Digital System Design and Laboratory Midterm

Switching Circuit

- Asynchronous ↔ Synchronous
- Combinational ↔ Sequential

Numbers and Codes

- 1's complement: Overflow carry wrap around
- 2's complement: Overflow carry don't care
- Binary Code: For display

Boolean Algebra

- Precedence: (brackets), NOT, AND, OR
- (2^{nd} *Distributive*) $X + YZ = (X + Y)(X + Z)$
- (1^{st} *Elimination*) $X + X'Y = X + Y$
- (1^{st} *Consensus*) $XY + X'Z + YZ = XY + X'Z$
- (2^{nd} *Consensus*) $(X + Y)(X' + Z)(Y + Z) = (X + Y)(X' + Z)$
- (*Sum Product Exchange*) $(X + Y)(X' + Z) = XZ + X'Y$
- XOR $\oplus$, XNOR $\equiv$

Truth Table and Karnaugh Map

- Minterm $\sum m(6) = ABC'$
- Maxterm $\prod M(6) = (A' + B' + C)$
- Don't care: $\sum d$ or $\prod D$
- Implicant: A product term
- Prime Implicant: A maximal Implicant
- Essential Prime Implicant: A PI with exclusive minterm(s)
- Quine-McCluskey: PI finding, systematic, useless for human

Circuit Design

- Level counts from output
- Two-level Circuit:
  AND-OR ↔ NAND-NAND ↔ OR-NAND ↔ NOR-OR,
  OR-AND ↔ NOR-NOR ↔ AND-NOR ↔ NAND-AND,
  others are degenerate
- Multi-level: use OR-AND alternating and bubble-pushing to form NAND-NAND and NOR-NOR
- Multiple-output: Pay more attention to essential and common implicants

Delay and Hazard (SOP)

- Delay causes hazard
- When **analyzing** or **constructing hazard free circuit**, DO NOT assume $XX' = 0, X + X' = 1$, i.e. using simple factoring and DeMorgan's law.
- Static 1-Hazard: Change from a implicant to another
- Static 0-Hazard: Multi-level, from $X'X\alpha$.
- Dynamic Hazard: Multi-level, from $X'X\alpha$, single difference propagates ≥ 3 different paths.