GATE 2027 EC, Section 5

Digital circuits

Number representation, Boolean algebra and K-maps, combinational and sequential circuits, timing, data converters, memories and computer organisation.

35 formulas in 6 topics. Equations are numbered section.topic.formula, for example (5.1.1).

5.1 Number representation

n-bit ranges (5.1.1)

Unsigned 0 to $2^n - 1$; two's complement -2^{n-1} to $2^{n-1} - 1$; sign-magnitude, one's complement $\pm(2^{n-1} - 1)$

Two's complement (5.1.2)

$-x = \text{invert all bits} + 1$

IEEE 754 single (5.1.3)

$(-1)^s \times 1.M \times 2^{E-127}$; 1 sign, 8 exponent, 23 mantissa bits

IEEE 754 double (5.1.4)

Bias 1023; 1 sign, 11 exponent, 52 mantissa bits

Gray code (5.1.5)

$g_{MSB} = b_{MSB}$; $g_i = b_i \oplus b_{i+1}$

BCD, excess-3 (5.1.6)

BCD: 4 bits per decimal digit; excess-3 = BCD + 0011

5.2 Boolean algebra

De Morgan (5.2.1)

$(AB)' = A' + B'$; $(A + B)' = A'B'$

Absorption (5.2.2)

$A + AB = A$; $A(A + B) = A$; $A + A'B = A + B$

Consensus (5.2.3)

$AB + A'C + BC = AB + A'C$

XOR / XNOR (5.2.4)

$A \oplus B = A'B + AB'$; $(A \oplus B)' = AB + A'B'$

Number of functions (5.2.5)

n variables → 2^{2^n} Boolean functions

K-map (5.2.6)

Group 1, 2, 4, 8… adjacent 1s (edges wrap); each group of 2^k removes k variables

Universal gates (5.2.7)

NAND alone or NOR alone can build any function

5.3 Combinational circuits

Half adder (5.3.1)

$S = A \oplus B$; $C = AB$

Full adder (5.3.2)

$S = A \oplus B \oplus C_{in}$; $C_{out} = AB + C_{in}(A \oplus B)$

Carry look-ahead (5.3.3)

$G_i = A_iB_i$; $P_i = A_i \oplus B_i$; $C_{i+1} = G_i + P_iC_i$

Multiplexer / decoder (5.3.4)

2^n:1 MUX needs n select lines; n-to-2^n decoder; any n-variable function fits a 2^n:1 MUX

Static CMOS gate (5.3.5)

Pull-up (PMOS) network is the dual of pull-down (NMOS); NAND = parallel PMOS + series NMOS

5.4 Sequential circuits

Flip-flop equations (5.4.1)

SR: $Q^+ = S + R'Q$ ($SR = 0$); JK: $Q^+ = JQ' + K'Q$; D: $Q^+ = D$; T: $Q^+ = T \oplus Q$

Counters (5.4.2)

Mod-N needs $\lceil \log_2 N \rceil$ flip-flops; ring counter N states with N FFs; Johnson 2N states

Ripple counter speed (5.4.3)

$f_{max} \approx 1/(n \cdot t_{pd})$

Setup constraint (5.4.4)

$T_{clk} \geq t_{clk \to q} + t_{logic,max} + t_{setup}$ (− clock skew if capture clock is later)

Hold constraint (5.4.5)

$t_{clk \to q} + t_{logic,min} \geq t_{hold}$ (+ clock skew)

Maximum frequency (5.4.6)

$f_{max} = 1/T_{clk,min}$, set by the critical path

5.5 Data converters

Resolution (5.5.1)

$LSB = V_{FS}/2^n$

Quantisation (5.5.2)

Error ±½ LSB; SQNR ≈ 6.02n + 1.76 dB (full-scale sine)

DAC output (5.5.3)

$V_o = V_{ref} \times (code)/2^n$ (binary-weighted or R-2R)

ADC speed (5.5.4)

Flash: 1 step, $2^n - 1$ comparators; SAR: n steps; counter/dual-slope: up to 2^n steps

Sample and hold (5.5.5)

Acquisition time, droop rate $= I_{leak}/C_{hold}$

5.6 Memories and computer organisation

Memory size (5.6.1)

n address lines → 2^n locations; capacity $= 2^n \times$ word length

Cells (5.6.2)

SRAM: 6T, no refresh; DRAM: 1T1C, needs refresh; ROM: non-volatile

CPU time (5.6.3)

Time = instruction count × CPI × clock period

Pipeline speedup (5.6.4)

k stages, n instructions: $S = nk/(k + n - 1) \to k$ as $n \to \infty$

Average memory access (5.6.5)

$T = h \cdot t_{cache} + (1 - h) \cdot t_{miss}$ path

Addressing modes (5.6.6)

Immediate, direct, indirect, register, register indirect, indexed, relative

My notes