multiple processors, and even then, programs must often be rewritten to take advantage of this.
- Analog computers utilize continuous variables, while a digital computer works with discrete numbers.

IS THE HUMAN BRAIN ANALOG OR DIGITAL?

IS THE HUMAN BRAIN ANALOG OR DIGITAL?



IT IS MASSIVELY PARALLEL

FACTS

- There are marked advantages to designing digital computers.

FACTS

- There are marked advantages to designing digital computers.
- However, it must be stated that the *historical* reason why we went digital has mostly to do with the electronics revolution (**transistors**, which are little on/off switches).

FACTS

- There are marked advantages to designing digital computers.
- However, it must be stated that the *historical* reason why we went digital has mostly to do with the electronics revolution (transistors, which are little on/off switches).
- In fact, some researchers still maintain that we should be building analog computers.

BASES AND BINARY ARITHMETIC

BASES

- We deal with base 10

BASES

- We deal with base 10

- What about base 20?

BASES

- Base 2 (binary)

BINARY

- A “bit” is a binary digit (either 0 or 1).

BINARY

- A “bit” is a binary digit (either 0 or 1). Binary numbers means we are in base 2.

BINARY

- A "bit" is a binary digit (either 0 or 1). Binary numbers means we are in base 2.
- Base 2 deals with the set, *B* = {0, 1}.

BINARY

- A “bit” is a <u>b</u>inary dig<u>it</u> (either 0 or 1).
- Binary numbers means we are in base 2.
- Base 2 deals with the set, $B = \{0, 1\}$.
- You should be able to count in base 2.

(*i.e.*, 0, 1, 10, 11, 100, 101, 110, …)

Binary-to-Decimal Conversion Table

Binary	Decimal	Binary	Decimal
0	0	10000	16
1	1	10001	17
10	2	10010	18
11	3	10011	19
100	4	10100	20
101	5	10101	21
110	6	10110	22
111	7	10111	23
1000	8	11000	24
1001	9	11001	25
1010	10	11010	26
1011	11	11011	27
1100	12	11100	28
1101	13	11101	29
1110	14	11110	30
1111	15	11111	31

BYTES

- The name for a “string” of 8 bits:

BYTES

- The name for a "string" of 8 bits:
  - 00000000 through 11111111

Bytes

- The name for a "string" of 8 bits:
  - 00000000 through 11111111
- Why 8 bits?

ASCII: AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE

Decimal - Binary - Octal - Hex – ASCII
Conversion Chart

Decimal	Binary	Octal	Hex	ASCII	Decimal	Binary	Octal	Hex	ASCII	Decimal	Binary	Octal	Hex	ASCII	Decimal	Binary	Octal	Hex	ASCII
0	00000000	000	00	NUL	32	00100000	040	20	SP	64	01000000	100	40	@	96	01100000	140	60	`
1	00000001	001	01	SOH	33	00100001	041	21	!	65	01000001	101	41	A	97	01100001	141	61	a
2	00000010	002	02	STX	34	00100010	042	22	"	66	01000010	102	42	B	98	01100010	142	62	b
3	00000011	003	03	ETX	35	00100011	043	23	#	67	01000011	103	43	C	99	01100011	143	63	c
4	00000100	004	04	EOT	36	00100100	044	24	$	68	01000100	104	44	D	100	01100100	144	64	d
5	00000101	005	05	ENQ	37	00100101	045	25	%	69	01000101	105	45	E	101	01100101	145	65	e
6	00000110	006	06	ACK	38	00100110	046	26	&	70	01000110	106	46	F	102	01100110	146	66	f
7	00000111	007	07	BEL	39	00100111	047	27	'	71	01000111	107	47	G	103	01100111	147	67	g
8	00001000	010	08	BS	40	00101000	050	28	(	72	01001000	110	48	H	104	01101000	150	68	h
9	00001001	011	09	HT	41	00101001	051	29	)	73	01001001	111	49	I	105	01101001	151	69	i
10	00001010	012	0A	LF	42	00101010	052	2A	*	74	01001010	112	4A	J	106	01101010	152	6A	j
11	00001011	013	0B	VT	43	00101011	053	2B	+	75	01001011	113	4B	K	107	01101011	153	6B	k
12	00001100	014	0C	FF	44	00101100	054	2C	,	76	01001100	114	4C	L	108	01101100	154	6C	l
13	00001101	015	0D	CR	45	00101101	055	2D	-	77	01001101	115	4D	M	109	01101101	155	6D	m
14	00001110	016	0E	SO	46	00101110	056	2E	.	78	01001110	116	4E	N	110	01101110	156	6E	n
15	00001111	017	0F	SI	47	00101111	057	2F	/	79	01001111	117	4F	O	111	01101111	157	6F	o
16	00010000	020	10	DLE	48	00110000	060	30	0	80	01010000	120	50	P	112	01110000	160	70	p
17	00010001	021	11	DC1	49	00110001	061	31	1	81	01010001	121	51	Q	113	01110001	161	71	q
18	00010010	022	12	DC2	50	00110010	062	32	2	82	01010010	122	52	R	114	01110010	162	72	r
19	00010011	023	13	DC3	51	00110011	063	33	3	83	01010011	123	53	S	115	01110011	163	73	s
20	00010100	024	14	DC4	52	00110100	064	34	4	84	01010100	124	54	T	116	01110100	164	74	t
21	00010101	025	15	NAK	53	00110101	065	35	5	85	01010101	125	55	U	117	01110101	165	75	u
22	00010110	026	16	SYN	54	00110110	066	36	6	86	01010110	126	56	V	118	01110110	166	76	v
23	00010111	027	17	ETB	55	00110111	067	37	7	87	01010111	127	57	W	119	01110111	167	77	w
24	00011000	030	18	CAN	56	00111000	070	38	8	88	01011000	130	58	X	120	01111000	170	78	x
25	00011001	031	19	EM	57	00111001	071	39	9	89	01011001	131	59	Y	121	01111001	171	79	y
26	00011010	032	1A	SUB	58	00111010	072	3A	:	90	01011010	132	5A	Z	122	01111010	172	7A	z
27	00011011	033	1B	ESC	59	00111011	073	3B	;	91	01011011	133	5B	[	123	01111011	173	7B	{
28	00011100	034	1C	FS	60	00111100	074	3C	<	92	01011100	134	5C	\	124	01111100	174	7C	|
29	00011101	035	1D	GS	61	00111101	075	3D	=	93	01011101	135	5D	]	125	01111101	175	7D	}
30	00011110	036	1E	RS	62	00111110	076	3E	>	94	01011110	136	5E	^	126	01111110	176	7E	~
31	00011111	037	1F	US	63	00111111	077	3F	?	95	01011111	137	5F	_	127	01111111	177	7F	DEL

