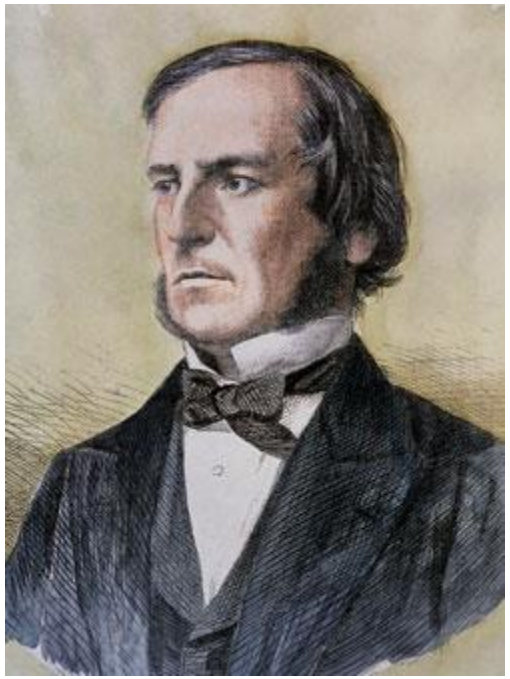
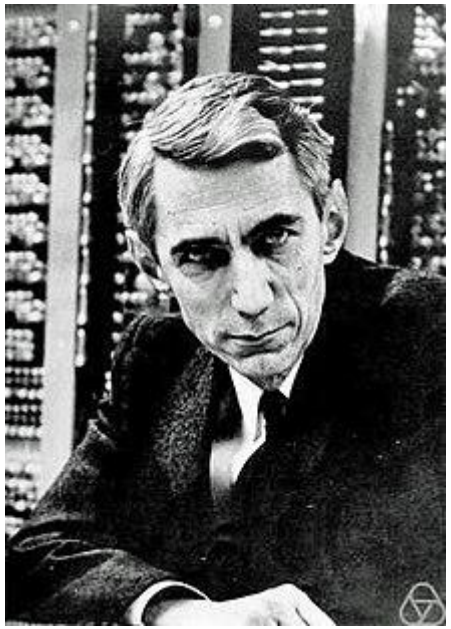
Applied Digital Logic Design



George Boole:
The Mathematical Analysis of Logic (1847)



Claude E. Shannon:
A Symbolic Analysis of Relay and Switching Circuits (1937)

Maurice Karnaugh:
Karnaugh map (1954)

Ludwig Wittgenstein, Emil Leon Post, C. S. Peirce
Truth table of Logic (1921)

BINARY REPRESENTAION: SIGNIFICANCE

- Number system: Decimal, Binary, Hexadecimal, Octal, BCD: Conversion each other
- 0/1: Simplify the computational complexity >> Analog to Digital world
- 0/1: Normalized form of representation
- Multi-leveling the Analog data : Leveling by multiple of 2 for Binary representation
- More number of Level: Accuracy more

BINARY (4-level)		DECIMAL
0	0	0
0	1	1
1	0	2
1	1	3

BINARY (8-level)			DECEIMAL
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

BOOLEAN LOGIC or ALGEBRA

- x and $x^{/}$, y=AB, and y=A+B : few basic Boolean Logic: NOT, AND or OR
- NOT: if input 0/1 : output 1/0
- AND: if and only both input 1, output 1: other wise 0
- OR: if and only both input 0, output 0: other wise 1
- Logical symbol of NOT, AND and OR explained
- Next question:
  - *Practical use of those logic in binary operation*
  - *Practical (Electrical) realization*
- Multiple of such logic functions : Processor (mobile/tap/laptop…)
- Programmable logic devices (PLD)
- Complex programmable logic devices (CPLD)
- Field programmable gate array (FPGA)

FUNCTION and TRUTH TABLE

- Logic to Truth Table: Function generation
- Example: 2 bit ADDER
- Function to Truth Table
- Truth table to Function generation
- Truth table with 1: AND then OR
- Truth table with 0: OR then AND

ELECTRICAL REPRESENTAION

- Switch & Relay: NOT, AND, OR: Power hungry, not practical
- CMOS: NMOS & PMOS form: NOT example drawn
- Electrical point any logic design :

➢ Noise Margin: logical threshold

➢ Timing : Speed

➢ Power

➢ Area

BOOLEAN LAWS

T1 : Commutative Law

(a) $A + B = B + A$

(b) $A\,B = B\,A$

T2 : Associative Law

(a) $(A + B) + C = A + (B + C)$

(b) $(A\,B)\,C = A\,(B\,C)$

T3 : Distributive Law

(a) $A\,(B + C) = A\,B + A\,C$

(b) $A + (B\,C) = (A + B)\,(A + C)$

T4 : Identity Law

(a) $A + A = A$

(b) $A\,A = A$

T5 : Negation Law

(a) $\overline{(\overline{A})} = \overline{A}$

(b) $\overline{(\overline{A})} = A$

T6 : Redundancy Law

(a) $A + A\,B = A$

(b) $A\,(A + B) = A$

T7 :

(a) $0 + A = A$

(b) $1\,A = A$

(c) $1 + A = 1$

(d) $0\,A = 0$

T8 :

(a) $\overline{A} + A = 1$

(b) $\overline{A}\,A = 0$

T9 :

(a) $A + \overline{A}\,B = A + B$

(b) $A\,(\overline{A} + B) = A\,B$

T10 : De Morgan's Theorem

(a) $\overline{(A + B)} = \overline{A}\ \overline{B}$

(b) $\overline{(A\,B)} = \overline{A} + \overline{B}$