# Digital Electronics Important Questions PDF

Digital Electronics usually covers number systems and codes, Boolean algebra and minimisation, combinational circuits, sequential circuits with counters and registers, and logic families, memories and converters. The most repeated questions are code conversions, K-map and Quine-McCluskey minimisation, MUX-based design, flip-flop conversion, counter design and TTL or CMOS characteristics.

## Key takeaways

- Number systems and codes: base conversion, complements, BCD, Gray and excess-3 codes.
- Minimisation: Boolean laws, K-maps with don't cares and the Quine-McCluskey method.
- Combinational and sequential circuits: adders, MUX, decoders, flip-flops, counters, registers and state machines.
- Logic families and memories: TTL, CMOS, fan-out, noise margin, ROM, RAM, PLA, PAL, ADC and DAC.
- These are practice questions based on common syllabus topics, not verified past papers.

## About these important questions

These are **practice questions** built from topics that appear in most Indian B.Tech syllabi for Digital Electronics, including AKTU and universities that follow the AICTE model curriculum. They are **not verified past papers**, and we do not claim that any question will appear in your exam. Unit numbers and topic order differ between universities, so first match each unit below with your own university syllabus. For real previous papers, use your university's official archive. AKTU students can download them from the AKTU old question paper page.

Each unit lists key definitions, derivations or theory questions, numericals with short answer outlines, and long-answer questions. Use the outlines to check your method, then write full solutions yourself.

## Unit 1: Number Systems and Codes

**Key definitions:** radix, binary, octal, hexadecimal, 1's and 2's complement, signed magnitude, BCD, excess-3 code, Gray code, ASCII, parity, Hamming code.

**Numericals (answer outline):**

- Convert (45) decimal to binary. Answer: 101101. Convert (101101.101) binary to decimal. Answer: 45.625.

- Convert binary 1011 to Gray code. Outline: keep the MSB, then XOR adjacent bits: 1, 1⊕0 = 1, 0⊕1 = 1, 1⊕1 = 0, giving 1110.
- Perform 9 - 5 in 4-bit 2's complement and show the carry that is discarded.
- Add two BCD numbers and apply the +6 correction when a digit is above 9 or a carry is produced.
- Write the excess-3 code for decimal 59. Answer: 1000 1100.

## Unit 2: Boolean Algebra and Minimisation

**Key definitions:** Boolean laws, duality, De Morgan's theorems, minterms and maxterms, canonical SOP and POS, K-map, prime implicant, essential prime implicant, don't-care condition.

**Numericals (answer outline):**

- Simplify F(A, B, C, D) = Σm(0, 1, 2, 3). Answer: F = A'B' (a single group of four).
- Simplify a four-variable function with don't cares using a K-map, and write both SOP and POS forms.
- Minimise a five-variable or six-term function using the Quine-McCluskey tabulation method and a prime implicant chart.
- Prove the consensus theorem: AB + A'C + BC = AB + A'C.
- Implement a given function using only NAND gates or only NOR gates.

## Unit 3: Combinational Circuits

**Key definitions:** half and full adder, half and full subtractor, ripple carry adder, carry look-ahead adder, multiplexer, demultiplexer, decoder, encoder, priority encoder, magnitude comparator, code converter.

**Important questions:**

- Design a full adder and implement it with two half adders and an OR gate.
- Explain the carry look-ahead adder and why it is faster than a ripple carry adder. Write the generate (G = AB) and propagate (P = A ⊕ B) terms.
- Implement F(A, B, C) = Σm(1, 3, 5, 6) using an 8:1 MUX and then using a 4:1 MUX.
- Design a 3-to-8 line decoder and use it to implement a full adder.
- Design a 2-bit magnitude comparator.
- Design a BCD to excess-3 code converter.

## Unit 4: Sequential Circuits

**Key definitions:** latch, flip-flop, SR, JK, D and T flip-flops, race-around condition, master-slave flip-flop, edge triggering, setup and hold time, registers, shift registers, ripple and synchronous counters, Mealy and Moore machines.

**Results to remember:** characteristic equations: JK: Q(next) = JQ' + K'Q; D: Q(next) = D; T: Q(next) = T ⊕ Q.

**Important questions:**

- Explain the race-around condition in a JK flip-flop and how a master-slave flip-flop removes it.
- Convert a JK flip-flop to a D flip-flop and to a T flip-flop using excitation tables.
- Design a synchronous mod-6 counter using JK flip-flops.
- Design a 3-bit ripple (asynchronous) up counter and draw its timing diagram. A mod-N ripple counter needs n flip-flops where $2^n \geq N$.
- Explain the ring counter and the Johnson counter. An n-bit Johnson counter has 2n states.
- Design a sequence detector (for example 1011) as a Mealy machine and as a Moore machine.