http://web.alfredstate.edu/faculty/weimandn/miscellaneous/ascii/ascii_index.html

ASCII: AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE

$2^0 = 1$

$2^1 = 2$ (0,1)

$2^2 = 4$ (00,01,10,11)

$2^3 = 8$ (000,001, …, 111)

$2^4 = 16$

$2^5 = 32$

$2^6 = 64$

$2^7 = 128$

$2^8 = 256$ (00000000 to 11111111)

BINARY REPRESENTATION OF NUMERIC AND TEXTUAL INFORMATION

- Binary numbering system
  - Base-2
  - Built from ones and zeros
  - Each position is a power of 2

    $1101 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$
- Decimal numbering system
  - Base-10
  - Each position is a power of 10

    $3052 = 3 \times 10^3 + 0 \times 10^2 + 5 \times 10^1 + 2 \times 10^0$

PRINCIPLE ADVANTAGE OF BINARY

- A base 10 computer would rely on electrical switches that had 10 states (think about drifting tolerances).

PRINCIPLE ADVANTAGE OF BINARY

- A base 10 computer would rely on electrical switches that had 10 states (think about drifting tolerances).
- A base 2 computer relies only on two states.

PRINCIPLE ADVANTAGE OF BINARY

- A base 10 computer would rely on electrical switches that had 10 states (think about drifting tolerances).
- A base 2 computer relies only on two states.
- These states can be represented by ON/OFF (*a.k.a.* bistable states)

PRINCIPLE ADVANTAGE OF BINARY

- A base 10 computer would rely on electrical switches that had 10 states (think about drifting tolerances).
- A base 2 computer relies only on two states.
- These states can be represented by ON/OFF (*a.k.a.* bistable states)
- This is the most reliable situation.

EXAMPLES

A Decimal Device

Voltage Level	Digit
0	0
+ 5	1
+10	2
+15	3
+20	4
+25	5
+30	6
+35	7
+40	8
+45	9

Requires a 6% tolerance

EXAMPLES

A Decimal Device

Voltage Level	Digit
0	0
+ 5	1
+10	2
+15	3
+20	4
+25	5
+30	6
+35	7
+40	8
+45	9

Requires a 6% tolerance

A Binary Device

Voltage Level	Digit
0	0
+45	1

(*i.e.*, full on or full off)

Requires no tolerance

PRINCIPLE ADVANTAGE OF BINARY

- Hence, we use BINARY (0/1)

PRINCIPLE ADVANTAGE OF BINARY

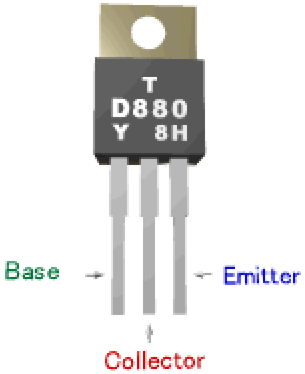
- Hence, we use BINARY (0/1)
- This is consistent with the transistors work (on/off).

PRINCIPLE ADVANTAGE OF BINARY

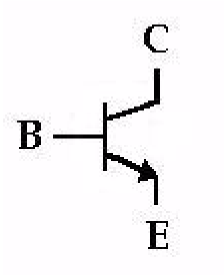
- Hence, we use BINARY (0/1)
- This is consistent with the transistors work (on/off).
- This is also consistent with logic (true/false)

TODAY'S FUNDAMENTAL BINARY DEVICE

- The Transistor



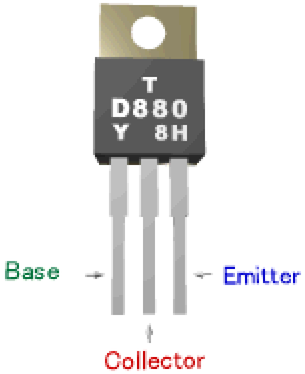
A "macroscopic" transistor

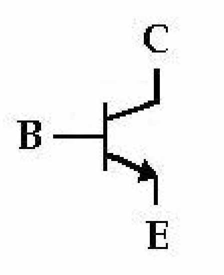


Circuit Symbol

TODAY'S FUNDAMENTAL BINARY DEVICE

- The Transistor





Today, they are *microscopic* on/off switches made from semi-conducting materials (solid-state physics) – the region between metals and non-metals on the periodic table of the elements.

H 1																	He 2
Li 3	Be 4											B 5	C 6	N 7	O 8	F 9	Ne 10
Na 11	Mg 12											Al 13	Si 14	P 15	S 16	Cl 17	Ar 18
K 19	Ca 20	Sc 21	Ti 22	V 23	Cr 24	Mn 25	Fe 26	Co 27	Ni 28	Cu 29	Zn 30	Ga 31	Ge 32	As 33	Se 34	Br 35	Kr 36
Rb 37	Sr 38	Y 39	Zr 40	Nb 41	Mo 42	Tc 43	Ru 44	Rh 45	Pd 46	Ag 47	Cd 48	In 49	Sn 50	Sb 51	Te 52	I 53	Xe 54
Cs 55	Ba 56	Lu 71	Hf 72	Ta 73	W 74	Re 75	Os 76	Ir 77	Pt 78	Au 79	Hg 80	Tl 81	Pb 82	Bi 83	Po 84	At 85	Rn 86
Fr 87	Ra 88	Lr 103	Db 104	Jl 105	Rf 106	Bh 107	Hn 108	Mt 109	110	111	112	113	114	115	116		

La 57	Ce 58	Pr 59	Nd 60	Pm 61	Sm 62	Eu 63	Gd 64	Tb 65	Dy 66	Ho 67	Er 68	Tm 69	Yb 70
Ac 89	Th 90	Pa 91	U 92	Np 93	Pu 94	Am 95	Cm 96	Bk 97	Cf 98	Es 99	Fm 100	Md 101	No 102

Periodic Table courtesy of Chemtutor.

Semi-metals or Metalloids: The staircase-shaped line between metals and non-metals has several elements on or near it that have properties somewhere between the two categories or exhibit some of the qualities of both. For this reason they are called semi-metals or metalloids (meaning metal-like). Advances rely on QUANTUM PHYSICS.

This category includes **silicon** (Si), **germanium** (Ge), **arsenic** (As), **antimony** (Sb), and **tellurium** (Te).
Some chemists also include **boron** (B), **aluminum** (Al), **polonium** (Po), and **astatine** (At) in the semi-metals.

SUMMARY

- Electronic devices are amenable to speed and miniaturization.

SUMMARY

- Electronic devices are amenable to speed and miniaturization.
- Digital design is the basis of contemporary computers, though analog computers exist.

SUMMARY

- Electronic devices are amenable to speed and miniaturization.
- Digital design is the basis of contemporary computers, though analog computers exist.
- Binary is the most reliable digital base.

SUMMARY

- Electronic devices are amenable to speed and miniaturization.
- Digital design is the basis of contemporary computers, though analog computers exist.
- Binary is the most reliable digital base.
- Transistors are binary devices (on/off switches) and are made of semiconducting elements.

END PRESENTATION