## Unit 5: Logic Families, Memories and Converters

**Key definitions:** fan-in, fan-out, propagation delay, power dissipation, noise margin, figure of merit, TTL, CMOS, ECL, totem-pole and open-collector outputs, tri-state logic, ROM, PROM, EPROM, SRAM, DRAM, PLA, PAL, DAC, ADC.

**Numericals (answer outline):**

- Find the noise margins of a logic family: NMH = VOH - VIH and NML = VIL - VOL.
- An 8-bit DAC has a full-scale output of 10.2 V. Find the step size. Outline: $10.2/(2^8 - 1) = 10.2/255 = 0.04$ V.
- Find the conversion time of a successive approximation ADC for n bits and a given clock period (about n clock cycles).

**Long-answer questions:** Explain a TTL NAND gate with totem-pole output. Explain a CMOS inverter and why CMOS has low static power. Compare PLA and PAL. Explain the R-2R ladder DAC and the successive approximation ADC.

## How to score well in Digital Electronics

- Draw truth tables, K-maps, excitation tables and logic diagrams for every design question.
- Circle K-map groups clearly and write the term next to each group.
- For counter design, follow the fixed steps: state diagram, state table, excitation table, K-maps, logic diagram.
- Practise timing diagrams for flip-flops and counters.

## Short-answer (2-mark) questions for Digital Electronics

Short questions test definitions and one-line results. Practise answering each in two to four lines:

1. Convert hexadecimal 3F to decimal. (Answer: 63.)
2. Why is Gray code called a unit-distance code?
3. What is the 2's complement of 0101? (Answer: 1011.)
4. State De Morgan's theorems.
5. What is an essential prime implicant?
6. How many select lines does a 16:1 MUX need? (Answer: 4.)
7. Differentiate between a latch and a flip-flop.
8. What is the race-around condition?

9. Define setup time and hold time.
10. How many states does a 4-bit Johnson counter have? (Answer: 8.)
11. Define fan-out.
12. Differentiate between SRAM and DRAM.

## A 7-day revision plan for Digital Electronics

Use this plan in the last week before the exam. Adjust it to the units and weightage in your own syllabus.

- **Day 1 (Unit 1, Number Systems and Codes):** revise the definitions, then solve the numericals and one long answer from this unit without notes.
- **Day 2 (Unit 2, Boolean Algebra and Minimisation):** revise the definitions, then solve the numericals and one long answer from this unit without notes.
- **Day 3 (Unit 3, Combinational Circuits):** revise the definitions, then solve the numericals and one long answer from this unit without notes.
- **Day 4 (Unit 4, Sequential Circuits):** revise the definitions, then solve the numericals and one long answer from this unit without notes.
- **Day 5 (Unit 5, Logic Families, Memories and Converters):** revise the definitions, then solve the numericals and one long answer from this unit without notes.
- **Day 6:** attempt one full three-hour practice paper made from these questions, or an official previous paper from your university archive.
- **Day 7:** go through your formula sheet, the short-answer list and every question you got wrong in the practice paper.

## Frequently asked questions

### Which topics in Digital Electronics are most important?

K-map minimisation, MUX-based design, flip-flop conversions, synchronous counter design and sequence detectors are the most common long questions. Number system conversions are common short questions.

### How do I design a synchronous counter step by step?

Draw the state diagram, write the present-state and next-state table, fill the flip-flop excitation table, simplify each input with a K-map and draw the final circuit with a common clock.

### Is Digital Electronics useful for software careers?

Yes. Binary arithmetic, logic, state machines and memory concepts help with computer organisation, embedded systems and low-level programming, and they appear in many placement aptitude rounds.

### Are these questions from real past papers?

No. These are practice questions based on topics that are common in B.Tech syllabi. They are not verified past papers, and no question is promised to appear in your exam. Check your own

university syllabus and official previous papers; AKTU students can use the AKTU old question paper archive.

**Is this PDF free?**

Yes. Fill in the short form on this page and the PDF downloads right away.

## Sources and further reading

- AKTU: Old question papers (official archive)
- AKTU: Syllabus
- NPTEL: Free online engineering courses
- AICTE: All India Council for Technical Education

## About this PDF

Prepared by the Uncodemy academic team from common B.Tech syllabus topics. These are practice questions, not past papers. Corrections: info@uncodemy.com.

More free resources and courses: uncodemy.com | info@uncodemy.com | +91 98183 66550